Universidad de Granada

Departamento de Ciencias de la Computación e
Inteligencia Artificial

Fusion of Knowledge towards the Identification of Genetic Profiles in the Systemic Inflammation Problem

Tesis Doctoral

Cristina Rubio Escudero

Granada, Diciembre de 2007

Editor: Editorial de la Universidad de Granada
Autor: Cristina Rubio Escudero
D.L.: Gr. 2722 - 2007
ISBN: 978-84-338-4689-1

Universidad de Granada

Fusion of Knowledge towards the Identification of Genetic Profiles in the Systemic Inflammation Problem

MEMORIA QUE PRESENTA

Cristina Rubio Escudero

PARA OPTAR AL GRADO DE DOCTOR EN INFORMATICA

Diciembre de 2007

DIRECTORES

Igor Zwir
Óscar Cordón García

Departamento de Ciencias de la Computación e Inteligencia Artificial

La memoria titulada “Fusion of Knowledge towards the Identification of Genetic Profiles in the Systemic Inflammation Problem”, que presenta Cristina Rubio Escudero para optar al grado de doctor, ha sido realizada dentro del programa de doctorado “Diseño, Análisis y Aplicaciones de Sistemas Inteligentes”del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada bajo la dirección de los doctores D. Igor Zwir y D. Oscar Cordón García.

Granada, Diciembre de 2007

El Doctorando

Los Directores

Fdo: Cristina Rubio Escudero

Fdo: Igor Zwir y Óscar Cordón García

Agradecimientos

Me gustaría dar las gracias en primer lugar a mis directores de tesis, por su dedicación y ayuda a lo largo de todo este tiempo. También al grupo de Soft Computing and Intelligent Information Systems, muy especialmente a Francisco Herrera, y mencionar a Igor, Coral, Marcela, Rocio, Oscar, Cristopher y Pat, que tanto me han ayudado en todo, siempre, para cualquier cosa que he necesitado. Hay cosas que no se pueden agradecer en unas líneas. Quiero también agradecer a la gente del Centro Alemán de Investigación del Cáncer (DKFZ), que me permitieron pasar con ellos una temporada investigando y aprendiendo cosas de gran valor tanto científico como personal, y a la gente del departamento de Lenguajes y Sistemas Informáticos de la Universidad de Sevilla, que han hecho que me sienta como en casa. Y por último, agradecer a mi familia y a mis amigos por tanto cariño y tantos buenos ratos pasados, y los que quedan...

Tesis Doctoral parcialmente subvencionada por el Ministerio de Educación y Ciencia con los proyectos TIN2006-12879 y BIO2004-0270-E

MEC
TIN2006-12879 y BIO2004-0270-E

X

Contents

List of Figures

List of Tables

Introduction

I Approach to the Problem

The sequencing of the human genome and the spectacular advances in molecular biology are opening the door to the systematic application of computational techniques for the studying of the molecular processes underlying biological systems (Durbin et al., 1998). One of the greatest challenges of the postgenomic era is the discovery of when, how and for how long a gene is turned on or off. Particularly, microarray technology has revolutionized modern biomedical research by its capacity to monitor changes in relative RNA abundance for thousands of genes simultaneously (Brown and Botstein, 1999), while traditional methods can only handle the one-gene at a time approach. The development and application of microarray technology has risen many problems that need to be addressed by the collective knowledge and skills of the mathematical, physicists, computer and biological scientists. The greatest challenge in the microarray technology development is the analysis of the data generated. The current bottleneck in the processing of microarray data occurs after the data are generated; the magnitude of the problem is proving to be on a par with developing the technology itself.

The objective when carrying out microarray experiments essentially is identifying genes which behave in a different way between experimental conditions, and from a computational point of view, to develop analytical methods to retrieve them. We will show that the application of conventional statistical methods to microarray data analysis returns different results applied over the same set of data, since the methods do not identify all genes with different behavior between experimental conditions; moreover, none of the methods subsumes

the results obtained by the other methods. The microarray analysis methods applied are not capable to extract on their own all the information present in microarray data sets. Missing genes, (i.e., genes not recovered by some of the methods) might contain significant information for the experiment under study.

There is a lack of decision making methodologies capable to decide which methods are the most appropriate for a given microarray experiment. Therefore, new methods or meta-methods are necessary to suggest which microarray analysis method is appropriate for the identification of genes behaving different between experimental conditions.

Along with microarray analytical techniques, the reconstruction of genetic networks is becoming an essential task to understand data generated by microarray techniques (Gregory, 2005). The enormous amount of information generated by this high-throughput technique is raising the interest in network models to represent and understand biological systems. Systems biology research arises at this point as the field to explore the life regulation processes in a cohesive way making use of new technologies. Proteins have a main role in the regulation of genes (Rice and Stolovitzky, 2004), but unfortunately, for the vast majority or biological datasets available, there is no information about the level of protein activity. Therefore, we use the expression level of the genes, obtained from microarray experiments, as an indicator of the activity of the proteins they generate. Gene networks represent these gene interactions. A gene network can be described as a set of nodes which usually represent genes, proteins or other biochemical entities. Node interaction is represented with edges corresponding to biologic relations.

There is a wide range of models available for the build up of genetic networks. One of the criterium to classify such models whether they represent static or dynamic relations. Static modeling explains causal interactions by searching for mutual dependencies between the gene expression profiles of different genes. Dynamic models represent the expression of a gene at a given time based on the expression of the other genes in the network at a previous time (van Someren et al., 2002).

II Objectives

Our goal throughout this work will be to propose a methodology for the broad and complete analysis of gene expression data coming from microarray experiments. We aim to create an integrated resource that enables researchers to extract results from experiments carried out applying microarray technology.

We will provide a computational framework to identify reliable targets providing statistically meaningful results and characterizing novel expression patterns. This environment will also fuse genetic information from different sources including experimental knowledge and biological databases. This integrated analysis suite will allow us to sort the information acquired by functional groups, gene interaction through time, metabolic pathways, disease associations and DNA sequence information.

The research work will be performed on a set of data acquired from an experiment carried out over an inflammation and host response to injury problem. Inflammation is a hallmark of many human diseases. Understanding the inflammation process is critical because the body uses inflammation to protect itself from infection or injury (e.g., crushes, massive bleeding, or a serious burn) which, in extreme cases (e.g., car accidents or gun shootings), can lead to massive organ malfunction and death. According to the U.S. Centers for Disease Control's National Center for Health Statistics, unintentional injury is the leading cause of death for people ages 1 to 35. The host response to trauma and burn is a collection of biological and pathological processes that depends critically upon the regulation of the human immuno-inflammatory response. This study, in part carried out at the Cellular Injury and Adaptation Laboratory, Washington University School of Medicine, is a piece of a Large-scale Collaborative Research Project sponsored by the National Institute of General Medical Sciences (www.gluegrant.org)

Besides the importance of the biological problem under study, analysis of the set of gene expression profiles obtained from this experiment is complex, given the number of samples taken and variance due to treatment, time, and subject phenotype. Therefore, we believe this problem is typical and informative as a RNA microarray case study.

The objectives to achieve are:

- Creation of a methodology to perform a broad and complete analysis of gene expression data from microarray experiments. This methodology will successfully extract all reliable targets from microarray experiments providing statistically meaningful results by means of combining the advantages of several microarray analysis methods. The gene information will be extracted based on the differential profiles genes exhibit over time, treatment, patient, or other experimental conditions. Classical microarray analysis methods are not capable to extract all the information present in microarray data sets, therefore this methodology will overcome this problem.

- Functional annotation of the gene expression information, grouped in differential profiles, obtained from microarray experiments, with data from biological databases. These annotations will be achieved through mining of the databases and fusing the knowledge retrieved. The mining will be performed with already existing algorithms as well as new algorithms developed by us as part of this PhD. work.The databases used throughout this work are human gene and diseases, biological pathways, gene product and DNA sequence. Information obtained from this databases will allow us to, on the one hand, asses the cohesiveness of the differential profiles obtained from our experiments, and on the other hand, perform a deeper research of the experiment under study, providing a wider and more complete set of information to the experts.

- Comparison of gene network creation methods and fusion of the information given by these genetic networks with the already known problem related information. We will compare the behavior of static vs. dynamic modeling. On the one hand, static modeling searches for relations between the expression levels of genes throughout time. The relation found by static methods might not only be similar behavior throughout time (direct correlation), but an inverse correlation (two genes having exactly opposite profiles over time), a proximity on the expression values (distance measures such as Euclidean Distance or City block distance). On the other hand, dynamic modeling retrieves temporal dependencies among genes, i.e., it detects dependencies of a gene at time t_{+1} related to some other(s) gene at time t.We will make a study of the performance of these two models over a real problem, the immuno-inflammatory response problem. The gene networks, created based on the differential profiles obtained from the problem being studied, will provide us with information about the regulation process underlying the genes retrieved as significant from the experiment.

III Summary

To achieve the proposed objectives, this work is divided in several chapters which are structured as follows:

Chapter 1, where we introduce some basic biology concepts. We include a brief description of the components which make up living organisms, followed by the molecular processes that underlie biological systems, and a short note to describe biological methods which study DNA sequences. We also make an introduction to the Bioinformatic topic, a relatively young discipline, which has

attracted the attention of a great part of the scientific community and has grown up in a world of high-throughput large volume data that requires automatic analysis to enable us to make use of it all. We also make an introduction to microarray experiments, describing the different technologies underlying DNA microarrays and their scope of application.

Chapter 2, where we review the microarray development and analysis state of the art. We describe the analysis processes necessary to extract information from microarray experiments, such as high and low level analysis, and show the results obtained applying some conventional microarray analysis methods to a set of data acquired from an experiment carried out over inflammation and host response to injury problem. Such problem, which will be analyzed in detail throughout this work, is described in detail as part of this chapter.

Chapter 3, where we propose a conceptual clustering approach which combines the advantages of several microarray analysis methods in an attempt to retrieve all significant gene expression changes from microarray experiments by identifying differential profiles (i.e., sets of genes with coordinate changes in RNA abundance). We define both, gene expression profiles and differential profiles, which are a basic component of our methodology and will be used throughout this work. We show the results obtained applying the proposed methodology on the inflammation and host response to injury problem, and compare them with the results obtained applying conventional microarray analysis methods. We also apply the methodology to an artificial microarray data set created to test our proposal.

Chapter 4, where we provide some biological meaning to the differential profiles obtained in Chapter 3, which are a basic components of our methodology. We mine into several biological databases, human gene and diseases, biological pathways, gene product and DNA sequence, in order to, on the one hand, asses the biological cohesiveness of the differential profiles obtained from the problem being studied, and on the other hand, acquire a further understanding of the gene behavior in an inflammation process.

Chapter 5, where we apply different methodologies to create gene networks from the differential profiles obtained from the problem under study. We compare genetic network building algorithms from two of the main categories they can be divided in: static and dynamic models. We also use both discrete and continuous data inputs, to get a better knowledge of how this methods work.

Chapter 1

Conceptos de Biología y Bioinformática

Todos los seres vivos están formados por células que comparten una maquinaria común para sus funciones más básicas. Los seres vivos, aunque infinitamente diversos por fuera, son muy similares por dentro (Fig. 1.1). En este primer capítulo expondremos las características universales de todos los seres vivos, analizando brevemente la diversidad celular, y veremos cómo, gracias a un código común en el que están escritas las especificaciones de todos los organismos, es posible leer, medir y desentrañar estas especificaciones para alcanzar un conocimiento coherente de todas las formas de vida, de las más simples a las más complejas.

1 Material Genético en la Célula

Se calcula que las células llevan evolucionando y diversificándose más de tres mil millones y medio de años (Stryer et al., 2003). Todas las células vivas, sin ninguna excepción conocida, guardan su información hereditaria en el material genético: moléculas de ADN (abreviatura de ácido desoxirribonucleico) de doble cadena –dos largos polímeros paralelos no ramificados formados por cuatro tipos de monómeros (el material esencia o unidad con la cual se construye un polímero.)–. Estos monómeros están unidos entre sí formando una larga secuen-

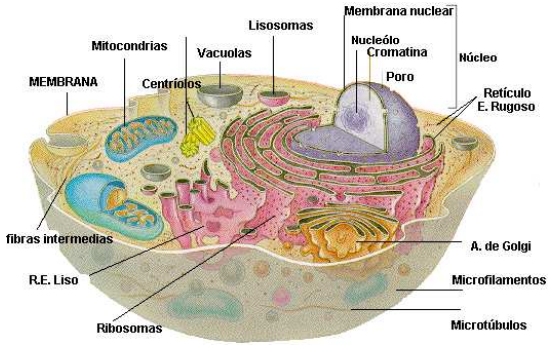


Figure 1.1: Célula eucariota y detalle de sus orgánulos

cia lineal que codifica la información genética de la célula (Stryer et al., 2003; Alberts et al., 2003).

Los organismos vivos pueden clasificarse en dos grupos atendiendo a su estructura: los organismos **eucariotas** y los **procariotas**. Los eucariotas guardan su ADN en un compartimiento intracelular denominado núcleo. Los procariotas no presentan un comportamiento nuclear diferenciado para almacenar su ADN. Las plantas, los hongos y los animales son eucariotas; las bacterias son procariotas (Alberts et al., 2003).

Para comprender los mecanismos biológicos, primero tenemos que conocer la estructura de la molécula de ADN. Cada monómero de una de las cadenas sencillas del ADN –denominado **nucleótido** (Fig. 1.2)– tiene dos partes: un azúcar (la desoxirribosa, Fig. 1.3) con un grupo fosfato unido y una *base* que puede ser adenina (A), guanina (G), citosina (C) o timina (T) (Fig. 1.4). Cada azúcar está unido al siguiente azúcar de la cadena por el grupo fosfato mediante un enlace fosfodiéster, formando un polímero cuyo eje central está compuesto por los azúcares fosfato y del cual sobresalen las bases. El polímero de ADN crece por la unión de monómeros a uno de sus extremos. En el caso de una cadena sencilla de ADN, los monómeros pueden incorporarse al polímero de forma aleatoria, sin un orden preestablecido, ya que todos los nucleótidos pueden unirse entre sí en el sentido del crecimiento del polímero de ADN.

Por el contrario, en la célula viva existe una limitación, ya que el ADN no se sintetiza como una cadena libre aislada sino sobre un patrón o molde de ADN de otra cadena preexistente. Las bases contenidas en la cadena patrón se unen a las bases de la nueva cadena siguiendo una estricta norma de complementariedad: A se une a T, y C se une a G (Fig. 1.5). Este emparejamiento controla la selección

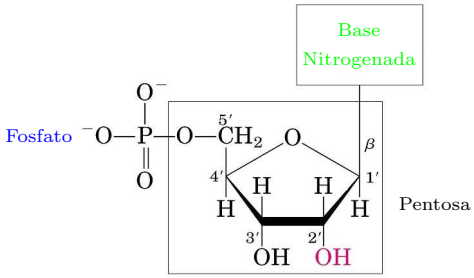
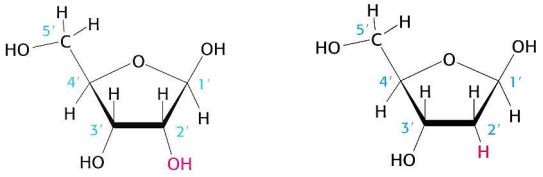


Figure 1.2: Esquema de un nucleótido

del monómero que se añade a la cadena. De esta forma, una estructura de doble cadena consiste en dos secuencias complementarias de A, C, G y T. El orden de la secuencia es muy importante, ya que en él reside la información contenida en el ácido nucleico. La orientación viene dada en el sentido 5'-3' o 3'-5', donde el 5' representa el extremo terminal del fosfato y el 3' el extremo final del átomo de carbono de la desoxirribosa. Además, las dos cadenas de nucleótidos se enrollan una sobre la otra generando una doble hélice (Fig. 1.6).



(a) Ribosa (b) Desoxirribosa

Figure 1.3: Azúcares

Los enlaces establecidos entre las bases son débiles si se comparan con las uniones azúcar-fosfato del resto del esqueleto. Esta debilidad permite separar las dos cadenas de ADN sin forzar la rotura de su esqueleto. Cada una de las cadenas puede comportarse como un molde para la generación de su pareja mediante la formación de pares de bases específicos. Es precisamente esta capacidad para la generación de nuevas hebras de ADN la que le permite crear nuevas células con idéntico material genético a la célula replicada.

El ADN tiene la capacidad de expresar su información para gobernar el comportamiento de otras moléculas de la célula. El mecanismo responsable de este proceso es el mismo en todos los organismos vivos y se inicia con la síntesis

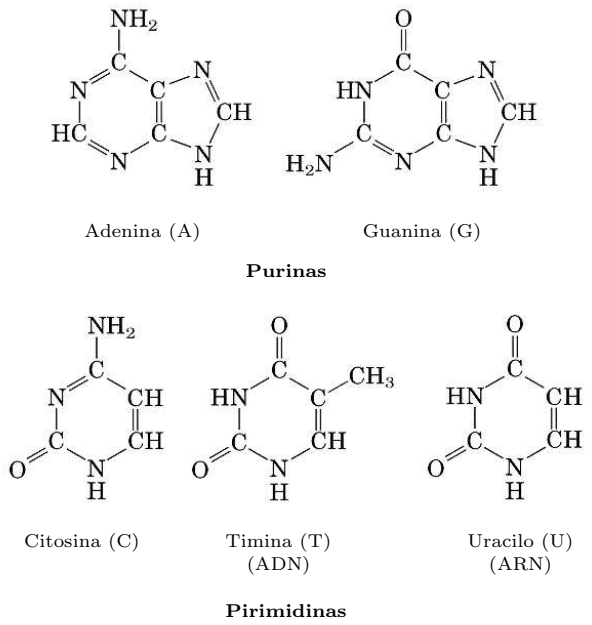


Figure 1.4: Bases nitrogenadas

secuencial de dos tipos de moléculas: el ácido ribonucleico (ARN) y las proteínas. El proceso comienza con la polimerización sobre un patrón, denominada **transcripción**, proceso en el que diferentes segmentos de la secuencia de ADN se utilizan como molde para la síntesis de moléculas cortas de un polímero muy relacionado con el ADN: el **ácido ribonucleico** o **ARN**. Después de un proceso complejo denominado **traducción**, muchas de estas moléculas de ARN se utilizan para dirigir la síntesis de polímeros de una clase química radicalmente diferente: las *proteínas*.

En el ARN, el esqueleto del polímero está formado por azúcares ligeramente diferentes a los del ADN –ribosa en lugar de desoxirribosa– y, además, una de las cuatro bases es diferente –uracilo (U) (Fig. 1.4) en el lugar de la timina (T)–, pero las otras tres bases –A, C, G– son las mismas y se emparejan con su complementaria, como en el ADN –la A, la U, la C y la G del ARN se unen con la T, la A, la G y la C del ADN, respectivamente. Durante la transcripción, los monómeros de ARN se seleccionan para la polimerización del ARN sobre una cadena molde de ADN, de la misma manera que se seleccionan los monómeros de ADN durante la replicación del ADN.

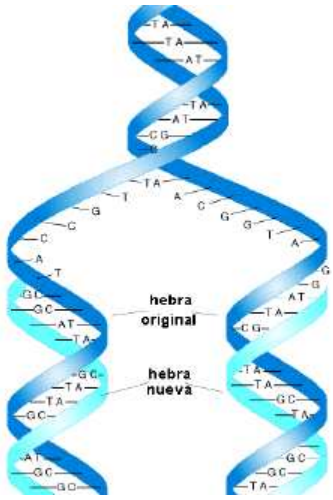


Figure 1.5: Replicación de las hebras de ADN

El resultado de la transcripción es un polímero de ARN que contiene una parte de la información genética de la célula, aunque escrita en un alfabeto diferente de monómeros de ARN en lugar de monómeros de ADN.

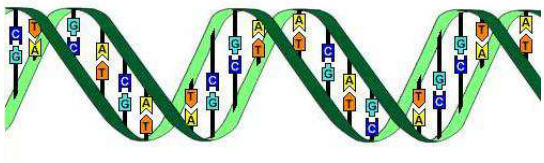


Figure 1.6: Estructura en doble cadena del ADN

El papel principal de muchas secuencias de ADN es el de codificar secuencias de las **proteínas**, el componente mas activo de la célula, que participan en todos los procesos esenciales. Al igual que el ADN y el ARN, las proteínas son polímeros no ramificados formadas por monómeros, los **aminoácidos**, muy diferentes de los del ADN o el ARN y de los que existen veinte tipos diferentes en lugar de tan sólo cuatro (Fig. 1.8). Los aminoácidos tienen una estructura central semejante por la que pueden unirse entre ellos. Junto a esta estructura central, se encuentra un grupo lateral que confiere a cada aminoácido su carácter químico característico. Cada una de las moléculas proteicas o *polipéptidos*, formadas por la unión de varios aminoácidos siguiendo una secuencia determinada, se pliega en una estructura tridimensional elaborada y muy bien definida que

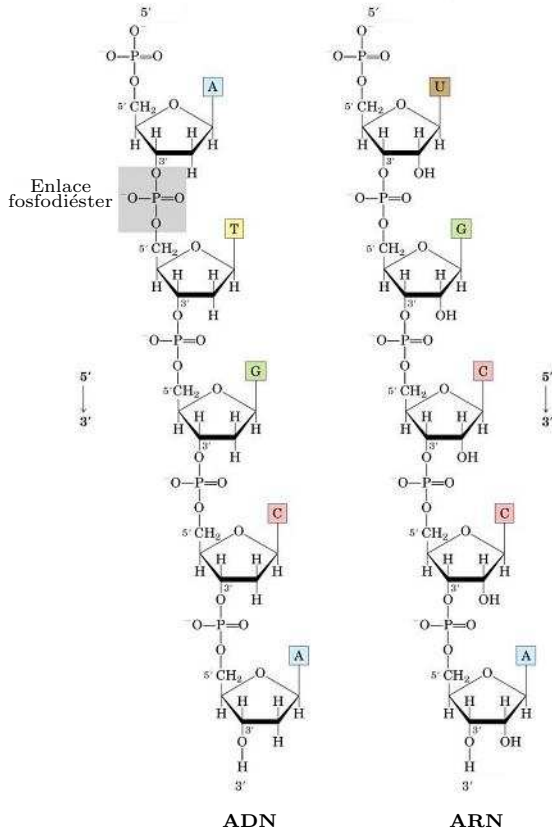


Figure 1.7: Polinucleótidos de ADN y ARN

está determinada por la secuencia de aminoácidos de su cadena .Esta capacidad de auto-ensamblarse de las proteínas es la responsable de su papel primordial en bioquímica. Las proteínas tienen muchas funciones –ser catalizadores de reacciones (enzimas), mantener estructuras celulares, generar movimientos, traducir señales, etc.– y cada una cumple una función específica según su secuencia de aminoácidos, determinada genéticamente.

Un mismo fragmento de la secuencia del ADN se puede usar varias veces para guiar la síntesis de muchos transcritos de ARN idénticos. Así, mientras que el archivo de información de la célula es fijo –el ADN–, los transcritos de ARN se producen en gran número y son desechables. La función de la mayoría de estos transcritos es servir de intermediarios en la transferencia de la información genética, actuando como un **ARN mensajero** (ARNm) que dirige la síntesis

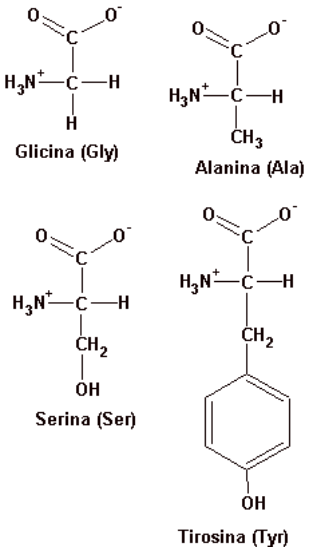


Figure 1.8: Estructura química de 4 de los 20 aminoácidos que componen las proteínas

de proteínas según las instrucciones almacenadas en el ADN.

La información contenida en la secuencia de ARNm se lee en grupos de tres nucleótidos; cada triplete de nucleótidos o *codón* especifica (codifica) un aminoácido de una proteína. Debido a que hay 64 posibles codones, pero sólo veinte aminoácidos, necesariamente hay muchos casos en los que varios codones corresponden a un mismo aminoácido. El código se lee por una clase especial de pequeñas moléculas de ARN, el **ARN de transferencia** (ARNt). Cada tipo de ARNt une en uno de sus extremos un aminoácido y tiene una secuencia específica de tres nucleótidos en su otro extremo –un *anticodón*– que le permite reconocer un codón o subgrupo de codones del ARNm por emparejamiento de bases.

Para la síntesis de proteínas, un conjunto de moléculas de ARNt cargadas con sus aminoácidos respectivos se une a un ARNm por emparejamiento de sus anticodones con cada uno de los codones sucesivos del ARNm. Después, los aminoácidos se van uniendo de forma que la proteína naciente va creciendo y cada ARNt, relegado de su carga, se libera.

Las moléculas de ADN son muy largas y contienen la especificación de miles

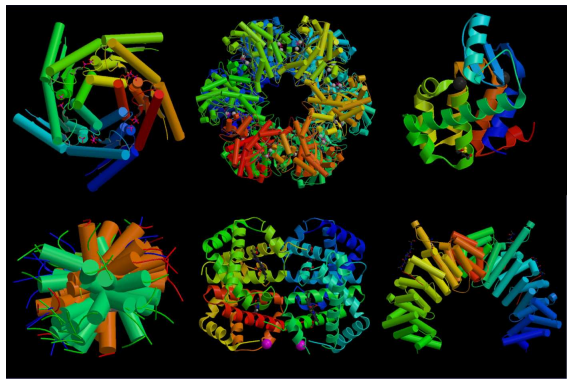


Figure 1.9: Algunas estructuras tridimensionales de proteínas

de proteínas. Por tanto, fragmentos de esta secuencia completa de ADN se transcriben en diferentes moléculas de ARNm, cada uno de los cuales codifica una proteína diferente. Un **gen** se define como un fragmento de la secuencia de ADN que corresponde a una sola proteína (o a una molécula de ARN catalítica o estructural, para los genes que producen ARN pero no proteína).

En todas las células, la expresión de determinados genes está regulada: en lugar de sintetizar el catálogo completo de posibles proteínas en todo momento, la célula ajusta la velocidad de transcripción y de traducción de diferentes genes de forma independiente y de acuerdo con sus necesidades. En el ADN celular existen secuencias de ADN no codificantes –denominadas *ADN regulador*– que están distribuidas entre las regiones codificantes de proteínas, y estas regiones no codificantes se unen a proteínas especiales que controlan la velocidad local de transcripción. Existen también otras regiones no codificantes, algunas de las cuales actúan como elementos de puntuación, indicando el inicio y el final de la información de una proteína. La región del ADN donde se establece cómo y cuándo se expresará el gen que se codifica en la región codificante inmediatamente adyacente se conoce como *región promotora*. En este sentido, el **genoma** de una célula –la totalidad de la información genética incluida en su secuencia completa de ADN– dicta no sólo la naturaleza de las proteínas celulares, sino también cuándo y dónde se sintetizarán.

2 ADN y Evolución

El material básico de la evolución es la secuencia de ADN que ya existe. No hay ningún mecanismo natural por el que se generen grandes cadenas de ADN de secuencia nueva aleatoria. Así, ningún ADN es completamente nuevo.Tanto durante el almacenamiento como durante el copiado del material genético se pueden producir accidentes y/o errores aleatorios que pueden alterar la secuencia de nucleótidos –es decir, generar **mutaciones**–. Como consecuencia de ello, cuando una célula se divide, a menudo sus dos células hijas no son idénticas entre sí o a su progenitora. Algunas veces poco frecuentes, el error puede representar un cambio favorable; más probablemente, el error no supondrá diferencias importantes en las capacidades de la célula; y en muchos casos, el error causará daños importantes –por ejemplo, alterando la secuencia de una proteína clave–. Cambios debidos a errores del segundo tipo pueden ser o no perpetuados, dependiendo de si la célula o sus familiares tienen o no éxito en la competencia por los recursos limitados del ambiente donde viven. Los cambios que causan daños importantes no conducen a la célula a ninguna parte, por lo general provocan su muerte, y por tanto, no dejan descendencia. Mediante la repetición de este ciclo de ensayo y error –de *mutación* y *selección natural*– los organismos van evolucionando y sus especificaciones genéticas van cambiando, proporcionándoles nuevas vías de aprovechamiento del entorno más eficaces para poder sobrevivir en competencia con otros organismos, reproduciéndose con más éxito. Las variaciones en fragmentos de ADN pueden ser generadas por varios métodos: (Stryer et al., 2003)

- *Mutación intragénica:* un gen ya existente puede ser modificado por mutaciones en su secuencia de ADN.
- *Mezcla de fragmentos:* dos o más genes existentes pueden romperse y reagruparse generando un gen híbrido formado por segmentos de ADN que originariamente pertenecían a genes independientes.
- *Transferencia horizontal:* un fragmento de ADN puede ser transferido desde el genoma de una célula al de otra célula, incluso de una especie diferente, si ambos organismos comparten el mismo ambiente. Este proceso contrasta con la *transferencia vertical* de información genética, habitual entre los progenitores y la progenie.

Una célula ha de duplicar todo su genoma cada vez que se divide en dos células hijas. Sin embargo, algunos accidentes pueden causar la duplicación de una parte del genoma, manteniendo el genoma original. Cuando un gen se ha duplicado por esta vía, una de las dos copias queda libre para mutar y especializarse

en la realización de una función diferente en la misma célula. Repetidos ciclos de este proceso de duplicación y divergencia, durante millones de años, han permitido que algunos genes generen una familia completa de genes en un mismo genoma.Cuando los genes se duplican y divergen de esta manera, los individuos de una especie resultan dotados de diferentes variantes del gen inicial. Este proceso evolutivo ha de distinguirse de la divergencia genética que ocurre cuando una especie se separa en dos líneas de descendencia diferentes en una bifurcación del árbol de la vida. En este punto, los genes se vuelven diferentes en el curso de la evolución, pero continúan teniendo funciones correspondientes en las dos especies hermanas. A los genes que están relacionados de esta forma –es decir, genes de dos especies separadas que derivan de un mismo gen ancestral presente en el último ancestro común de ambas especies– se los denomina **ortólogos**. A los genes relacionados que derivan de una duplicación en el mismo genoma –y que posiblemente divergirán en sus funciones– se los denomina **parálogos** (son parálogos, por ejemplo los genes que determinan las distintas clases de hemoglobinas que se producen a lo largo de la vida fetal y adulta). A los genes que están relacionados por una descendencia de cualquier tipo se los denomina **homólogos**, un término general que se utiliza para englobar ambos tipos de relación.

Cabe destacar que los intercambios horizontales de la información genética juegan un papel muy importante en la evolución bacteriana en el mundo actual. La reproducción sexual genera una transferencia horizontal de información genética a gran escala entre dos linajes celulares inicialmente separados –los de los progenitores–. Independientemente de si esto ocurre entre especies o dentro de una misma especie, la transferencia horizontal de genes deja una huella característica: genera individuos que están más relacionados entre sí con un grupo de parientes con respecto a determinados genes y con otros con respecto a otro grupo de genes.

3 Biología y Avances Tecnológicos

Hasta principios de los años setenta, el ADN era la molécula de la célula que planteaba más dificultades para su análisis bioquímico. Actualmente, el ADN ha pasado a ser la macromolécula más estudiada. Ahora podemos separar una región determinada del ADN, obtener un número de copias casi ilimitado y determinar su secuencia de nucleótidos.

Estos adelantos técnicos en la ingeniería genética han tenido un impacto espectacular en la biología celular, permitiendo el estudio de las células y de sus macromoléculas mediante sistemas que antes eran inimaginables. La tecnología

del ADN recombinante constituye un conjunto variado de técnicas, algunas de las cuales son nuevas y otras han sido adoptadas de otros campos de la ciencia, como la genética microbiana. Las más importantes son:

- La rotura específica del ADN mediante nucleasas de restricción, que facilita enormemente el aislamiento y la manipulación de los genes.
- La clonación del ADN, con el uso de vectores de clonación o de la reacción en cadena de la polimerasa, de tal forma que una molécula sencilla de ADN puede ser reproducida generando muchos miles de millones de copias idénticas (Fig. 1.10).
- La hibridación de los ácidos nucleicos, que hace posible localizar secuencias determinadas de ADN o de ARN con una gran exactitud y sensibilidad, utilizando la capacidad que tienen estas moléculas de unirse a secuencias complementarias.
- La secuenciación rápida de todos los nucleótidos de un fragmento purificado de ADN, que hace posible identificar genes y deducir la secuencia de aminoácidos de las proteínas que codifican.
- El seguimiento simultáneo del nivel de expresión de cada uno de los genes de una célula, utilizando microchips de ADN (microarrays) que permiten efectuar simultáneamente decenas de miles de reacciones de hibridación.

A continuación describiremos en más detalle este último ítem, que un elemento fundamental en el desarrollo de esta tesis.

3.1 Microarrays de ADN

Las técnicas clásicas para el análisis de secuencias permiten examinar la expresión de un número muy limitado de genes simultáneamente. Los *microarrays*, desarrollados en los años noventa, han revolucionado la forma en la que actualmente se estudia la expresión génica, al permitir el estudio de los productos de ARN de miles de genes a la vez. Esto ha permitido la identificación y el estudio de los patrones de expresión génica que subyacen a la fisiología celular: podemos ver qué genes se encuentran activados (o reprimidos) bajo distintas condiciones o ante la presencia de agentes externos.

Un microarray (Fig. 2) o biochip es una colección de pequeños fragmentos de genes unidos a la superficie de pequeños cristales, o dicho con otras palabras, es

Figure 1.10: Portada de la revista Times dedicada a la clonación de la oveja Dolly, primer clon de mamífero obtenido a partir de una célula de animal adulto.

un dispositivo de pequeño tamaño que tiene inmovilizado material biológico, que permite la automatización simultánea de miles de ensayos encaminados a conocer en profundidad la estructura y funcionamiento de nuestra dotación genética. En ellos se integran decenas de miles de fragmentos de material genético, de secuencia conocida y de diferente tamaño, ordenados sobre un sustrato sólido, de manera que forman una matriz de secuencias en dos dimensiones. Si las secuencias son cortas, se denominan microarrays de oligonucleótidos, si tienen mayor tamaño, chips de ADNc (ADN complementario, sintetizado a partir de ARNm). A los fragmentos inmovilizados en el soporte, se les denomina sondas. Los ácidos nucleicos de las muestras a analizar se pueden marcar por diversos métodos (enzimáticos, fluorescentes, etc.), incubándose posteriormente sobre la matriz de sondas, produciéndose una hibridación entre las secuencias homólogas, es decir, sólo las cadenas complementarias a las del chip se hibridan. Después de la hibridación entre las secuencias del microarray y la muestra marcada con fluorescencia, los chips son leídos en un escáner, originándose un patrón de luz característico y una cuantificación de la intensidad de hibridación de cada punto, los datos obtenidos son interpretados mediante un ordenador. Esto permite una identificación y cuantificación del ADN o ARN presente en la muestra, así como conocer la estructura y función de la dotación genética, tanto en los diferentes estados de desarrollo normal como patogénicos del paciente.

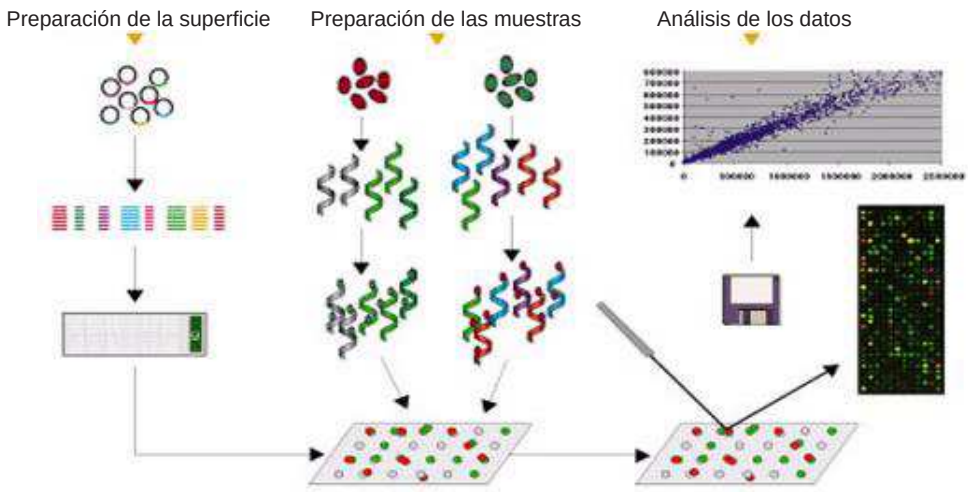


Figure 1.11: Proceso de creación de un microarray de ADN.

4 Biología computacional y Bioinformática

En las últimas décadas, los avances en la biología molecular y el equipamiento disponible para la investigación en este campo han permitido la rápida secuenciación de grandes porciones de genomas de diversas especies. En la actualidad, varios genomas de bacterias, tales como *Saccharomyces cerevisiae*, y algunos eucariotas simples ya han sido secuenciados por completo. El proyecto Genoma Humano (Collins et al., 2003), diseñado con el fin de secuenciar los 24 cromosomas del ser humano, también está progresando. Las bases de datos de secuencias más populares, como GenBank (Benson et al., 2007) y EMBL (Kanz et al., 2005), están creciendo de forma exponencial. Esta gran cantidad de información necesita de un alto nivel de organización, indexado y almacenamiento de las secuencias. Es por ello que la Informática ha sido aplicada a la Biología para producir un nuevo campo de investigación llamado *Bioinformática* que permita ayudar a esta organización (Attwood and Parry-Smith, 2002).

4.1 Objetivos de la Bioinformática

El término bioinformática ha sido adoptado por varias disciplinas diferentes . En su sentido más amplio, puede considerarse que el término significa tecnología de la información aplicada a la gestión y análisis de datos biológicos. Esto tiene

implicaciones en diversas áreas, desde la inteligencia artificial y la robótica al análisis de genomas. En el contexto de los proyectos genoma, el término se aplicó originalmente a la manipulación computacional y al análisis de datos de secuencias biológicas (ADN o proteínas). Sin embargo, a la vista de la rápida y reciente acumulación de estructuras de proteínas disponibles, el término ahora tiende a emplearse abarcando también la manipulación y análisis de datos de estructuras tridimensionales (3D).

Las tareas más simples de la Bioinformática conciernen la creación y mantenimiento de bases de datos de información biológica. Secuencias nucleotídicas (y las secuencias proteicas que derivan de las mismas) componen la mayoría de la información que está almacenada en estos repositorios. Mientras que el almacenamiento y organización de millones de nucleótidos está muy lejos de ser una tarea trivial, el diseño de una base de datos y el desarrollo de una interfaz con la cual los investigadores puedan tanto acceder a la información existente como agregar nuevas instancias, es simplemente el comienzo.

Tal vez, la tarea más apremiante sea la que involucra el análisis de la información de secuencias. *Biología Computacional* es el nombre dado a este proceso e incluye las siguientes tareas:

- Encontrar genes en secuencias de ADN pertenecientes a varios organismos.
- Desarrollar métodos para la predicción de la estructura y/o la función de nuevas proteínas y secuencias estructurales de ARN.
- Agrupar secuencias de proteínas en familias de secuencias relacionadas y el desarrollo de modelos de proteínas.
- Alinear proteínas similares y generar árboles filogenéticos para examinar las relaciones de la evolución.

El proceso de evolución ha producido secuencias de ADN que codifican proteínas con funciones muy específicas. Es posible predecir la estructura tridimensional de una proteína usando algoritmos derivados de nuestros conocimientos en el campo de la Física, la Química y, en mayor medida, del análisis de otras proteínas con secuencias de aminoácidos similares.

La mayoría de las bases de datos biológicas consisten en largas secuencias nucleotídicas y/o secuencias de aminoácidos. Cada secuencia representa un gen o proteína particular (o una sección de la misma), respectivamente. Mientras que la mayoría de las bases de datos biológicas contienen este tipo de información,

también existen otros repositorios que incluyen información taxonómica tales como características estructurales o bioquímicas de los organismos.

En las últimas tres décadas, las contribuciones al área de la Biología y de la Química han facilitado el aumento en la velocidad del proceso de secuenciación de genes y proteínas. El advenimiento de la tecnología de clonación ha permitido que secuencias de ADN foráneas sean introducidas en bacterias. De esta manera fue posible la rápida producción de secuencias de ADN particulares, un preludio necesario para la determinación de secuencias. La síntesis de oligonucleótidos dio a los investigadores la habilidad de construir pequeños fragmentos de ADN con secuencias elegidas por ellos mismos. Estos oligonucleótidos son luego utilizados como parte de bibliotecas de ADN y permiten la extracción de genes que contengan esta secuencia. Estos fragmentos de ADN también pueden ser utilizados en reacciones en cadena de polimerización para amplificar secuencias de ADN o modificar estas secuencias. Mediante estas técnicas, el progreso de la investigación biológica ha crecido exponencialmente.

Sin embargo, para que los investigadores puedan beneficiarse de esta información, es necesario cumplir con dos requisitos: (1) tener acceso inmediato al conjunto de secuencias coleccionadas y (2) tener una forma de extraer de este conjunto solamente aquellas secuencias que interesen al investigador. La simple colección, de forma manual, de toda la información necesaria para un proyecto dado a partir de un artículo de revista publicado puede convertirse rápidamente en una tarea epopéyica. Después de obtener los datos, es necesario organizarlos y analizarlos. La búsqueda manual de genes y proteínas relacionadas puede llevar semanas e incluso meses para un investigador.

La tecnología informática ha proporcionado la solución a este problema. Los ordenadores, no solo pueden acumular y organizar la información de secuencias en bases de datos, sino que también pueden analizar los datos de las secuencias muy rápidamente. La evolución del poder computacional y la capacidad de almacenamiento ha logrado lidiar con la creciente cantidad de información de secuencias que está siendo creada. Los científicos teóricos han desarrollado sofisticados algoritmos que permiten comparar secuencias mediante teoría de probabilidades. Estas comparaciones se han convertido en la base de la determinación de la función de genes, desarrollando relaciones filogenéticas y simulando modelos de proteínas.

La colección, organización e indexado de la información de secuencias en una base de datos es una tarea desafiante por sí misma y ha generado una gran cantidad de información pero de uso limitado. El poder de una base de datos no proviene de la colección de información que tenga, sino de su análisis. Una secuencia de ADN no necesariamente constituye un gen, puede constituir

solamente un fragmento de un gen o contener varios genes.

La investigación científica actual, de acuerdo con los principios de la evolución, muestra que todos los genes tienen elementos comunes. Para muchos elementos genéticos es posible construir secuencias consenso, las cuales representan de la mejor manera posible la norma de una clase dada de organismo. Algunos elementos genéticos comunes incluyen promotores, reforzadores, señales de poliadenización y sitios de binding de proteínas. Para estos elementos también se conocen algunas características de sus subelementos. Los elementos genéticos comunes comparten secuencias similares, siendo éste el hecho que permite la aplicación de algoritmos al análisis de secuencias biológicas.

5 Introduction to Microarray Technology

Advances in molecular biology and new computational techniques are enabling us to systematically investigate the complex molecular process underlying biological systems (Durbin et al., 1998). To take full advantage of the large and rapidly increasing body of sequence information, new technologies are required. Among the most powerful and versatile tools for genomics are high-density arrays of oligonucleotides or complementary DNAs. Also known as microarrays, they have revolutionized modern biological research by its capacity of monitoring the expression level of thousands of genes simultaneously (Brown and Botstein, 1999), while traditional methods could only handle the one-gene at a time approach.

A microarray works by exploiting the ability of a given mRNA molecule to bind specifically to, or hybridize to, the DNA template from which it originated. By using an array containing many DNA samples, scientists can determine, in a single experiment, the expression levels of hundreds or thousands of genes within a cell by measuring the amount of mRNA bound to each site on the array. With the aid of a computer, the amount of mRNA bound to the spots on the microarray is precisely measured, generating a profile of gene expression in the cell.

Microarrays are therefore useful for rapid surveying large number of genes or when the sample to be studied is small. Microarrays may be used to assay gene expression within a single sample or to compare gene expression in two different cell types or tissue samples, such as in healthy and diseased tissue.

DNA Microarrays are small, solid supports onto which the sequences from thousands of different genes are immobilized, or attached, at fixed locations.

The supports themselves are usually glass microscope slides, the size of two side-by-side pinky fingers, but can also be silicon chips or nylon membranes. The DNA is printed, spotted, or actually synthesized directly onto the support.

The whole microarray experiment process is based on hybridization probing, a technique that uses fluorescently labeled nucleic acid molecules as "mobile probes" to identify complementary molecules, sequences that are able to base-pair with one another. Each single-stranded DNA fragment is made up of four different nucleotides, adenine (A), thymine (T), guanine (G), and cytosine (C), that are linked end to end. Adenine is the complement of, or will always pair with, thymine, and guanine is the complement of cytosine. Therefore, the complementary sequence to G-T-C-C-T-A will be C-A-G-G-A-T. When two complementary sequences find each other, such as the immobilized target DNA and the mobile probe DNA, cDNA, or mRNA, they will lock together, or hybridize.

Two main types of DNA chips can be discerned, either oligonucleotides or complementary DNAs (cDNA). Both are based on the same principle, however the method of addition of the nucleotide stretches to the chip differs. We now briefly describe each of the technologies.

5.1 Spotted Arrays

In spotted microarrays (or two-channel or two-colour microarrays), the probes are cDNA or small fragments of PCR products that correspond to mRNAs (messenger RNA) and are spotted onto the microarray surface. This type of array is typically hybridized with cDNA (complementary DNA) from two samples to be compared (e.g. diseased tissue versus healthy tissue) that are labeled with two different fluorophores (e.g. Rhodamine (Cyanine 5, red) and Fluorescein (Cyanine 3, green)). The two samples are mixed and hybridized to a single microarray that is then scanned in a microarray scanner to visualize fluorescence of the two fluorophores (see Fig.1.12). Relative intensities of each fluorophore are then used to identify up-regulated and down-regulated genes in ratio-based analysis. Absolute levels of gene expression cannot be determined in the two-colour array, but relative differences in expression among different spots (=genes) can be estimated with some oligonucleotide arrays. Examples of providers for such microarrays includes Agilent with their Dual-Mode platform, Eppendorf (company) with their DualChip platform and ArrayIt.

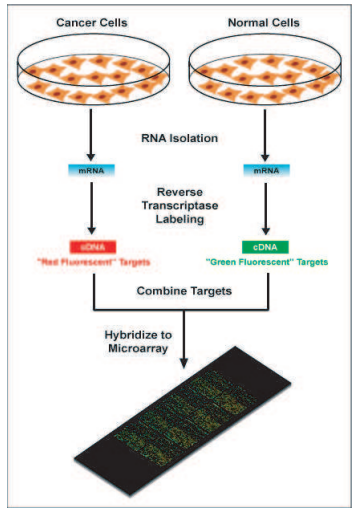


Figure 1.12: Diagram of typical dual color microarray experiment.

5.2 Oligonucleotide Arrays

In oligonucleotide microarrays (or single-channel microarrays), the probes are designed to match parts of the sequence of known or predicted mRNAs. There are commercially available designs that cover complete genomes from companies such as GE Healthcare, Affymetrix, Ocimum Biosolutions, or Agilent. These microarrays give estimations of the absolute value of gene expression and therefore the comparison of two conditions requires the use of two separate microarrays (see Fig.1.13).

In this type of arrays, the oligonucleotides are synthesised directly onto the chip. The solid surface is prepared such, that there are 3'-OH ends sticking out, to which nucleic acids can be attached in sequence. These are the arrays used for the main experiment related to this work, in particular the Affymetrix GeneChip® HG133A, comprised of more than 22,000 probe sets and 500,000 distinct oligonucleotide features including 14,500 well characterized human genes. Probe sets are the "basic unit" that Affymetrix uses for its array (see Fig.1.14). Each *probe set* is made up of a number of *probe pairs*, between 11 and 20. Each of this probe pairs is made of two positions, a "Perfect Match" (PM) and a "Miss Match" (MM) which are complementary. The PM is made out of 25 oligonucleotides, designed to be perfectly compatible with the RNA sequence. The MM is made out of 25 oligonucleotides compatible with the RNA sequence it

Figure 1.13: Affymetrix Chips

hybridizes, except in its central position, number 13, which serves as a control for the specific hybridation, since MM hybridation should always be less than PM hybridation. Therefore, some probe sets are compatible with intragenic regions, and thus the probe set will be associated to an specific gene, some other are compatible with intergenic region etc. Note than two or more probe sets might correspond to different regions of the same gene.

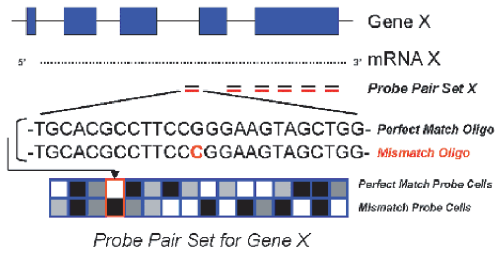


Figure 1.14: Probe Set structure in Affymetrix GeneChips®.

Affymetrix provides for each chip several files associated to it in order to process the information acquired. The raw image data (see Fig.1.15) from chip scanner is saved in .DAT file. The information about the expression leves of individual probe sets is extracted from the image data, .DATA file, and stored in a .CEL file. The probe set information in the .CEL file by itself is not

particularly useful as there is no indication in the file as to which probe set a probe belongs. This information is stored in the .CDF library file associated with a GeneChip® type. The Affymetrix probe set IDs are not particularly descriptive (e.g. 200008_s_at, 200015_x_at or 200035_at). The mapping between the IDs and the gene names is stored in the .GIN file. Affymetrix also provides for each particular GeneChip® an annotation file, for use by any interested party to understand what biological entities are represented on Affymetrix arrays. In such files probe sets are related to its sequence source, UniGene ID, Gene title and symbol, RefSeq protein ID, SwissProt entry, Gene Ontology Association, Pathway information and much more information. The .CHP file contains the results of the experiment. These include the average signal measures for each probe set as determined by the Affymetrix software and information about which probe sets are called as present, absent or marginal and the p-values for these calls.

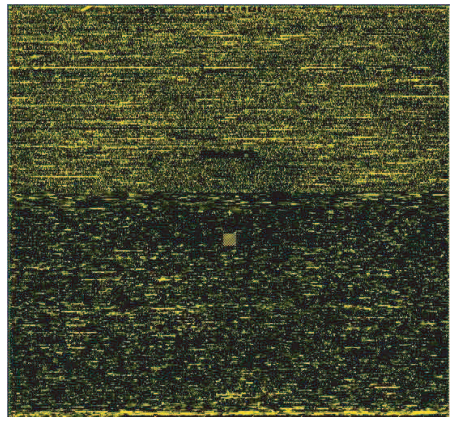


Figure 1.15: Scanned image of an Affymetrix array.

5.3 Microarray Scope of Application

One of the most important applications for arrays so far is the monitoring of gene expression (mRNA abundance). The collection of genes that are expressed or transcribed from genomic DNA, sometimes referred to as the expression profile or the "transcriptome", is a major determinant of cellular phenotype and function. The transcription of genomic DNA to produce mRNA (messenger RNA) is the first step in the process of protein synthesis, and differences in gene expression are responsible for both morphological and phenotypic differences as well as indicative of cellular responses to environmental stimuli and

perturbations. Unlike the genome, the transcriptome is highly dynamic and changes rapidly and dramatically in response to perturbations or even during normal cellular events (Lockhart and Winzeler, 2000) such as DNA replication and cell division (Cho et al., ; Spellman et al., 1998). In terms of understanding the function of genes, knowing *when, how and for how long a gene is turned on/off* is central to understanding the activity and biological roles of its encoded protein. In addition, changes in the multi-gene patterns of expression can provide clues about regulatory mechanisms and broader cellular functions and biochemical pathways. In the context of human health and treatment, the knowledge gained from these types of measurements can help determine the causes and consequences of disease, how drugs and drug candidates work in cells and organisms, and what gene products might have therapeutic uses themselves or may be appropriate targets for therapeutic intervention.

Gene expression profiles as "fingerprints". An often overlooked aspect of measurements of global gene expression is that the sequence or even the origin of the arrayed probes does not need to be known to make interesting observations - the complex profiles, consisting of thousands of individual observations, can serve as transcriptional "fingerprints". These fingerprints are extremely interesting to be known. They can be used, for instance, for classification purposes or as tests for relatedness, in a similar manner to the way in which DNA fingerprints are used in paternity testing. Many papers have been published, where these "fingerprints" are used as classification features for different phenotypes, specially in cancer classification: (Alizadeh et al., 200; Ben-Dor et al., 2000).

Transcriptional fingerprints have been also used to determine the target of an specific drug (Marton et al., 1998). The basic idea is that if a drug interacts with and inactivates a specific cellular protein, the phenotype of the drug-treated cell should be very similar to the phenotype of a cell in which the gene encoding the protein has been genetically inactivated, usually through mutation. Thus, by comparing the expression profile of a drug-treated cell to the profiles of cells in which single genes have been individually inactivated, specific mutants can be matched to specific drugs, and therefore, targets to drugs (Lockhart and Winzeler, 2000).

Finally, expression profiles can be used to classify drugs and their mode of action. For example, the functional similarity and specificity of different purine analogues have been determined by comparing the genome-wide effects on treated yeast, murine and human cells (Rosania et al., 2000).

Chapter 2

Microarray Analysis State of the Art

Microarray technology has revolutionized modern biological research by its capacity of monitoring the expression level of thousands of genes simultaneously (Brown and Botstein, 1999), while traditional methods can only handle the one-gene at a time approach. The development and application of microarray technology has risen many problems that need to be addressed by the collective knowledge and skills of the mathematical, physicists, computer and biological scientists. The greatest challenge in the microarray technology development is the analysis of the data. The current bottleneck in the processing of microarray data occurs after the data are generated; the magnitude of the problem is proving to be on a par with developing the technology itself.

Microarray experiments present a wealth of steps (see Fig.2.1): first, the design of the experiments. Here, the researchers have to decide which genes are to be printed on the arrays, which sources are to be hybridized and on how many arrays the hybridizations will be replicated. Second, the image scanning process to extract the information contained in each of the microarray probes. Third, a number of low-level analysis of the microarray data to account for problems such as microarrays experiments carried out in different conditions (e.g. different days, different labs, different scanning intensities), which can cause microarray expression levels to run on different ranks. Therefore, microarrays

will be on a not not comparable scale. Moreover, this step accounts for problems related to variation other than that due to the differences between the RNA samples being studied, such as dye-bias or systematic/random variation (Nadon and Shoemaker, 2002). The methods applied to solve these problems are scaling and normalization. Fourth, once the microarray data are comparable and all sources of variation other than the experimentally related are removed, high-level analysis can be performed to identify genes behaving in a different way between experimental conditions. This analysis include both gene filtering, which excludes genes from any further analysis based on some criterion, (such as the level of hybridization throughout the arrays, and identification of differentially expressed genes,by means of applying statistical tests to compare the experimental conditions.

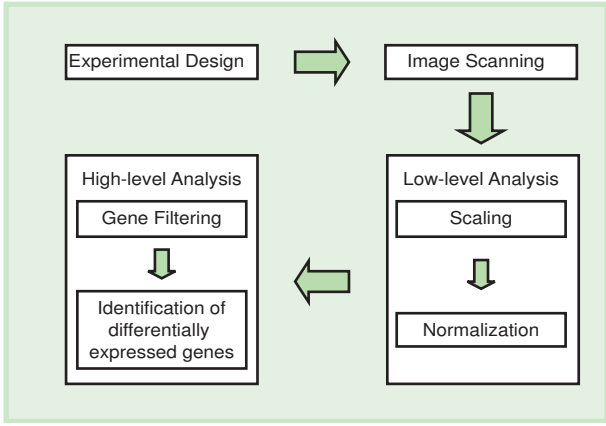


Figure 2.1: Schema of microarray experiments design and analysis process.

The third and fourth steps, low and high-level analysis of microarray data, involve detailed statistical analysis outside of the pool of well established routine statistical procedures. For example, a microarray experiment provides a set of measurements containing several thousand numbers, one for each probe on the array. Methods based on conventional p-values provide a probability that a difference in gene expression occurred by chance (Galitski et al., 1999). Although $p = 0.01$ is significant in the context of experiments designed to evaluate small numbers of genes, a microarray experiment for 10,000 genes would identify 100 genes by chance, being this number unacceptable. As a consequence, new statistical methods tailored to microarrays continue to be developed and adapted. The use of microarrays would not be longer of interest without a proper methodology to handle this new challenges microarray technology has risen.

In this work we study a problem derived from longitudinal blood expression profiles of human volunteers for the study of the inflammation and the host response to injury, as part of a Large-Scale Collaborative Research Project sponsored by the National Institute of General Medical Sciences (www.gluegrant.org) (Calvano et al., 2005) in a particular microarray experiment carried out at the Washington University School of Medicine in St. Louis, Missouri, in collaboration with the Cellular Injury and Adaptation Laboratory. Analysis of the set of gene expression profiles obtained from this experiment is complex, given the number of samples taken and variance due to treatment, time, and subject phenotype. Therefore, we believe this problem is typical and informative as a RNA microarray case study.

Throughout this chapter we will describe in detail the problem under study (Section 1), the experimental design and image scanning (Section 2)the microarray data analysis process (Section 3), both low-level analysis (Section 3.1) and high-level analysis (Section 3.2), highlighting the new challenges present at each of the steps and the most widely applied methods and we will also show that the application of these methods is not as successful as expected in the inflammation problem (Section 2).

1 Problem Description: Inflammation and the host response to injury

Inflammation is a hallmark of many human diseases (Coussens and Werb, 2002). Understanding the inflammation process is critical because the body uses inflammation to protect itself from infection or injury (e.g., crushes, massive bleeding, or a serious burn) which, in extreme cases (e.g., car accidents or gun shootings), can lead to massive organ malfunction and death. According to the U.S. Centers for Disease Control's National Center for Health Statistics, unintentional injury is the leading cause of death for people ages 1 to 35 (Calvano et al., 2005). The host response to trauma and burns is a collection of biological and pathological processes that depends critically upon the regulation of the human immuno-inflammatory response.

The problem under study is focussed on blood leukocytes and other tissues of critically injured patients, in order to better elucidate the mechanisms underlying systemic inflammatory responses (Bone et al., 1992). This approach cannot be fully replicated using animal models or human cell lines, and studies of injury in humans can be complicated by antecedent illnesses and concurrent treatment regimes that may alter the recovery process. This has been the first

study to evaluate the genome-wide response to systemic inflammation in the context of a fully predictable recovery.

No single research center or small group of centers has the resources to delineate the integrated response of this complete biological system, which involves multiple molecular and genetic interactions that vary in time. This study, in part carried out at the Cellular Injury and Adaptation Laboratory, Washington University School of Medicine, is a piece of a Large-scale Collaborative Research Project sponsored by the National Institute of General Medical Sciences (www.gluegrant.org), devoted to profile leukocyte gene expression and plasma proteins of burn and trauma patients (Calvano et al., 2005). Prior to initiating studies in actual patients, it was proposed that the human endotoxin model could serve as a starting point and test bed for subsequent studies. Our proposal will help to promote the identification of significant relationships, which regulate the integration of this complex biological system, with the expectation that this understanding will ultimately impact the treatment of hospitalized patients.

Besides the importance of the biological problem under study, analysis of the set of gene expression profiles obtained from this experiment is complex, given the number of samples taken and variance due to treatment, time, and subject phenotype. Therefore, we believe this problem is typical and informative as a RNA microarray case study.

2 Experimental Design and Image Scanning

Two patient groups and one category of human volunteers were selected to study: burn injury, severe trauma, and low-dose lipopolysaccharide- (LPS) challenged normal volunteers because it is believed that there is an enormous need to discern which responses are common from those that are specific for each condition. The LPS-challenge causes an acute, discrete inflammatory process, which can be contrasted with the acute inflammatory process that is produced by injury. We anticipate that there will be common as well as contrasting features to the proteomic and genomic data that derive from our studies suggesting molecular mechanisms that may be involved in each group as well as within the phenotypes identified in each group. We propose 4 clinical trajectories, which are described below, for patients suffering from traumatic and burn injury and we anticipate that each trajectory will satisfactorily describe at least one phenotype that can be discerned from others.

To develop a rational approach for the investigation of these injured patient

populations, a consensus was reached among the participants that an unproven but testable paradigm describes the potential 4 trajectories for patients who have sustained injury.

(1) The first trajectory (Early Death) is a failure of resuscitation and early death within 24 hours of severe early MODS (multiple organ dysfunction syndrome). This is an unfortunate path, and our ability to study these patients is limited.

(2) The second trajectory (Mild MODS) is one in which the patient is successfully resuscitated, the reason for the post-traumatic shock is corrected or the burn injury is addressed appropriately, and the patient recovers from early, mild MODS. These patients are discharged uneventfully within the initial weeks after the injury and when they are seen later in the office, they are normal from a physiological perspective.

(3) The third trajectory (Severe MODS), is one in which the patient demonstrates a more severe, immediate immuno-inflammatory response and early, severe MODS. A variant of this trajectory is the "early 2-event model" in which the patient's immune system, particularly the neutrophils, is primed and an early second insult (e.g. recurrent hemorrhage, pulmonary aspiration, abdominal compartment syndrome, or intramedulary rod fixation) occurs during this vulnerable window. This second event provokes unbridled systemic inflammation which culminates in early, severe MODS. The MODS in these patients may resolve and the patient may have a prolonged recovery with a gradual return to normal physiology. Otherwise, these patients suffer from overwhelming MODS and expire.

(4) The fourth trajectory (2-hit MODS) is one in which the patient survives the early immuno-inflammatory insult, but receives a later, "second hit" which includes a nosocomial infection, endotoxemia, or the persistence of devitalized tissue. These patients have a reprogrammed genome as a result of their initial injury and these patients respond to this second hit or injury with an exaggerated immuno-inflammatory response. They frequently develop late, severe MODS and have a higher mortality risk. Those who survive are eventually discharged to their homes, acute rehabilitation facilities, or other long-term care facilities after many months of complicated hospitalization. These latter patients infrequently return to a normal physiome, and likely never return to normal gene expression patterns.

We propose to study injured patients in each of the 4 trajectories from a physiological, proteomic, and genomic perspective. The entry criteria for the severe trauma patients will be 16 years of age, an injury severity score (ISS) >

15, and documented presence of shock; patients with severe head injury will be excluded. The entry criteria for burn injured patients will be 16 years of age and a burn injury total body surface area (TBSA) of 40-80from any etiology. The characteristic hospital length of stay (LOS) for patients in the Mild MODS or 2nd trajectory will be 1 - 2 weeks (LOS for burned patients in this trajectory will likely exceed 2 weeks) and for the latter two trajectories, hospital length of stay for both severe trauma and burn injury is likely to exceed 28 days with most of their in-hospital stay in the intensive care unit. Blood samples will be obtained at intervals beginning upon admission until 28 days post-injury or at the time of hospital discharge. Additional tissue samples including bronchoalveolar lavage and resected surgical specimens will be obtained whenever feasible and stored for possible future usage in a tissue bank. Selected pro-inflammatory and anti-inflammatory cytokines, as well as selected other protein and cellular features that characterize the acute phase and immuno-inflammatory status of the patient will be evaluated in the PACB Core.

Intravenous endotoxin challenge in normal volunteers is a well-characterized stimulus that reproducibly induces flu-like symptoms that resolve by 24 hours (Richardson et al., 1989). These symptoms are associated in the model with significant changes in circulating leukocyte gene expression profiles (Calvano et al., 2005). Notably, there is an initial proinflammatory phase and a subsequent counterregulatory phase, with resolution of virtually all clinical perturbations within 24 h. Gene expression in whole blood leukocytes was determined using 48 GeneChips® HG-U133A v2.0 from Affymetrix Inc., derived from samples taken from human blood of eight patients: four treated with intravenous endotoxin (i.e., patients 1 to 4) and four with a placebo (i.e., patients 5 to 8), and expression retrieved over time immediately before and at 2, 4, 6, 9 and 24 hours after the intravenous administration of bacterial endotoxin.

Six healthy male and female subjects between 18 and 40 years of age (1 female, 5 males) provided written informed consent. Subjects were intravenously administered the same dose of endotoxin. Arterial blood samples were collected before endotoxin infusion (0 hours) and at post infusion times of 2, and 6 hours. Same protocols were used in endotoxin administration, blood sampling, and leukocyte RNA isolation, as summarized in the Methods of the manuscript. cRNA synthesis and Chip hybridization. cRNA synthesis was performed with 4 ìg of total cellular RNA, and hybridized onto the Human Genome U133 Plus 2.0 Array (Affymetrix) and processed based on an updated protocol1 outlined by Affymetrix Inc. (Santa Clara, CA). Microarray data analysis. 54,675 probe sets on the U133 Plus 2.0 Array were analyzed, including complete coverage of the HU133A and HU133B set plus 6,500 additional genes. Because of the platform and protocol differences between the U133 Plus 2.0 Array and the U133 (A, B) Set, it is not feasible to directly compare the signal level of a gene in the

verification experiment (U133 Plus 2.0) with that in the initial study (U133 Set) 2. Besides, the initial time course study was conducted at 6 time points (0, 2, 4, 6, 9 and 24 hours), among which 3 time points (0, 2, and 6 hours) were selected in the verification study.

- **Type of experiment.** Gene expression profiling of human blood leukocytes and skeletal muscle from healthy human subjects, and hospitalized patients following severe traumatic or burn injury.

- **Experimental factors.** Blood was obtained from either healthy male and female subjects, or from critically ill patients following severe traumatic or burn injury. Waste skeletal muscle tissues were obtained from severely burned patients at time of surgical resection, for clinical management. Blood leukocytes were isolated and muscle or leukocyte RNA were analyzed using Affymetrix $GeneChip^{TM}$ arrays.

GeneChip® arrays were scanned using the HP GeneArray Scanner (Affymetrix). The .cel files were generated using the Microarray Suite v5 software from Affymetrix

3 Microarray Data Analysis

The greatest challenge in microarray technology development is analytical. The current bottleneck in the processing of microarray data occurs after the data are generated. We focuss on this bottleneck: the methods applied for the low and high-level analysis of microarray data. We describe the conventionally applied methods at these steps. These methods which have also been applied by us to the inflammation and host response to injury problem.

3.1 Low-level analysis

Microarray data needs to undergo a number of low-level analysis (see Fig. 2.1) to account for the problems their analysis poses such as microarrays experiments being carried out in different conditions or presence of expression level variation other than that due to the differences between the RNA samples being studied, such as dye-bias or any systematic or random variation (Nadon and Shoemaker, 2002).

The first task to perform is *scaling* of the microarray data. Scanned images may have different overall brightness. This might be due to several possible factors: experiments being carried out on different days, different labs, by different technicians, using different scanners for extracting the intensities and a long etcetera. In order to make comparisons among microarrays, we need them to have their level of intensities on a comparable level. We use the scaling method proposed by Li and Wong (2001b), termed the *Invariant Set* Approach. For a group of microarrays, we scale all arrays (except from one acting as the baseline microarray) so that they all end up having the same median overall brightness without losing the true expression level variation due to RNA changes in the experiment.

The scaling should be based only on probes which are not differentially expressed throughout the experiment, but at this step of the analysis process we do not know which are such probe sets. Nevertheless, we expect that a probe of a non-differentially expressed gene in two microarrays to have similar intensity ranks. An iterative process is applied to identify a set of probes which presumably is non-differentially expressed (the so called "invariant set"). The scaling curve is the running median curve in the scatterplot of probe intensities of the two arrays (the baseline array on the Y-axis and the array to be scaled on the X-axis). When fitting the running median curve at the two ends, 5% of the "invariant" points are used to fit at one end fixed. This makes the high-end normalization relationship more smooth and robust. The final running median curve is a piece-wise linear curve. After performing this process both arrays have the same median overall brightness. In Fig. 2.2 we can see the change in microarray brightness after scaling both of them to a common array.

Once the data are in a comparable scale, we need to account for problems related to variation other than that due to the differences between the RNA samples being studied, such as dye-bias or any systematic or random variation (Nadon and Shoemaker, 2002). We apply a *normalization* process. Normalization deals with the two types of measurement error: random and systematic (Nadon and Shoemaker, 2002). The *random error* is a measure of uncertainty in the measurement and is therefore central to statistical inference. Random errors are not "mistakes" in the colloquial sense. Rather, they reflect inevitable uncertainties in all scientific measurements, making statistical procedures necessary. For example, consider the case of a probe that is not differentially expressed. Because of random measurement error, its measured differential expression ratio will deviate from its true value of 1:1. Deviations from this 1:1 ratio are due to "chance". Random error cannot be eliminated, but instead is estimated from observed data. Random error is minimized by controlling extraneous factors and by obtaining more repeated measurements (replicates). *Systematic errors* are biases; they result in a constant tendency to over or underestimate true

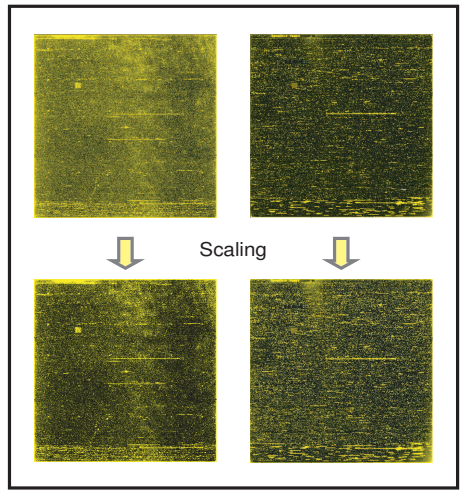


Figure 2.2: Changes in the microarray brightness after scaling.

values, thereby decreasing accuracy. Biasing factors come in many forms and are partially dependent on spotting, scanning and labeling technologies. Bias can affect all expression values on an array equally or depend on other aspects (e.g. spatial location, spotting pins, signal intensity). Sources of bias are in theory identifiable by quality control studies. However, biasing effects from various sources can be nonorthogonal and they are often nonlinear. This fact, along with the typically few replicates available for estimation, complicate quantifying the specific sources of bias. Systematic errors (bias) are controlled experimentally as far as possible, although additional statistical correction is invariably necessary with current microarray technology.

We have applied the *Model-Based Expression Index* (MBEI) approach, which is a multiplicative model proposed by Li and Wong (2001a) that identifies and isolates biological variability from systematic and random errors. We will describe the model particularly in relation to the Affymetrix microarrays used, with the *PM/MM* structure described in Chapter 1. Following this approach, a probe set in the Affymetrix oligonucleotide microarray has the form

$$y_{ij} = PM_{ij} - MM_{ij} = \theta_i \phi_j + \varepsilon_{ij}, \sum_j \phi_j^2 = J, \varepsilon_{ij} \sim N(0, \sigma^2) \qquad (2.1)$$

This equation states the the perfect match (PM)/mismatch(MM) difference

in array i, probe j, of this probe set is the product of model-based expression index (MBEI) in array $i\,(\theta_i)$ and probe-sensitivity index of probe $j\,(\phi_j)$ plus random error ε_{ij}. Here J is the number of probe pairs in the probe set. Fitting the model, we can identify cross-hybridizing probes (ϕ_j) with large standard error (SE), which are excluded during iterative fitting and arrays with image contamination at this probe set (θ_i) with large (SE)) as well as single outliers which are replaced by the fitted values. In effect, the estimated expression index θ_i is a weighted average of PM/MM differences:

$$\widetilde{\theta_i} = \frac{(\sum_j y_{ij}\phi_j)}{J} \tag{2.2}$$

with larger weights given to probes with larger ϕ. The image of outliers (array or probe sets) identified through model-fitting can be used to assess the quality of an experiment.

This approach performs the calculations over all microarrays together, differing from methods such as MAS v.5 (Affymetrix Inc), where the calculations are performed independently for each microarray. Normalizing the microarrays all at once makes them remain comparable for further analysis.

3.2 High-level analysis

The objective when carrying out microarray experiments essentially is identifying genes which behave in a different way between experimental conditions. The high-level analysis (see Fig.2.1) is performed for this gene selection. It includes both gene filtering and identification of differentially expressed genes applying statistical tests.

We are interested in genes showing large variation across samples or present in most samples. Therefore, we filter out probe sets which do not satisfy having a ratio of the standard deviation and the mean across all samples to be greater than a certain threshold. This ratio is also known as Coefficient of Variation (CV). The more variable a probe set is across samples, the larger the ratio is. We have used the default upper limit of 1000, which is a reasonably large number that is usually satisfied.

Once the filtering has been performed, we identify differentially expressed genes. Statistical considerations are frequently to the fore in this type of analysis of microarray data, as researchers sift through massive amounts of data and

adjust for various sources of variability in order to identify the important genes amongst the many which are measured. To address the statistical challenge of analyzing such large data sets, new methods have emerged (Inza et al., 2004; Li and Wong, 2003; Pan et al., 2001; Park et al., 2003; Tusher and Chu, 2001; Vaquerizas et al., 2005). We now describe the methods applied for the analysis in the identification of differentially expressed genes. They are the conventionally used statistical methods applied in microarray data analysis and all they are implemented in different software platforms for the analysis of microarrays, both in commercial and non-commercial packages.

As we highlighted before, analysis of the set of gene expression profiles obtained from the inflammation problem under study is complex, given the number of samples taken (48 samples in total) and variance due to treatment (inoculated vs. placebo), time (6 different sampling times), and subject phenotype (4 biological replicates in each experimental group). Therefore, we apply each of the methods to account for the time factor as well as for the treatment vs. control factor.

3.2.1 Student's *T*-Test

The Student's T-Test belongs to the Hypothesis Test family, allowing us to assign a probability level to describe the likelihood that we can reject the Null Hypothesis or accept $H_0 : tm_j \neq cm_j$, where tm_j and cm_j are the treatment and control means over replicates for each gene j in the microarray, we calculate $t_a : .$

$$t_j = \frac{tm_j - cm_j}{\sqrt{\frac{t(sd)_j^2 + c(sd)_j^2}{\#samples}}} \tag{2.3}$$

where $t(sd)_j^2$ and $c(sd)_j^2$ represent the treatment and control standard deviations. The value t_j is used to accept or reject the Null Hypothesis of no statistical difference between groups based on the t-distribution at a certain confidence level or p-value. We have used the implementation proposed by Li and Wong (2003).

To account for the time factor, we have applied T-Tests at each time point in our experimental conditions (e.g., perform a T-Test to compare time 0-mean in treatment against time 0 mean in control, then to compare time 2 means, and repeat it for each time point). Once all T-Tests have been calculated, we

combine the genes retrieved in each application of the method by using the union set operator.

3.2.2 Permutation Test

Permutation Tests also belong to the Hypothesis Tests Family and, like T-tests, are based on the mean values of compared populations. T-tests and Permutation tests differ in that Student's T-Test uses a p-value as the measure of genes identified as differentially expressed by chance. In small experiments, a p-value provides statistically significant results, but in large experiments, as it is our case, of 10,000 genes or more, the same p-value means that as many as 100 genes may be identified as informational by chance, an unacceptably high rate. The Permutation Test proposed by (Tusher and Chu, 2001) addresses this problem.Similarly to the T-Test, the Permutation Test estimates the behavior of each gene in different experimental groups by calculating the mean of the gene in each experimental condition. While T-Test uses such mean as a parameter to accept or reject the Null Hypothesis at a certain confidence level or p-value, the Permutation Test compares it to an expected mean, obtained by resampling techniques as the expected difference when comparing statistically equal populations. So, the permutation test decides which genes are significatively differentiated by performing gene-specific d tests, based on the *ratio change* and *standard deviation* of each gene. Define $y_{ij}, i = 1, 2, ...I$ samples and $j = 1, 2, ...J$ genes. The following statistic is performed over each gene

$$d_j = \frac{r_j}{t(sd)_j + s_0} \tag{2.4}$$

where r_j is a score related to the difference in expression in each experimental group of gene a, $t(sd)_j$ is the standard deviation of repeated expression measurements, and s_0 is a smoothing factor to ensure that the variance d_j is independent of gene expression. The computation of r_j and $t(sd)_j$ depends on the type of experiment carried out. Once the gene-specific statistics are computed genes are ordered over their d value.

$$d_1 \leq d_2 \leq .... \leq d_J \tag{2.5}$$

so that d_1 is the largest relative difference, d_2 is the second largest and so on. The control data set is created by permutation of the experimental data sets. The number of permutations depends on the type of experiment studied, i.e, if

there are one, two or several classes or if there are different batches or cell lines involved. With B permutations, d-tests for each permutated data set and we order them over their d_b value.

$$d_1^b \leq d_2^b \leq .... \leq d_J^b \tag{2.6}$$

The expected relative difference $d_{E(j)}$ is calculated as $d_{E(j)}$ for $j = 1, 2, \ldots, J$. The d_j values are plotted against the $d_{E(j)}$ values so that non significative differentiated genes are plotted over the $d_j = d_{E(j)}$ line, and a threshold Δ is defined to determine the distance from the $d_j = d_{E(j)}$ line where genes are selected as significative for being over/under expressed. A valuable feature of this proposal is that it gives estimates of the False Discovery Rate (FDR), which is the proportion of genes likely to have been identified by chance as being significant. This is done by calculating the median number of genes called significant in each of the B permutations with the set threshold Δ. The FDR is computed as the number of falsely called significant divided by the number of genes called significant. We have used the implementation proposed by Tusher *et al.* (2001).

The time factor has been taken into account by only allowing resampling among samples belonging to the same time points, (e.g., allow permutations only among the samples at time 0, then among the samples at time 2 , and repeat it for each time point). Once all the $d_{E(j)}$ values have been calculated, we combine the genes retrieved in each application of the method by using the union set operator.

3.2.3 Analysis of Variance

In an Analysis of Variance (ANOVA) analysis, the relationship between more than two populations is compared simultaneously bases on the mean and variance values among the populations. This provides a significant benefit for problems where more than one factor is being considered simultaneously. This is specially useful since the number of experiments being carried out with array technology taking into account the time factor besides any other treatment/control factor is rapidly increasing, as it shows the fact that over 30% of datasets published in 2005 were time course experiments. ANOVA makes use of an F-distribution, which is based on both, the *intra*-samples *inter*-samples mean. If the calculated F-value is greater than the established F-value for the same degrees of freedom, it can be concluded that the Null hypothesis of no variance between samples is incorrect, and that the values for the different samples are indeed significantly different at a established level of confidence. The

ANOVA model for two factors is defined as:

$$y_{ijkl} = \mu_j + \alpha_{ij} + \beta_{kj} + (\alpha\beta)_{ijk} + \epsilon_{ijkl} \tag{2.7}$$

where i accounts for one experimental group, k for the other experimental group, l for replicates and j for the number of genes. The term μ_j represents the overall mean intensity in expression level for genes in all arrays and experimental groups. The term α_{ij} accounts for one of the factor effects, (e.g. treatment versus control effects), representing the overall differences between the two groups. The term β_{kj} represents the other factor effects (e.g. time, capturing the differences in the overall gene expression level in samples from different time points). The term $(\alpha\beta)_{ikj}$ accounts for the interaction of the two factors, and replicates are needed to estimate this value. The final term, ϵ_{iklj} represents the systematic error present in any of these models. Note that the use of ANOVA over only one of the factors is theoretically equivalent to the use of a Student's T-Test to compare two populations. We have applied ANOVA over one factor for time and treatment vs. control independently, and to account for both factors. We have applied the implementation proposed by Li and Wong (2003).

3.2.4 Repeated Measures Analysis of Variance

The distinguishing characteristic of this statistical method is that it handles observations made under different conditions involving the same subjects, making these observations correlated rather than independent. If the condition being considered is time, the Repeated Measures Analysis of Variance receives the name of Longitudinal Data Analysis. The model assumed is:

$$y_{ijklm} = \mu_j + \alpha_{ij} + \beta_{kj} + (\alpha\beta)_{ijk} + \gamma_m + (\gamma\alpha)_{ijm} + (\gamma\beta)_{ikm} + (\gamma\alpha\beta)_{ijkm} + \epsilon_{ijklm} \tag{2.8}$$

where γ_m is a constant associated with subject and m, and $(\gamma\alpha)_{ijm}$ represents the interaction effects of subject m with each of the factors. All other terms are defined as in Section 3.2.3. Repeated Measures ANOVA has been applied over treatment, over time and over both factors combined with the implementation proposed in the commercial software SAS (statistical analysis systems).

4 Results Applying Statistical Methods

We carry out a detailed evaluation of the performance of the previously described statistical methods to identify genes of interest (i.e., genes with a high level of variation among experimental conditions). Methods are relabeled for

making reference to them in an easier way: Student's T-Test is relabeled as M_1, Student's T-Test considering time as M_2, Permutation Test as M_3, Permutation Test considering time as M_4, ANOVA over treatment vs. control condition as M_5, ANOVA over time as M_6, ANOVA over treatment and time as M_7, RMANOVA over treatment as M_8, RMANOVA over time as M_9 and RMANOVA over treatment and time as M_{10}. The application of each of these methods returns different results applied over the same set of data. In Table 2.1 we show the number of probe sets retrieved by each of the methods, finding numbers varying in a very wide rank: from 612 genes retrieved by M_3 (Permutation Test) to 1734 probe sets retrieved by M_5 (ANOVA over treatment vs. control condition).

Method	Probe Sets Retrieved
M_1	962
M_2	1582
M_3	612
M_4	1676
M_5	1734
M_6	1128
M_7	1410
M_8	1175
M_9	950
M_{10}	810

Table 2.1: Probe sets retrieved as differentially expressed by each of the methods applied (M_1 to M_{10})

We have graphically represented the genes retrieved by each of the methods in Fig. 2.3 and we see how they all differ from each other. None of the methods subsumes the results obtained by the other methods as we can see in the coincidence Table 2.2. The percentages of coincidence are much lower than expected, having methods that only retrieve a 31.11% of probe sets in common (M_3 and M_6) or many other values around 50% (M_3 and M_4, M_5 and M_9, M_1 and $M_{10}\ldots$).

On the one hand, in our particular problem, there are probe sets highly related to the inflammation which are not retrieved applying some of the classic microarray analysis methods individually. That is the case of probe set 206011_at, which is related to probe sets 211367_s_at and 211368_s_at in behavior and in function (apoptosis-related cysteine peptidase), stated as relevant for the inflammation problem in (Calvano et al., 2005). For this particular probe set, the isolated application of classical methods such as M_1 or M_3 with the default p-value and False Discovery Rate respectively would not retrieve such

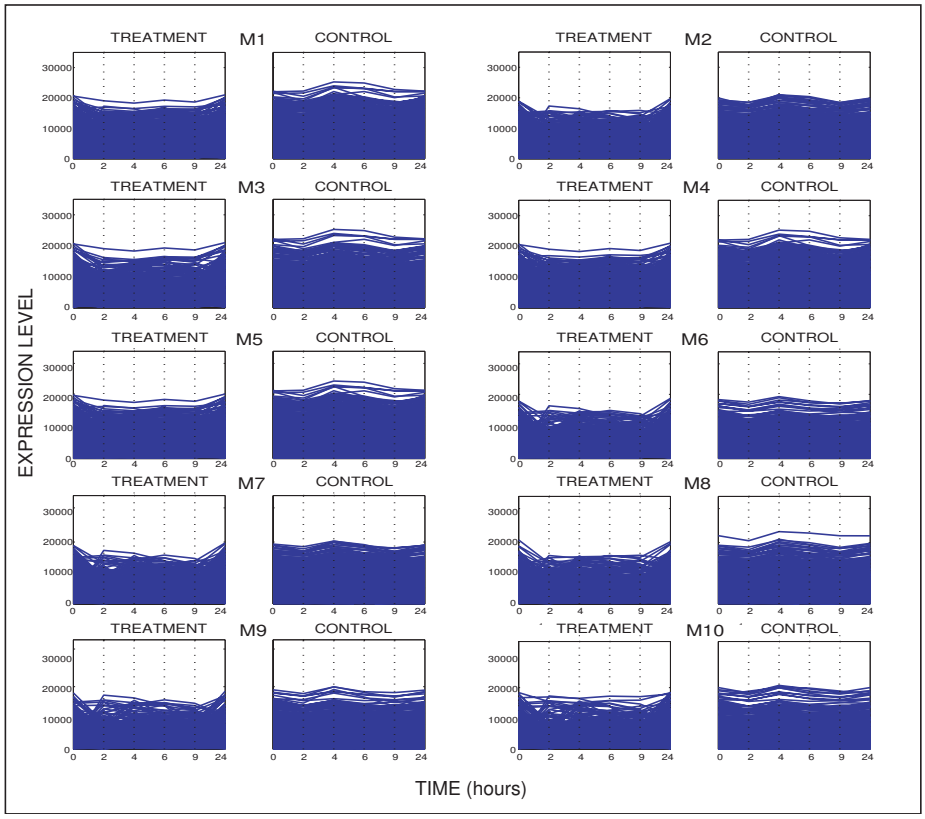


Figure 2.3: Representation throughout time of the probe sets retrieved as differentially expressed by each of the methods applied (M_1 to M_{10}).

probe set as differentially expressed. The same situation applies to probe sets 202076_at and 210538_s_at, related both in behavior and in function (inhibitor of apoptosis protein 2 and 1 respectively). On the other hand, some methods retrieve probe sets that do not differ between the experimental conditions and therefore have no interest for the study. That is the case of ANOVA, with 43% of the probe sets it retrieves applied with the default parameters lacking an observable change with the default parameter values, probably caused by the violation of statistical constraints (Gao and Song, 2005). The increase in the specificity level of the ANOVA parameters generates severe effects on the sensitivity of other true changes. As we see individual methods suffer from missing important probe sets, and oppositely, recover some others which are not significant. Therefore, the described scenario presents a situation with the microarray

%	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}
M_1	–	92.20	52.29	75.05	96.48	69.23	85.55	70.06	61.33	50.52
M_2	56.06	–	34.07	57.84	85.27	59.54	71.11	62.64	50.57	42.98
M_3	82.19	88.07	–	96.24	94.77	57.35	78.75	72.87	56.86	46.73
M_4	67.22	85.19	54.84	–	95.16	55.49	73.65	70.20	51.49	42.83
M_5	55.20	77.80	33.45	58.94	–	50.28	66.72	66.38	46.42	38.93
M_6	59.04	83.51	31.11	52.84	77.30	–	89.63	56.56	60.64	49.38
M_7	58.36	79.79	34.18	56.10	82.05	71.70	–	62.34	57.23	49.07
M_8	57.36	84.34	37.96	64.17	95.96	54.30	74.80	–	49.62	40.51
M_9	62.10	84.21	36.63	58.21	84.74	72.00	84.95	61.36	–	72.31
M_{10}	59.56	83.34	35.05	56.37	82.72	68.26	84.80	58.34	84.19	–

Table 2.2: Percentage of coincidence between methods in retrieving probe sets. This percentage is calculated relative to the number of probe sets retrieved by the method in the column.

analysis methods commonly applied not being capable to extract on their own all the information present in microarray data sets.

5 Concluding Remarks

In this chapter we introduced the challenge in microarray technology, which is essentially identifying the differentially expressed genes, and from a computational point of view, to develop analytical methods to retrieve them. We focussed on studying a problem derived from longitudinal blood expression profiles of human volunteers for the study of the inflammation and the host response to injury, as part of a Large-Scale Collaborative Research Project sponsored by the National Institute of General Medical Sciences (www.gluegrant.org) (Calvano et al., 2005) in a particular microarray experiment carried out at the Washington University School of Medicine in St. Louis, Missouri, in collaboration with the Cellular Injury and Adaptation Laboratory. Analysis of the set of gene expression profiles obtained from this experiment is complex, given the number of samples taken and variance due to treatment, time, and subject phenotype. Therefore, we believe this problem is typical and informative as a RNA microarray case study.

We have described the microarray data analysis process, (see Fig. 2.1) focussing on the low-level analysis (Section 3.1) to account for problems such as microarrays experiments carried out in different conditions (e.g. different days, different labs, different scanning intensities) which will make microar-

ray expression levels run on different ranks, and therefore to be not comparable among each other. The scaling process proposed (*Invariant Set* Approach, (Li and Wong, 2001b)) deals with this problem. We also proposed a normalization process (*Model-Based Expression Index* (MBEI) approach, (Li and Wong, 2001a)) to deal with problems related to variation other than that due to the differences between the RNA samples being studied, such as dye-bias or any systematic or random variation (Nadon and Shoemaker, 2002). We approached high-level analysis (Section 3.2) to select the genes that exhibit a significant behavior throughout the experimental conditions. This analysis includes both gene filtering, which excludes genes from any further analysis based on some criterion, such as the level of hybridization throughout the arrays, and identification of differentially expressed genes applying statistical tests.

Furthermore, we have shown the results obtained by application of such methods to the inflammation and host response to injury problem under study (Section 2). As we have shown, the application of the conventional statistical methods proposed for microarray data analysis returns different results applied over the same set of data (Table 2.1 and Fig. 2.3) since the methods do not identify all observable differentially expressed probe sets; moreover, none of the methods subsume the results obtained by the other methods (see Table 2.2). The microarray analysis methods applied are not capable to extract on their own all the information present in microarray data sets. Missing probe sets, (i.e., probe set not recovered by some of the methods) might contain significant information for the experiment under study as shown in Section 2. Besides this, the methods recover some probe sets that do not differ between the experimental conditions and therefore have no interest for the study.

There is a lack of decision making methodologies capable to decide which methods is the most appropriate for a given microarray experiment. Therefore, new methods or meta-methods are necessary to suggest which microarray analysis method is appropriate for the identification of genes differentially expressed between time, treatment and control.

In an attempt to solve this problem, we propose a machine learning methodology, inspired on conceptual clustering and optimization techniques (Cheeseman and Oldford, 1994; Cooper and Herskovits, 1992) that combines the advantages of each method to extract as much information as possible from genes (or probe sets) present in the microarray data. The method associations and the information they are capable to extract are stored into decision making association rules. These association rules are devoted to discover optimal aggregations of microarray analysis methods in an effort to identify differential expression profiles (i.e., sets of genes with coordinate changes in RNA abundance) (Agrawal93; Zwir05a; Zwir05b). The association rules allow users

to query for the most appropriate method or aggregation of them to retrieve significant probe sets based on the expression profiles they exhibit. Yet, this guideline serves a seed to support decisions on new microarray problems based on matched preferences of differential profiles.

Chapter 3

A Methodology for the Identification of Expression Profiles using Decision Making Association Rules

The greatest challenge in microarray technology development is the analytical process followed to successfully analyze data acquired from microarray experiments. The current bottleneck in the processing of microarray data occurs after the data are generated and the magnitude of the problem is proving to be on a par with developing the technology itself. To address the statistical challenge of analyzing such large data sets, new methods have emerged (Inza et al., 2004; Li and Wong, 2003; Pan et al., 2001; Park et al., 2003; Tusher and Chu, 2001; Vaquerizas et al., 2005). However, we saw in Chapter 2 that microarray analysis methods are not capable to extract all the information present in microarray data sets. We applied some of the most commonly used statistical methods in microarray data analysis to an experiment carried out at the Washington University School of Medicine at St. Louis, Missouri (see Chapter 2 Section 1). Microarray analysis methods do not extract all the information present in microarray data sets. Missing genes, (i.e., genes not recovered by some of the methods), might contain significant information for the experiment under

study. The necessity of a novel methodology capable to retrieve all the information available from microarray experiments, specifically the one that reflects gene expression changes over time, patient and experiments.

In an attempt to solve this problem, here we propose a conceptual clustering approach (Cheeseman and Oldford, 1994; Cook et al., 2001b; Zwir et al., 2005b; Zwir et al., 2005c), which combines the advantages of several microarray analysis methods in an attempt to identify all significant gene expression changes from microarray experiments by identifying profiles exhibiting such profiles (i.e., sets of genes with coordinate changes in RNA abundance). The idea is to utilize the advantages of the microarray analysis methods by combining the methods themselves. This approach has been previously used in the ENCODE Project (Encyclopedia Of DNA Elements), which aims to provide a more biologically informative representation of the human genome by using high-throughput methods to identify and catalogue the functional elements encoded (Guigo and Consortium., 2007).

The proposed methodology is devoted to discover optimal associations of microarray analysis methods in an effort to identify gene expression profiles and encode such information as relations between these profiles and associations of microarray analysis methods capable to identify them. Such encoding produces an optimal set of association rules. These are decision making rules that suggest the best combination of methods designed to recover a desired differential gene expression profile that can change over time, patient and/or experiment. These decision making association rules provide information about which is the most appropriate method to apply for a certain set of microarray data, given the data constraints and the type of method. By storing this information in the form of association rules, it will be available for rapid and straightforward access to make non trivial predictions in new microarray data analysis. We combine concepts from data mining (Adriaans and Zantinge, 1996)(i.e., extraction of data to create the association rules), optimization (Chankong and Haimes, 1983) (i.e., evaluation of the data acquired for the association rules) and decision making (Evangelos, 2000)(i.e., use of the association rules in further microarray experimental data).

The methodology described throughout this chapter has three high level phases: (1) Identification of rules that suggest the best method association capable to retrieve the desired differential gene expression profiles; (2) The design of a method to biologically validate the extracted differential profiles; and (3) The incorporation of new models of profiles that encode the dynamics of gene expression. In this chapter, we extensively develop the firsts phase, while the others are described in Chapters 4 and 5 respectively. Here we are focussed in the creation of a set of association rules relating the methods for microarray

data analysis to the gene expression profiles they are able to recover. This phase includes identification of the gene expression profiles to be used, creation of the associations of methods which will be used for retrieving the gene expression profiles, evaluation of the performance of such methods associations and creation of the set of decision making association rules (see Fig. 3.1). We will make use of such decision making association rules for an appropriate retrieval of genes exhibiting an specific expression profile from other sets of microarray data. The association rules created based on this information present some noteworthy characteristics, harboring as the capacity to combine them under certain conditions without lost of optimality in the non-dominance relation; the possibility of sets of profiles to the antecedents of the rules and a conflict resolutor mechanism by some operator such as a weighted sum of the objectives, product or OWA operator (Herrera et al., 1994). We also provide a mechanism for simplifying the sets of rules, providing rules at different levels of granularity, which allows the derivation of several emerging properties.

Throughout this chapter we will describe each of the steps of the proposed methodology and the results obtained when applied to the Inflammation and Host Response to Injury problem described in the previous chapter and compare the performance with those methods proposed in the previous chapter as well as with random variations of profiles. Inflammation is a hallmark of many human diseases (Coussens and Werb, 2002). Understanding the inflammation process is critical because the body uses inflammation to protect itself from infection or injury (e.g., crushes, massive bleeding, or a serious burn) which, in extreme cases (e.g., car accidents or gun shootings), can lead to massive organ malfunction and death. According to the U.S. Centers for Disease Control's National Center for Health Statistics, unintentional injury is the leading cause of death for people ages 1 to 35 (Calvano et al., 2005). Besides, analysis of the set of gene expression profiles obtained from this experiment is complex, therefore we believe this problem is typical and informative as a RNA microarray case study given the number of samples taken and variance due to treatment, time, and subject phenotype.

1 Methodology Description

1.1 Identification of Differential Profiles

In order to identify genes of interest, we need software tools capable to select and screen candidate genes for further research. At the simplest level, we can determine which genes show significant expression changes compared with a control

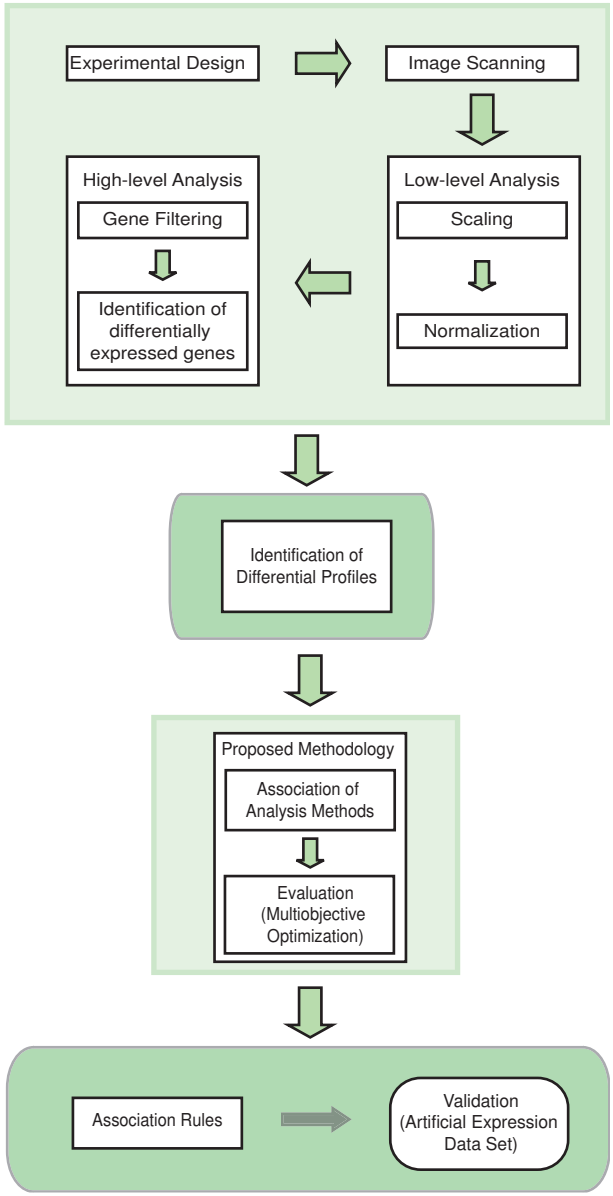


Figure 3.1: Microarray analysis modeling: Steps for the creation of the association rules

group in pair-wise comparison. As data sets become more complex, covering a variety of biological conditions or time series, one may think of several scoring methods for selecting the most interesting genes, considering if there has been a significant change at any condition, there has been a significant aggregate change over all conditions or whether the fluctuation pattern shows high diversity according to Shannon's entropy (Fuhrman et al., 2000). Beyond straightforward scoring methods, we would like to classify gene expression profiles to explore shared functions and regulations. Genes sharing the same expression profiles are likely to be involved in the same regulatory process. Though in theory there is a big step from simple correlation analysis to gene interaction networks, several papers indicate that the clustering of gene expression data does result in groups of genes that have related functions (D'haeseleer et al., 2000; Eisen et al., 1998).

Throughout this work we will refer to *gene expression profiles* as sets of genes which exhibit a common behavior throughout the conditions of the problem under study. Time and patient are the grouping factor for the gene expression profiles. The representation used for the gene expression profiles takes into account the treatment vs. control condition, calculating different groups of co-expressed genes for the treatment and control group. Therefore, we can find a gene grouped with some other genes sharing a common profile at the treatment data set, and the same gene sharing profile with other different genes at the control data set (see Fig.3.2). The time condition has also been taken into ac-

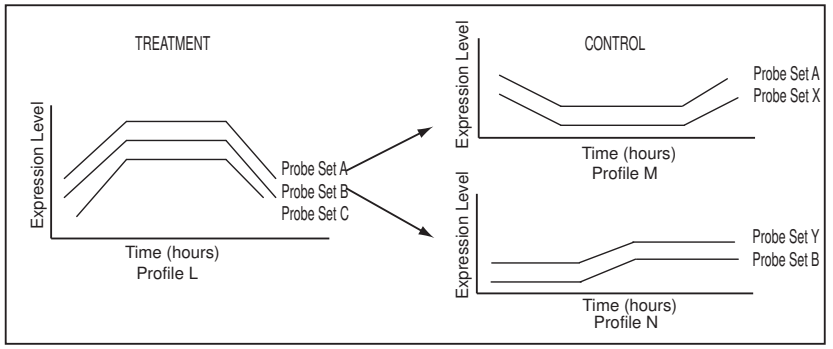


Figure 3.2: Profile representation. A gene can be associated to different genes in treatment and control groups

count for profile representation. The most commonly used practice when dealing with sample replicates and longitudinal data consists of calculating some sample replicate averages taking into account statistic measures to detect outliers and preventing from data bias affecting the final average (Li and Wong, 2003). If we

had followed this practice, we would have averaged for each gene the expression value of replicate or patient 1 at time 0, with replicate of patient 2, 3 and 4 at time 0. We would have followed a similar procedure with the four patients at times 2, 3, 6, 9 and 24. The same process would have been used for the control data set. This practice is useful since it aggregates the values of all sample replicates in just one data set, making it easier to handle and to compare to the control group. However, since we are dealing with biological replicates, human patients in this case, it might be the case that different individual behavior of some of the genes is not due to biased values but to conditions not previously considered in the experiment, such as gender of the sampling volunteer or age. In Fig. 3.3 we show an example of this, and the representation method applied individually for each of the patients throughout time. The construction of gene

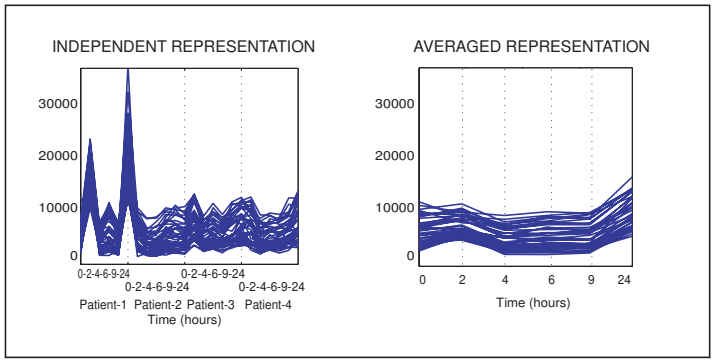


Figure 3.3: Difference between independent and averaged representation of the patients

expression profiles, based on condition correlated groups, is commonly accomplished using clustering methods. Clustering allows us to extract groups of genes that are tightly co-expressed over a range of different experiments. For example, Tavazoie *et al.,* (1999) identified 18 biologically significant DNA-motifs in the promoter region of gene clusters based on cell-cycle expression patterns. Most motifs were highly selective for the cluster in which they were found. We construct the gene expression profiles independently for the treatment and control datasets, calculating different groups of co-expressed genes for the treatment P_T and the control P_C group, taking time and independent representation of each patient as the grouping factor in each of these two conditions.

We apply a classic clustering algorithm, K-means (Duda and Hart, 1973), for identification of gene expression patterns in the inflammation problem data set. The K-means clustering can be described as a partitioning method which

groups the observations in your data into non-overlapping clusters. The algorithm runs an iterative process, where each record is assigned to the closest centroid. New centroids are calculated for the resulting clusters and the records are reassigned to the closest centroid. The process automatically stops once a steady state has been reached. The similarity measure chosen has been the Euclidean Distance, a classical distance measure, since distance measures have exhibited a better behavior than correlation based measures for gene grouping in this particular problem. The K-means algorithm needs the number of resulting clusters, k, as an input parameter. The value k is estimated by application of the *Davies-Bouldin* validity index (Davies and Bouldin, 1979). This index detects compact representations of K-means partitions (Bezdek, 1998b) by choosing the cluster size c that minimizes the following formula through different number of clusters (i.e., $c = 2$ to $c = \sqrt{n}$):

$$DB(U, \bar{V}, X) = \left(\frac{1}{c}\right) \sum_{i=1}^{c} \left[max_{(j.j \neq 1)} \frac{\alpha_i + \alpha_j}{\|\bar{v_i} - \bar{v_j}\|} \right] \tag{3.1}$$

where the dataset is partitioned as $X = U_{i=1}^{c} X_i; \|X_i\| = n_i; X_i \cap X_j = \varnothing$ for i $\neq j; \|\|$ is the Euclidean distance; each centroid defined as $\bar{v_i} = \sum_{x \in X} \frac{x}{n_i}$ for each X_i; the total cluster centroids are calculated as $\bar{V} = \bar{v_1}, \ldots, \bar{v_c} and \alpha_i = \sum_{x \in X_i} \frac{\|x - \bar{v_i}\|}{|X_i|}$.

As we said, we can find a gene grouped with some other genes sharing a common profile at the treatment data set, and the same gene sharing profile with other different genes at the control data set (see Fig.3.2). We are interested in knowing what profiles P_T are actually related to profiles P_C, (i.e., which genes exhibiting the behavior describe by P_{Tm} in treatment are exhibiting behavior P_{Cn} in control). We define a *differential profile* by a triplet (P_{T_m}, P_{C_n}, G) which represents a set of genes G which exhibit behavior P_{Tm} in the treatment data set and behavior P_{Cn} in control. The differential profiles consider all changes in time, patient, and experimental condition. The identification of the (P_{T_m}, P_{C_n}, G) is not an straight forward task since as we said before, we can find a gene grouped with some other genes sharing a common profile at the treatment data set, and the same gene sharing profile with other different genes at the control data set (see Fig.3.2). The, to identify differential profiles we apply a coincidence index (CI) based on the hypergeometric distribution (p-value 0.05) (Tavazoie et al., 1999), which determines the statistical significance of overlap between pairwise profile association in treatment and control conditions:

$$CI(P_T, P_C) = 1 - \sum_{q=0}^{p} \frac{\binom{h}{q}\binom{q-h}{n-q}}{\binom{g}{h}} \tag{3.2}$$

that gives the chance probability of observing at least p candidates from P_T of size h within another set P_C of size n, in a universe of g candidates. Therefore, genes belonging to a cluster in treatment, P_T , can fit in more than one cluster in control, P_C, and vice versa.

We propose a methodology that is devoted to discover optimal associations of microarray analysis methods in an effort to identify *differential profiles* (triplet (P_{T_m}, P_{C_n}, G), which represent sets of genes (or genes in our particular problem) G which exhibit different behavior in the treatment P_{Tm} and in the control P_{Cn}). It is noteworthy that we also account for distinct behavior among patients, based on independent representation. The methodology encodes such information, relation between associations of microarray analysis methods and the differential profiles they optimally recover in a set of association rules.

1.2 Creation of Microarray Analysis Method Associations

We propose a methodology that combines the advantages of several microarray analysis methods. Using this combination we can deal with the problem of methods not being capable to extract on their own all the information present in microarray data sets (see previous chapter). The microarray analysis methods can be associated by *combining the results obtained upon their application.* We propose an association based on set theory (Halmos, 1960), making use of two classical set operators: union ($\cup$) and intersection ($\cap$). We define M_i, union of two methods M_a and M_b ($M_a \cup M_b$), as the set resulting of including all genes retrieved by M_a (G_a) and all genes retrieved by M_b (G_b).

$$M_i = M_a \cup M_b = G_a \cup G_b \tag{3.3}$$

Similarly, we define the M_j, intersection of two methods M_a and M_b ($M_a \cup M_b$) as the set resulting of including all genes retrieved by M_a (G_a) which have also been retrieved by M_b (G_b).

$$M_j = M_a \cap M_b = G_a \cap G_b \tag{3.4}$$

All potential combinations of microarray analysis methods $M_1, \ldots, M_n$, using the $\cup$ and $\cap$ operators, conform a space of potential hypotheses which can be represented as a lattice structure. We will perform a search, moving from hypothesis to hypothesis, towards the most general (union of all methods), and the most specific (intersection of all methods), which are located at the top and the

bottom of the lattice respectively (Mitchell, 1997)(see Fig. 3.4). The positions between the top and the bottom of the lattice represent all possible methods and combinations of methods obtained by applying the union and intersection operators. The closer a solution is to the top of the lattice (i.e., more general solutions obtained by application of the union operator), the higher rate of statistical *Type I* error they suffer. The closer a solution is to the bottom of the lattice (i.e., more specific solutions obtained by application of the intersection operator) the higher the rate of statistical *Type II* error they suffer (see Section 1.3).

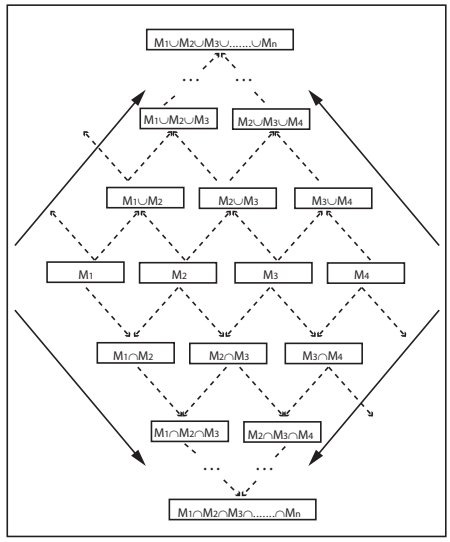


Figure 3.4: Lattice of potential hypotheses, method associations $M_1, \ldots, M_n$ using the $\cup$ and $\cap$ operators. The solid arrows show the direction of the search in the space of hypotheses

1.3 Evaluation of Method Association Performance

The methodology proposed in this work makes use of associations of microarray analysis methods as described in Section 1.2 so that we can combine the advantages of each method to extract as much information as possible related to the differential profiles (P_{T_m}, P_{C_n}, G) created in Section 1.1. We need to evaluate the performance of the method associations to retrieve the differential profiles

in order to find the optimals. We say a method M_i is capable to *retrieve* a differential profile (P_{T_m}, P_{C_n}, G), if it identifies at least a percentage t of the genes belonging to such profile in the data set. The value of t is set to satisfy the statistical power of 80% (Cohen, 1992).

Many real world problems involve simultaneous optimization of several competing objectives. Usually, there is no single optimal solution, but rather a set of alternative solutions (Deb and Reddy, 2003). These solutions are optimal in the wider sense that no other solutions in the search space are superior to them when all objectives are considered. Maximizing profit and minimizing the cost of a product; maximizing performance and minimizing fuel consumption of a vehicle; and minimizing weight while maximizing the strength of a particular component are examples of multi-objective optimization problems. If a multiobjective problem is well formed, there should not be a single solution that simultaneously minimizes each objective to its fullest. In each case we are looking for a solution for which each objective has been optimized to the extent that if we try to optimize it any further, then the other objective(s) will suffer as a result. Finding such a solution, and quantifying how much better this solution is compared to other such solutions (there will generally be many) is the goal when setting up and solving a multiobjective optimization problem (Chankong and Haimes, 1983; Deb and Reddy, 2003).

Given a problem with two competing objectives, O_i and O_j, and two solutions a and b, we define a *dominance* relation of a with b, a *dominates* b as:

$$a \succ b \quad iff \quad \forall i \; O_i(a) \geq O_i(b) \; \wedge \; \exists \, k O_j(a) > O_j(b) \tag{3.5}$$

This can be read as a performs better or equal than b in one of the objectives and a performs better than b in the other objective. If there is at least one objective so that the relation $O(a) \geq O(b)$ does not hold, we say that a and b are not comparable and have a *non-dominance* relation. A solution a is not dominated with respect to the set of all possible solutions if there is no solution that dominates a. The *Pareto optimal front* (Chankong and Haimes, 1983; Deb and Reddy, 2003) is defined as the set of non-dominated solutions with respect to the whole solution space (see Fig. 3.5).

The optimization of the method associations when retrieving differential profiles from a certain data set is based on three different objectives: (1) specificity, percentage of genes (or probe sets) retrieved by the method association that belong to the differential profile (P_{T_m}, P_{C_n}, G) under evaluation, (2) sensitivity, percentage of genes or probe sets in the microarray data set D belonging to differential profile (P_{T_m}, P_{C_n}, G) which are retrieved from the total amount of genes in D belonging to (P_{T_m}, P_{C_n}, G) and not to other differential profiles, and (3), cost of application of all methods. Before formally defining this terms, we

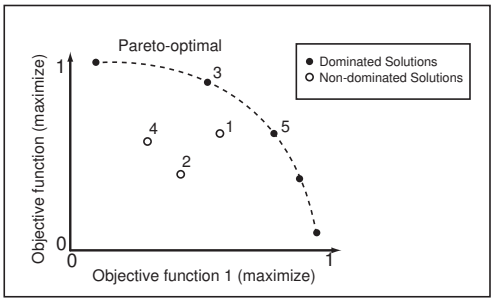


Figure 3.5: Example of Pareto optimal front. The objective function is the maximization of the two objectives.

will make some other definitions:

- True Positive (TN): probe sets retrieved by method M_i from the microarray data set D, which *do* belong to a differential profile (P_{T_m}, P_{C_n}, G) being asked for.
- True Negative (TN): probe sets *not* retrieved by method M_i from the microarray data set D, which *do not* belong to a differential profile (P_{T_m}, P_{C_n}, G) being asked for.
- False Positive (TN): probe sets retrieved by method M_i from the microarray data set D, which *do not* belong to the differential profile (P_{T_m}, P_{C_n}, G) being asked for.
- True Negative (TN): probe sets *not* retrieved by method M_i from the microarray data set D, which *do* belong to the differential profile (P_{T_m}, P_{C_n}, G) being asked for.

Based on these terms, we define specificity, sensitivity and cost as:

$$Specificity = \frac{TP}{TP + FP} \tag{3.6}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{3.7}$$

$$Cost = 1 - \frac{\#Methods}{Max(MethodsAvailable)} \tag{3.8}$$

The rate of FP is directly associated to the statistical *Type I* error, or error of rejecting a null hypothesis when it is actually true. The null hypothesis in our case is that there is no difference in the gene expression level between experimental conditions (i.e., the genes behave in a similar manner in all experimental conditions). Type I error represents genes which are not differentially expressed between experimental conditions but are retrieved as so by the applied methods. *Type II* error, the error of accepting a null hypothesis when the alternative hypothesis is the true state of nature, is related to FN. Type II error represents genes differentially expressed between experimental conditions which are not retrieved by the applied methods. These factors are affected by the operator applied in the combination of methods. The intersection operator provides a set of genes where each gene has been selected by all methods involved in the combination of methods. Therefore, intersection favors specificity since the rate of FP is decreased, at the cost of increasing Type II error. Meanwhile, the union operator provides a set of genes that only need to be selected as differentially expressed by one of the methods in the method combination being evaluated. The union favors sensitivity since the rate of FN is decreased, increasing the Type I error.

To evaluate the behavior of the method associations against the differential profiles, we will move in the lattice of potential hypotheses, evaluation each of the nodes (method associations), starting at the vertical center of the lattice (methods applied on isolation) and moving simultaneously towards the top and the bottom, of the lattice, evaluating methods associations containing a higher number of methods each time, associated with the $\cup$ and $\cap$ operators. The methodology is not fully exhaustive, since it does not necessarily reach the top (union of all methods) or the bottom (intersection of all methods) of the lattice. For each differential profile, we will reach a point where the introduction of new methods in the method associations, either for the $\cup$ or for the $\cap$ operators, will not improve the values of the specificity or sensitivity objectives, and cost objective will be increased with each new method added. At such point, the search in the lattice of the hypotheses will be stopped.

1.4 Creation of Decision Making Association Rules

We propose a methodology that makes use of associations of microarray analysis methods as described in Section 1.2 so that we can combine the advantages of each method to extract as much information as possible related to the differential profiles (P_{T_m}, P_{C_n}, G) created in Section 1.1. The idea beyond the methodology is to store into decision making association rules information related to which of the method associations is optimal to retrieve the differential profiles. There

are many different microarray data analysis methods available, and it is hard to know which methodology is the most appropriate to apply depending on a particular microarray dataset based on its constraints and the type of information being asked for. This task becomes even harder when there is not a high level of background knowledge on microarray statistical specific issues , as it is the case in many small laboratories, where the people to carry out the microarray experiments and to analyze them are the same, and usually closer to biological fields than to statistical and mathematical fields (Nadon and Shoemaker, 2002). The methodology we propose solves this problem by automatizing the decision of which is the most appropriate microarray analysis method to use given a particular dataset. By storing this information as decision making association rules, it will be available for rapid and straightforward access to make non trivial predictions in new microarray data analysis.

Decision making (Evangelos, 2000; Herrera et al., 1997) is the cognitive process of selecting the best alternative (or alternatives) from among multiple different alternatives. In our context we say that we have a finite set of feasible alternatives for the problem $X = x1, x2, ..., xn,\ n \geq 2$ from where we want to obtain a solution set of alternatives $S|S \subset X, S \neq 0$; (the best alternative(s) to solve the problem). These best or optimal alternatives to solve the problem, optimal method associations to recover differential profiles, are stored in what we call decision making association rules. Rule-based systems can be considered as a knowledge extraction or data mining tool to discover intrinsic relationships contained in a data base (Freitas, 2002). Association rules show attribute value conditions that occur frequently together in a given dataset. Thus, allowing to encode relationships among different variables and permitting to derive patterns contained in the examined data. In our case, these are relationships between methods that can detect different profiles. In knowledge discovery, the process to obtain these patterns must be automatic, or semiautomatic, discovered patterns must be comprehensible and they must provide useful information, and data must be invariably presented in substantial quantities (Witten and Frank, 2000). Useful patterns allow us to make non trivial predictions about new data.

In our problem, the decision making association rules store information related to the differential profile and the optimal method or association of methods capable to retrieve them. By storing this information in the form of association rules, it will be available for fast and straightforward access to make non trivial predictions in new microarray data analysis. We obtain a set of association by applying knowledge discovery to the information learned relating the differential profiles and the optimal methods or associations of methods capable to retrieve them. The decision of which method or association of methods is optimal for a differential profile is made by application of a multiobjective optimization tech-

nique based on the sensitivity, specificity and cost of the methods to retrieve the differential profiles (see Section 1.3). The rules are created as follows.

1.4.1 Creation of a Non-dominance Lookup Table

We extract all non-dominance relationships between method associations and differential profiles recognized by them from the evaluation phase (see Section 1.3) and store them in a lookup table, where method associations lie in rows and differential profiles in columns (see an example in Fig. 3.6).

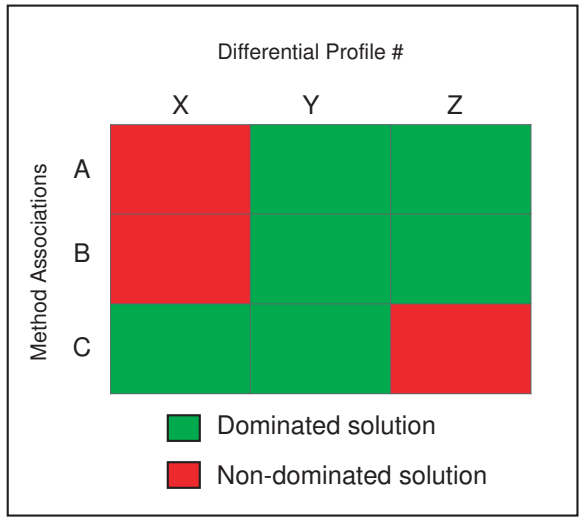


Figure 3.6: Sample lookup table. The red cell in the table represents that the differential profile in the column is retrieved by the method association in the row.

The red cell in the table represents that the differential profile in the column is retrieved by the method association in the row (e.g., differential profile X is optimally recognized by method association A). The lookup table contains the information necessary for the creation of the decision making association rules. From the example table in Fig. 3.6, three association rules can be derived:

- $R_1 : \; IF \; DP_X(P_{TX}, P_{CX}, G_X) \; THEN \; M_A$
- $R_2 : \; IF \; DP_X(P_{TX}, P_{CX}, G_X) \; THEN \; M_B$

- $R_3 :\ IF\ DP_Z(P_{TZ}, P_{CZ}, G_Z)\ THEN\ M_C$

where M_A, M_B and M_C can be either union or intersection of initial methods. At this point, by application of the proposed methodology, we have created a set of association rules which establish a link between differential profiles and the optimal method associations to retrieve them. We also propose a decision making mechanism, in order to decide which association rule is appropriate for each particular query against our methodology. This methodology is described as part of the inference methods to fire the appropriate association rules.

1.5 Decision Making

The inference process is devoted to decide which association rules is appropriate to use when trying to retrieve differential profiles from new or unseen data. The idea is the use of differential profiles as a query and obtain optimal recommendation of method associations from the already learned rules. In fact, the rule base can be updated based on new knowledge acquired from new data.

1.5.1 Rule firing and Conflict Resolution Approaches

A single rule can be fired using a distance norm to the prototype or centroid of a differential profile. We applied the Euclidean norm and calculate the centroid by averaging the observation of each differential profile. Then a rule

$$R_i :\ IF\ X\ Is\ DP_X(\overline{P_{TX}, P_{CX}}, G_X)\ THEN\ M_A$$

can be fired at different α degrees $i = |x - DP_X(\overline{P_{TX}, P_{CX}}, G_X)|$

If we were asked for the optimal method association to retrieve differential profile $DP_X(P_{TX}, P_{CX}, G_X)$, in the lookup table proposed in Fig. 3.6, two association rules would be fired, R_1 and R_2, with method associations M_A and M_B respectively.

- $R_1 :\ IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_A$
- $R_2 :\ IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_B$

and we would have to choose between the two of them. To do so, we propose an *aposteriori* treatment of the non-optimal solutions obtained in Section 1.3.

The decision criteria among this rules is based on the objective values achieved for each of the rules on the multiobjective evaluation. We define a coefficient C_i which is created based on a functional ƒ. The general formula is:

$$ƒ(M_i) = ƒ(\omega_1 O_1, \omega_2 O_2, \omega_3 O_,) = C_i \tag{3.9}$$

where ω_1, ω_2 and ω_3 are weights that we can apply to each of the objectives and O_1, O_2 and O_3 are the specificity, sensitivity and cost objectives respectively. ƒ can be implemented by different operators, such as a weighted sum of the objectives, product of the objectives or OWA operator (Herrera et al., 1997), depending on the kind of information which results more relevant for the expert performing the queries. For each functional ƒ implemented, a simplidied lookup table is extracted with a single method association for each differential profile queried (see Fig. 3.7). Method associations with the best coefficient C_i value are highlighted with a black line.

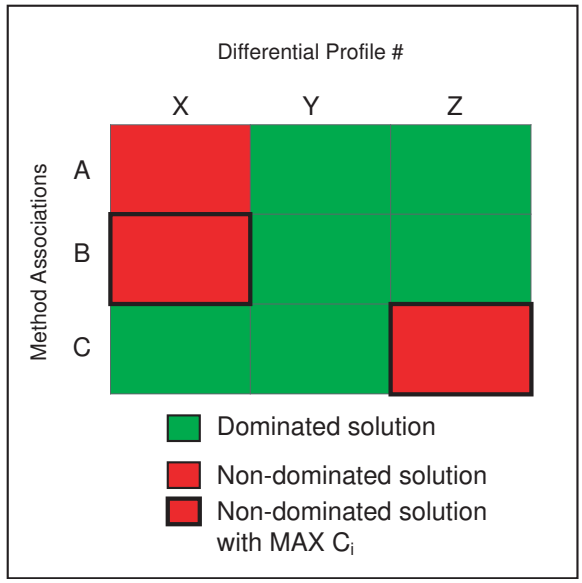


Figure 3.7: Sample lookup table to query an individual profile with implementation of the operator to create a coefficient C_i.

The new set of rules obtained from this lookup table would be

- $R_1 :\ IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_B\ with\ C_i$
- $R_2 :\ IF\ DP_Z(P_{TZ}, P_{CZ}, G_Z)\ THEN\ M_C\ with\ C_i$

Profile	Method Association
$DP_X(P_{TX}, P_{CX}, G_X)$	M_B
$DP_Z(P_{TZ}, P_{CZ}, G_Z)$	M_C

Table 3.1: Method associations for individual profiles with coefficient C_i

The information from the lookup table can be now rewritten in a single entry table such as Table. 3.1.

Note that different operators to obtain the C_i coefficient result in different tables of method association related to individual differential profiles over the same set of data.

1.5.2 Inference

Several fired rules can be consolidated in a single decision making recommendation by using a T-conorm fuzzy operator

$$R(X) = T - Conorm(R_1(X), \ldots, R_k(X))$$

The former inference process can be improved by using more complex classification rules (Herrera et al., 2007). To do so, we can re-write a rule as

$$R_i : \; IF \; X \; Is \; DP_X(\overline{P_{TX}, P_{CX}}, G_X) \; THEN \; M_A \; with \; C_i$$

where C_i is the confidence of the rule as defined in Section 1.5.1. Several antecedents can be combined by the typical fuzzy T-norm operators such as the minimum. The rules are fired as typical fuzzy classification rules (Cordón et al., 1999):

$$R_i = \alpha_i \times C_i$$

We can be queried for sets of more than one differential profile. For instance, our system could could be queried for "genes belonging to differential profiles exhibiting different behavior among patients in the treatment group". In our set of differential profiles, this condition is accomplished by differential profiles #5, #15 and #22 see Fig. 3.14 and Fig. 3.15. Or in our particular example, a query for differential profiles X and Z simultaneously. Therefore, we need a global decision making strategy to account for this possible situations. We now describe such strategy.

1.5.3 Identification of Single-profile Association Rules.

We build association rules that suggest the best method association for each *individual* differential profile, where optimality is based on non-dominance relationship among these associations. To do so, and for each individual rule R_i, we consider each differential profile as an antecedent, and we scan the lookup table by column, assigning a consequent (method association) for each non empty cell (red cell). From our example lookup table in Fig. 3.6 and differential profile $DP_X(P_{TX}, P_{CX}, G_X)$ the association rules obtained are

- $R_1 : \; IF \; DP_X(P_{TX}, P_{CX}, G_X) \; THEN \; M_A$
- $R_2 : \; IF \; DP_X(P_{TX}, P_{CX}, G_X) \; THEN \; M_B$

1.5.4 Identification of Multiple-profile Association Rules.

We construct decision making rules that suggest the best method association for a set of *more than one* differential profiles. To do so, we scan the lookup table by rows and assign the corresponding method association as the rule consequent, and recover all non-empty column cells (red cells) (differential profiles) as a conjunction of the rule antecedents. We can find three different situations. (

1) A first case, with all differential profiles being asked for having a common method association (consequent) to retrieve them (see Fig. 3.8),

In this first case, the rules will be made out of several antecedents, or list of distinct differential profiles, which are related to the same consequent, or method-association. It should be noted that the optimality -in terms of non-dominance- of these method-associations is preserved when two or more differential profiles included in a composite antecedent are optimally recognized by the same method-association. Given two differential profiles recognized by a method association, $DP_X(P_{TX}, P_{CX}, G_X) \; \rightarrow \; M_A$ and $DP_Y(P_{TY}, P_{CY}, G_Y) \; \rightarrow \; M_A$

- , or written in terms of the rules:
- $R_i : \; IF \; DP_X(P_{TX}, P_{CX}, G_X) \; THEN \; M_A$
- $R_j : \; IF \; DP_Y(P_{TY}, P_{CY}, G_Y) \; THEN \; M_A$

implies that a single rule

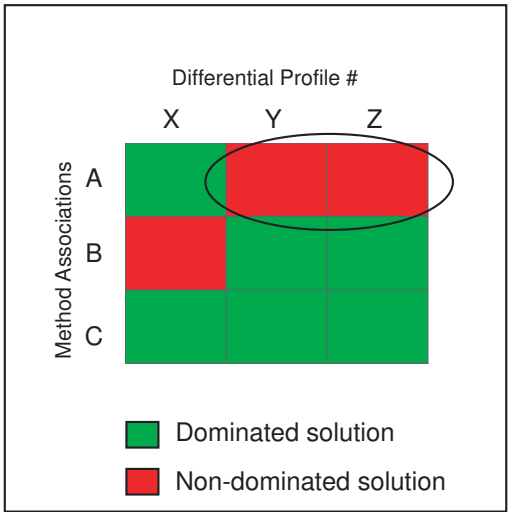


Figure 3.8: We show an example of differential profiles being asked for (Y and Z) having a common method association (A) to retrieve them.

$R_k : IF\ [DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]\ THEN\ M_A$ can be created by composition of the antecedents and keeping the consequent, without lost of optimality based on the non-dominance relation (i.e., M_A is non-dominated in relation to other method associations in the lattice of potential solutions to recover both $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$). This can be proved as follows. (1) M_A is optimal for the recovery of $DP_X(P_{TX}, P_{CX}, G_X)$, therefore, M_A is a non-dominated solution in relation to the sensitivity, specificity and cost objectives in relation to the other solutions in the lattice. The same situation applies to $DP_Y(P_{TY}, P_{CY}, G_Y)$. When M_A recovers both differential profiles $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$) together, the sensitivity value is still optimal, since M_A was optimal for both of them independently. The cost objective is also optimal, since we have not changed the number of methods in the method association M_A. Moreover, while we consider in current implementation the disjoint differential profiles. (2) Specificity levels in can M_A, although optimal, could have been decreased by M_A recognizes genes from other differential profiles than $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$ when evaluation of M_A for each of them independently. If such genes do not belong either to $DP_X(P_{TX}, P_{CX}, G_X)$ or to $DP_Y(P_{TY}, P_{CY}, G_Y)$ the specificity levels are conserved, still being optimal. However, if M_A recovers genes from $DP_X(P_{TX}, P_{CX}, G_X)$ while evaluating $DP_Y(P_{TY}, P_{CY}, G_Y)$ or viceversa, it is straightforward to infer that the specificity level will be improved by considering the retrieval of both differential

profiles. From (1) and (2) we can deduce that if there would be another method M_B that optimally recognizes $[DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]$ either dominating or being non-dominated with M_A, it would already be. This is one of the most important improvements obtained by using a multiobjective Pareto frontier instead of summarization of objectives into a single function (Zwir et al., 2005b; Ruspini and Zwir, 2002).

(2) A second case, where more than one method association satisfies the condition of being optimal for the differential profiles being asked for (see Fig. 3.9), then a conflict resolutor needs to be applied. We apply the same functional ƒ applied in Section for decision among non-dominated method associations for the retrieval of individual profiles (i.e., an operator such as weighted sum of the objectives, product of the objectives or OWA operator (Herrera et al., 1997))

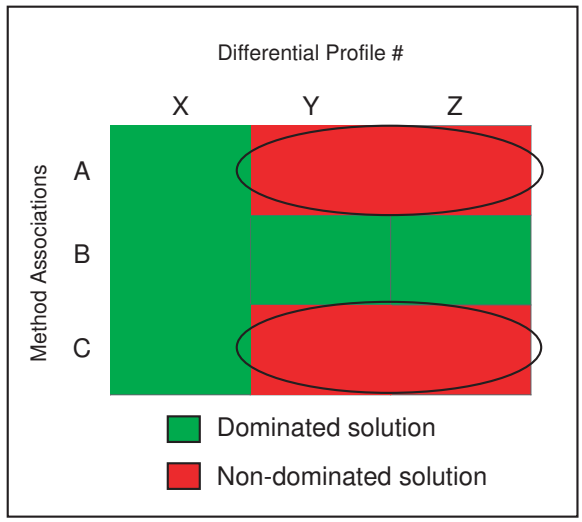


Figure 3.9: Example of more than one method association satisfying the condition of being optimal for the differential profiles being asked for. Differential profiles X and Z are both optimally recovered by method associations A and C .

(3)A third case, where the rules will be made out of several antecedents, or list of distinct differential profiles, which are related to different consequents, or method associations. In Fig. 3.10 we see an example of this.

We obtain, for differential profiles $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$ the following rules:

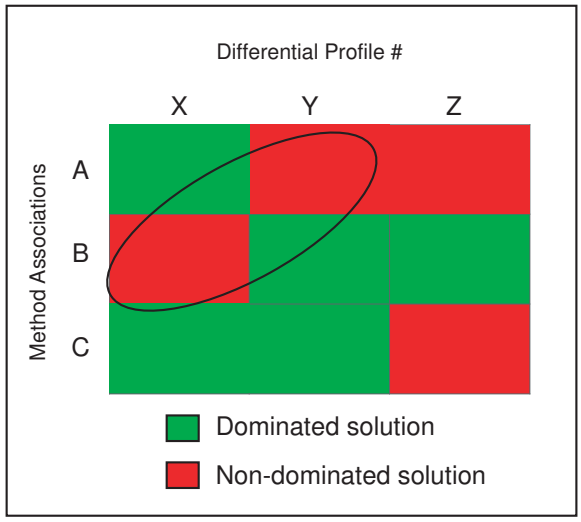


Figure 3.10: Example of distinct differential profiles, which are related to different consequents. Differential profile X is retrieved by method association B and Y is retrieved by method associations A.

- R_i : $IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_A$ and
- R_j : $IF\ DP_Y(P_{TY}, P_{CY}, G_Y)\ THEN\ M_B$ with $A \neq B$

Although rules with the same consequent (method associations) preserve optimality on the non-dominance relation with respect to other solutions in the lattice, as we have just seen, when two or more consequents (method associations), are combined, we can not assure that the optimality is preserved. Given

- R_i : $IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_A$ and
- R_j : $IF\ DP_Y(P_{TY}, P_{CY}, G_Y)\ THEN\ M_B$ with $A \neq B$,

does not imply the existence of

R_k : $IF\ [DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]\ THEN\ [M_A; M_B]$
This happens because it could be the case that we found in the lattice another solution M_C discarded in the non-dominance individual evaluation of $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$ that resulted non-dominated

for the evaluation of both $[DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]$ together. It could be the case that M_C had equal sensitivity and cost values than M_A for recognizing $DP_X(P_{TX}, P_{CX}, G_X)$, and the same situation applies with M_C and M_B in relation to $DP_Y(P_{TY}, P_{CY}, G_Y)$. However, M_C might have worst specificity values than M_A and M_B when recovering $DP_X(P_{TX}, P_{CX}, G_X)$ and $DP_Y(P_{TY}, P_{CY}, G_Y)$ because it recovers mixed genes from both differential profiles. This could be the reason why M_C is dominated by M_A and M_B when both profiles are independently retrieved. It is easy to see that the specificity value of M_c can be increased by considering both profiles together $[DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]$, thus overcoming the specificity results of M_A and M_B and becoming a non-dominates solutions. If the specificity values of M_C overcome M_A and M_B, and as we said in the sensitivity and cost levels of M_C, M_A and M_B are equal, then M_C is now a non-dominated solution for the recovery of $[DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y)]$. Upon the risk of ignoring an optimal solution for combination of different consequents, we need to recalculate the evaluation process of each of the solutions in the lattice over the antecedents (set of differential profiles), and create new rules based on this evaluation. This is crucial when the user requirements specifically ask for retrieval of a fixed set of differential profiles (e.g., "genes belonging to differential profiles exhibiting different behavior among patients in the treatment group". In our set of differential profiles, this condition is accomplished by differential profiles #5, #15 and #22 see Fig. 3.14 and Fig. 3.15).

1.6 Hierarchical Association Rules

The decision rules obtained can be summarized and reduce the rule base complexity by clustering their scope. This provides a minimal set of rules with maximum coverage of antecedents (differential profiles). To achieve this goal we apply hierarchical clustering (Li and Wong, 2003; Salvador and Chan, 2004) to the data representing the behavior of the method associations to each of the differential profiles (see Fig. 3.17). This type of clustering has been chosen since it creates a hierarchy represented as a tree (or dendrogram). Therefore, relations can be established at different levels of the dendrogram, from the leaves (individual elements to clusterize) to the root (single cluster containing all elements), providing different levels of granularity to the association rules obtained. The similarity measure chosen for the hierarchical clustering has been the Euclidean distance for inter clustering measurement and the UPGMA (Unweighted Pair-Group Method with Arithmetic mean) as the measure for intra clustering. We apply double hierarchical clustering, to the method associations (row dendrogram) and to the differential profiles (column dendrogram). One row/column clustering set of cells suggest that a profile or group of profiles can be identified

by several methods (see Fig. 3.11).

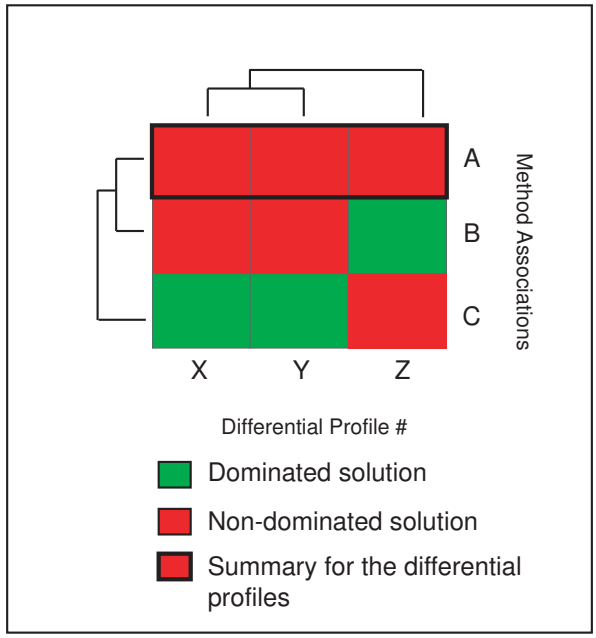


Figure 3.11: Hierarchical double clustering. The set of differential profiles grouped by the clustering X and Y are optimally retrieved by method association A.

The hierarchization of the association rules provides a much smaller set of rules at the cost of not guaranteing the maximum optimality of the association rule chosen in terms of the coefficient C_i, but the solution provided is non-dominated among the others, therefore, even though we can not guarantee its optimality it is at least a candidate for it. We use the double dendrogram cluster partitions of the lookup table, (see Section 1.4.1) of both rows and columns, and combine them by their intersection. These partition provide us with rules with different granularites, from very general rules that can be applied to many differential profiles and englobe many method associations, to very specific rules. We see in Fig. 3.11 a sample of this. Differential profiles X, Y and Z are clustered together, and they have in common method association A to retrieve them. Differential profiles X and Y are also optimally retrieved by method association B, and Z by C. Therefore, we have the following rules extracted from our sample lookup table:

- $R_1 : IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_A$

- $R_2: IF\ DP_X(P_{TX}, P_{CX}, G_X)\ THEN\ M_B$
- $R_3: IF\ DP_Y(P_{TY}, P_{CY}, G_Y)\ THEN\ M_A$
- $R_4: IF\ DP_Y(P_{TY}, P_{CY}, G_Y)\ THEN\ M_B$
- $R_5: IF\ DP_Z(P_{TZ}, P_{CZ}, G_Z)\ THEN\ M_A$
- $R_6: IF\ DP_Z(P_{TZ}, P_{CZ}, G_Z)\ THEN\ M_C$

We can reduce the number of decision making association rules by creating the rules in relation to the clusters of differential profiles and method associations (i.e., more general rules with higher level of granularity). Therefore, we can summarize rules R_1 to R_6 as only one rule

$R_1: IF\ [DP_X(P_{TX}, P_{CX}, G_X); DP_Y(P_{TY}, P_{CY}, G_Y);$
$DP_Z(P_{TZ}, P_{CZ}, G_Z)]\ THEN\ M_A$
It should be noted that the a posteriori application of a weighted sum of other criteria does not bias the searching space and is not in contradiction with the non dominance a priori approach.

2 Results

We evaluate the behavior of the proposed methodology in the retrieval of each of the differential profiles derived from genes retrieved in our inflammation problem by all available methods (i.e., 29 differential profiles, see Section 1.1). Originally, the database was built based on the inflammatory response patterns, which is based on a very robust microarray experiment (Rubio-Escudero et al., 2005; Romero-Zaliz et al., 2007).

2.1 Results of the Identification of Differential Profiles

Different microarray analysis methods retrieve different results applied over the same set of data (see Chapter 2 Table 2.1), and we conclude that results obtained by different methods are different, and none of the methods subsumes the results obtained by the other methods (see Chapter 2 Table 2.2).

At this point we have to decide which genes (probe sets in our particular problem), from all the sets of genes retrieved by each of the microarray analysis

methods applied, will be used throughout the remaining phases of the methodology proposed. We have decided to keep the maximum number of differentially expressed genes (i.e., we keep the union set of all genes retrieved as differentially expressed by any of the methods). This group is made out of 2155 probe sets, and from now on the work will focussed on them. The decision of using the union and no other consensus measure such as intersection, has been based upon the fact that we are trying to develop an exploratory work, meaning that we are not looking for any specific information but trying to guess how to solve problems with microarray analysis methods in a general trend. Therefore, we are interested in dealing with as much information as possible for developing this task.

The construction of the gene expression profiles associated to the 2155 probe sets, is accomplished using a clustering method, K-means. It partitions the observations of the data into non-overlapping clusters. By application of the *Davies-Bouldin* validity index (Davies and Bouldin, 1979) (Equation 1.1), we have partitioned the treatment dataset 24 groups, therefore obtaining different expression profiles P_T (see Fig. 3.12). Analogously, for the control group we have obtained 8 groups by application of the K-means clustering, conforming a total of 8 different expression profiles P_C in the control group (see Fig. 3.13).

Now that we have obtained the profiles contained in our set of data by classifying the probe sets in condition correlated groups, we identify differential profiles (P_{T_m}, P_{C_n}, G) by application of the coincidence index (CI) with a p-value of 0.05 . We obtain a 29 differential profiles, (i.e., 29 different triplets (P_{T_m}, P_{C_n}, G)) that are present in our inflammation dataset (see Fig. 3.14 and Fig. 3.15).

2.2 Results of the Creation of Method Association

The decision making association rules between the 29 differential profiles and the optimal method associations to retrieve them are created based on the evaluation of the behavior of the method associations with each of the 29 differential profiles. The methods associations are created as all potential combinations of microarray analysis methods $M_1, \ldots, M_{10}$, using the $\cup$ and $\cap$ operators, conforming a space of potential hypotheses which can be represented as a lattice structure, showing the hypotheses from the most restrictive at the top (intersection of all methods) to the most general at the bottom (union of all methods), 1024 combinations of methods (see Fig. 3.16).

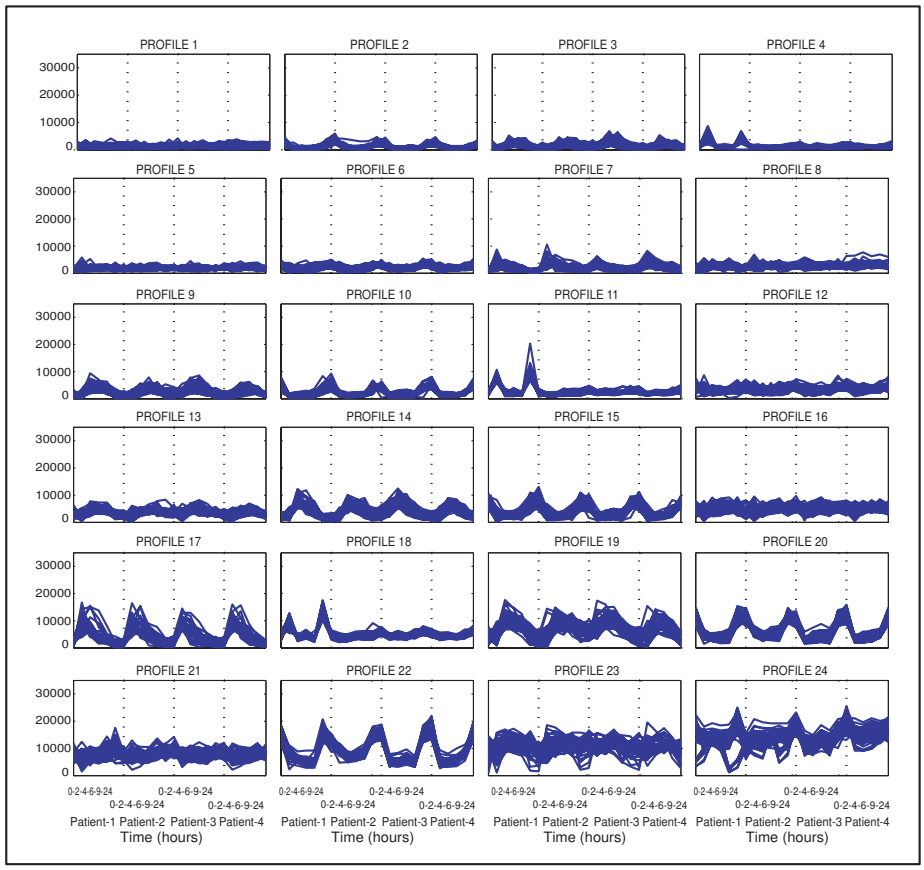


Figure 3.12: The 24 profiles P_T extracted from the treatment group.

2.3 Results of the Evaluation of Method Association Performance

We have evaluated the performance of each potential method associations of $M_1, M_2, \ldots, M_{10}$ (see Fig.3.4) over the set of differential profiles (P_{T_m}, P_{C_n}, G) extracted from the inflammation problem (see Fig. 3.14 and Fig.3.15). The evaluation has been based on a multiobjective procedure based on the maximization of three objectives: specificity (see Equation 3.6), sensitivity (see Equation 3.7) and cost (see Equation1.3), which takes normalized values between [0-1]. We see in Table 3.2 the results of the multiobjective optimization. This table is a summary table of the optimal results obtained for each of the 29 differential

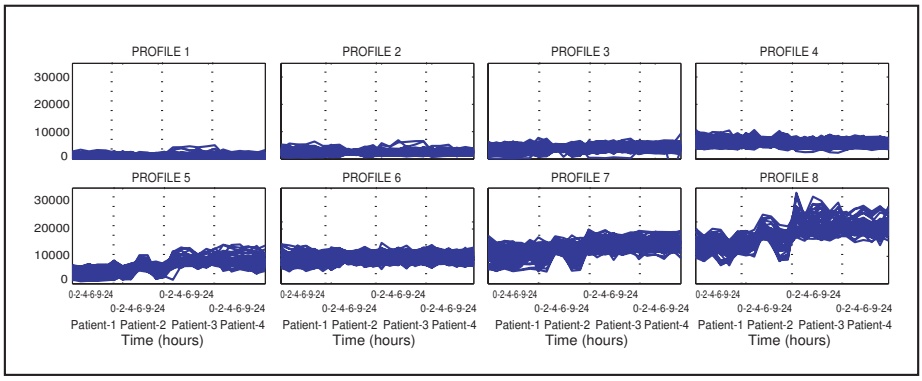


Figure 3.13: The 8 profiles P_C extracted from the control group.

profiles. The second column contains the number of method associations which resulted optimal for each differential profile. The other three columns show a summary of the specificity, sensitivity and cost levels obtained by any of these solutions, for legibility purposes. For example, #15, is optimally retrieved by 3 different method associations M_2, $M_2 \cup M_{10}$ and $M_2 \cup M_4$, with specificity, sensitivity and cost values $(0.0158028, 0.925926, 0.9)$, $(0.0151339, 0.962963, 0.8)$ and $(0.0157112, 0.955674, 0.8)$ respectively. For this particular differential profile, the table shows the best specificity value obtained, (0.0158028) from M_2, the best sensitivity value obtained (0.962963) from $M_2 \cup M_4$, and the best cost value obtained, (0.9) from M_2. The best cost value will always be 0.9 representing a non-dominated solution by application of only one method.

We have graphically represented this information (behavior of the method associations with the differential profiles) a lookup table (see Fig.3.17) for the union operator. The method associations are in the rows and the differential profiles in the columns. Red cells denote that the method association in the row is optimal to retrieve the differential profile in the column (i.e., such method association is a non-dominated solution from our lattice of potential hypotheses).

We can see how some of the differential profiles are easily retrieved by many methods associations using the $\cup$ operator, as it is the case of profile #17 (see Fig. 3.15), which is is retrieved by 13 different associations of methods with a non-dominance relation in the objective values (see Fig. 3.17), while other differential profiles, like #15 (see Fig. 3.15), are only retrieved by 2 method associations. These methods have a non-dominance relation to each other, meaning that each of them is sufficient by itself to recover the profile but emphasizing different objectives such as sensitivity, specificity or cost (non-dominated so-

Profile	# Optimal Rules	Best Specificity	Best Sensitivity	Best Cost
#1	16	0.0966562	0.994819	0.9
#2	3	0.0925926	0.992537	0.9
#3	10	0.247967	1	0.9
#4	10	0.2	1	0.9
#5	5	0.0343137	0.953488	0.9
#6	5	0.0472362	0.989796	0.9
#7	10	0.0631934	0.977612	0.9
#8	12	0.0514184	0.988764	0.9
#9	5	0.053448	0.964286	0.9
#10	4	0.0566667	1	0.9
#11	5	0.0515717	0.963964	0.9
#12	6	0.1	0.966667	0.9
#13	3	0.0413043	1	0.9
#14	4	0.066474	0.943396	0.9
#15	2	0.0158028	0.962963	0.9
#16	3	0.0547677	0.982456	0.9
#17	13	0.0531915	0.988636	0.9
#18	4	0.103321	1	0.9
#19	9	0.151515	0.972973	0.9
#20	1	0.0448179	1	0.9
#21	2	0.0582011	1	0.9
#22	2	0.0237154	1	0.9
#23	2	0.107692	1	0.9
#24	4	0.0653266	1	0.9
#25	2	0.0819672	1	0.9
#26	7	0.056926	1	0.9
#27	2	0.0802469	1	0.9
#28	5	0.0458221	1	0.9
#29	3	0.0538201	1	0.9

Table 3.2: Number of Optimal Rules for each differential profile using the $\cup$ operator and summary of the specificity, sensitivity and cost values obtained.

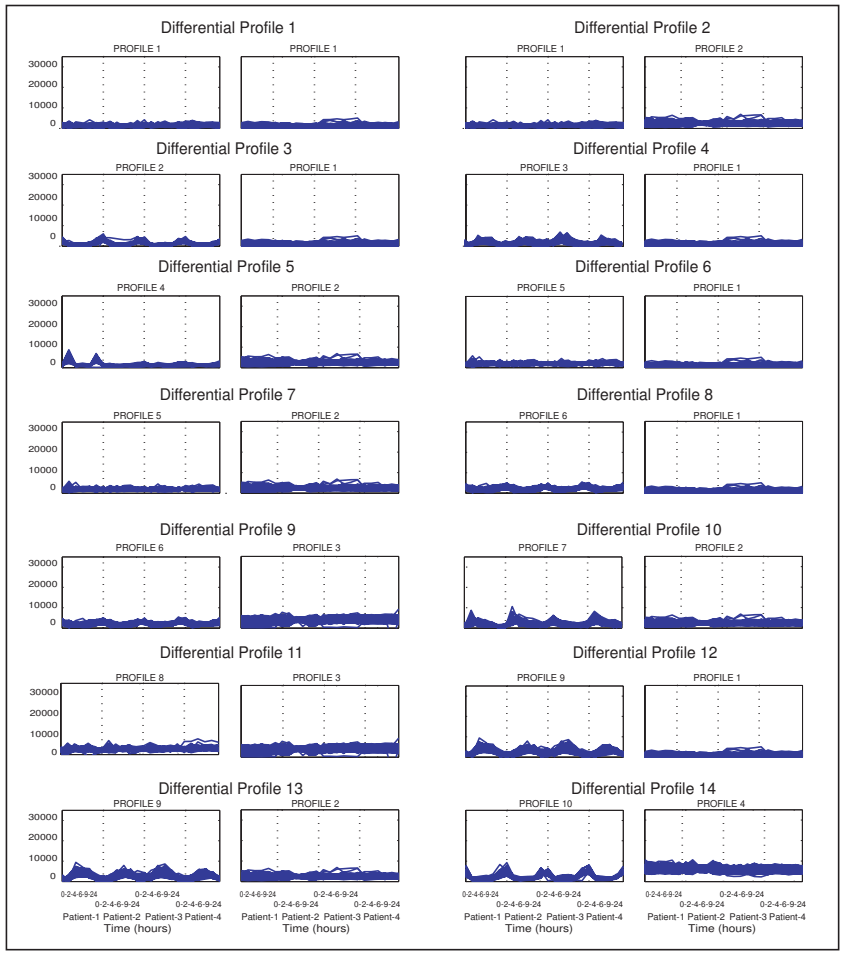


Figure 3.14: Differential profiles (P_{T_m}, P_{C_n}, G) 1 to 14 from the inflammation data problem.

lutions, Section 1.3). Regarding the optimization objectives, we see that the associations of methods obtain the best sensitivity levels possible, 1 or almost 1 for all the differential profiles. For the cost objective, (i.e., the number of applied methods), some differential profiles achieve optimal scores in the multiobjective evaluation with a small number of methods, meaning that application of more methods over the data set will not improve the retrieval of such profile (e.g., profile #15 reaches its optimal with an association of only 2 methods). As we said, our method is not fully exhaustive.

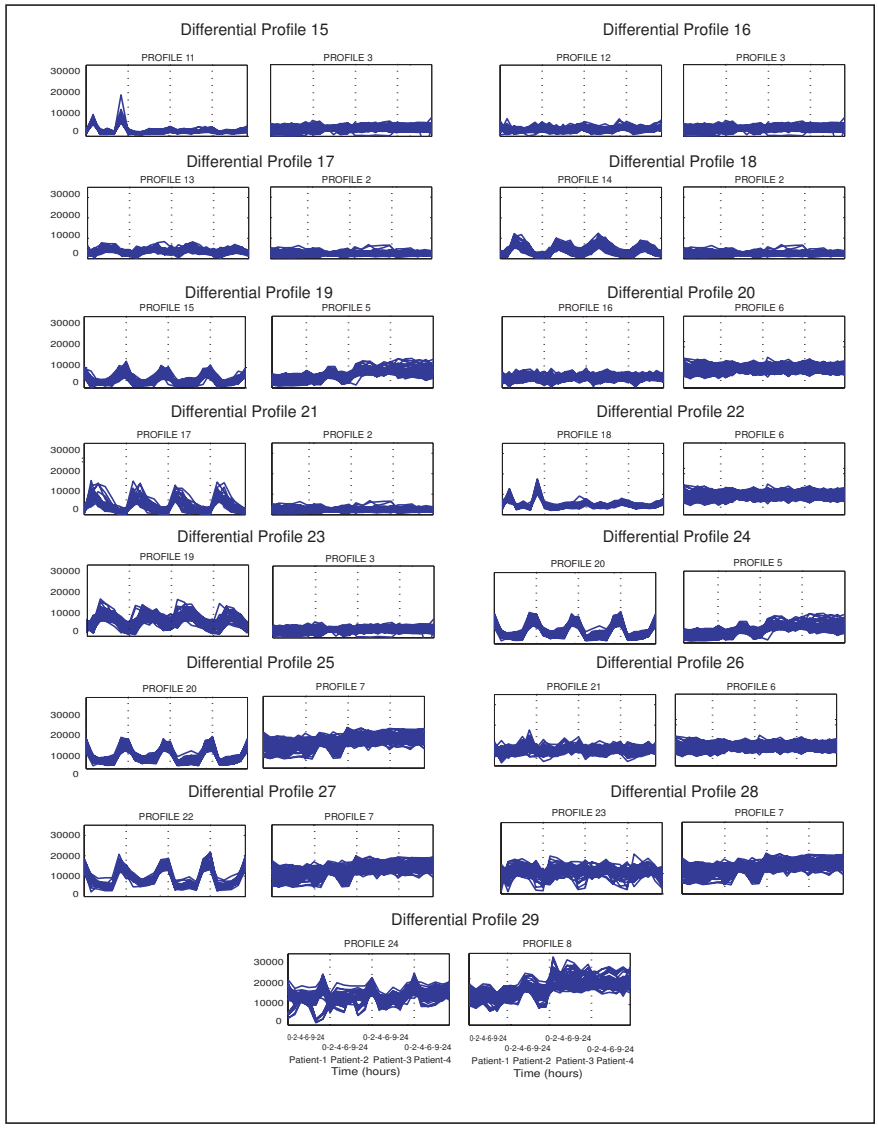


Figure 3.15: Differential profiles (P_{T_m}, P_{C_n}, G) 15 to 29 from the inflammation data problem.

The search over the lattice of potential hypotheses for a certain differential profile stops when increasing the cost objective (number of methods associated) does not result in an improve of any of the two other objectives, specificity and

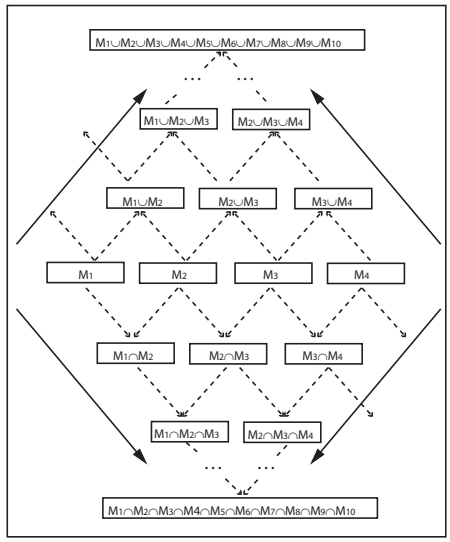


Figure 3.16: Lattice of potential hypotheses, method associations $M_1, \ldots, M_{10}$ using the $\cup$ and $\cap$ operators. The solid arrows show the direction of the search in the space of hypotheses.

sensitivity. Some of the differential profiles are easy to retrieve by application of any of the methods, obtaining good levels or specificity and sensitivity, as it is the case of differential profiles #3 and #4 #17 (see Fig. 3.14). Some others, like profiles #5 or #15, are only successfully retrieved by associations of methods including method M_2, Student's T-test considering time. Profiles #5 and #15 exhibit different levels of expression in the samples throughout the treatment group, that is why methods which take into account the time condition, in particular Student's T-test considering time, are capable to retrieve them. The same situation applies to differential profile #22, which also exhibits a different behavior of the samples in the treatment group throughout time, but is more easily retrieved by other associations of methods since the expression levels in the samples behaving different are higher and therefore easier to statistically detect. We can also see how in general, differential profiles #18 to #29 are easier to generally retrieve by all methods, since those profiles conglomerate the genes with higher levels of expression, which are therefore easier to retrieve by methods in general. Differential profiles which comprise genes with lower levels of expression, are generally well retrieved my methods M_2, Student's T-Test considering time and M_5, Analysis of variance over the treatment vs. control

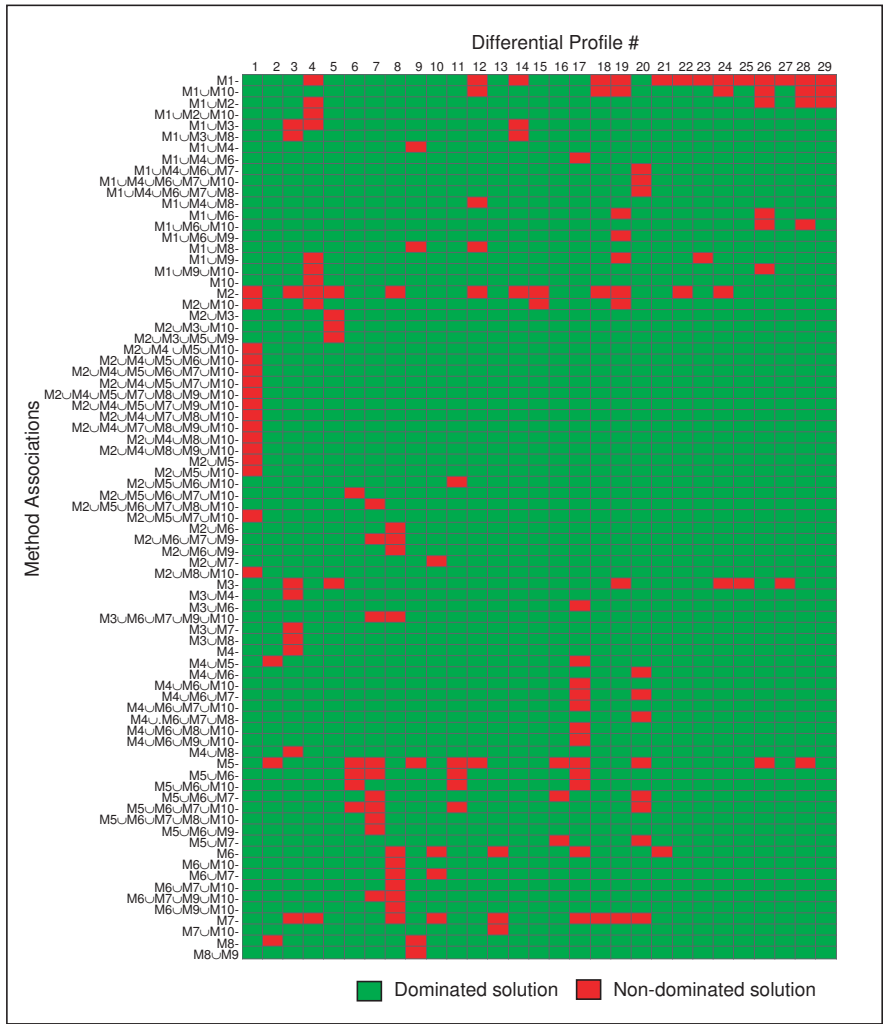


Figure 3.17: Lookup table for the optimal method associations using the ∪ operator in retrieving the differential profiles. Red cells denote that the method association in the row is optimal to retrieve the differential profile in the column condition.

The table of values and the lookup table for the intersection operator can be found in Appendix A in Table 4 and Fig.10 respectively.

2.4 Results of the Creation of Decision Making Association Rules

In Section 1.2 we have described how to associate the microarray analysis methods so that we can combine the advantages of each method to extract as much information as possible related to the differential profiles (P_{T_m}, P_{C_n}, G). This information is stored into decision making association rules, which will provide for each differential profile the optimal method, or association of methods, to retrieve it. Therefore, we create a set of decision making association rules for each of the differential profiles available in the database, based on the multi-objective optimization results shown in Table 3.2. We obtain a total number of 518 association rules. We focus on the 169 of them (see Fig. 3.17) which are obtained from associating the methods using the union operator and the non-dominance criterion.

2.4.1 Identification of Single-profile Association Rules

We propose a decision making mechanism to summarize the 169 association rules obtained from direct scanning of the lookup table (see Fig. 3.17), as association rules with individual antecedent and consequent. Some sample rules obtained from direct scanning of the look up table are:

- $R_1 :\ IF\ DP_5(P_{T5}, P_{C5}, G_5)\ THEN\ M_1$
- $R_2 :\ IF\ DP_5(P_{T5}, P_{C5}, G_5)\ THEN\ M_1 \cup M_{10}$
- ...
- $R_{169} :\ IF\ (DP_{13}(P_{T13}, P_{C13}, G_{13})\ THEN\ M_5$

To do so, we create a set of association rules for individual profiles profiles by implementing the coefficient C_i in terms of a weighted sum operator for the three objectives evaluated: specificity, sensitivity and cost. Such operator is implemented as:

$$WS = \frac{(\omega_1 \times O_1)\ +\ (\omega_2 \times O_2)\ +\ (\omega_3 \times O_3)}{3} \tag{3.10}$$

where O_1 refers to specificity, O_2 to sensitivity and O_3 to cost. The weights assigned have been $\omega_1 = 0.30$, $\omega_2 = 0.55$ and $\omega_3 = 0.15$. The sensitivity has been favored in the weighted sum since it is the objective that we are more

interested in to get a wider picture of the behavior of the genes in our current problem. From such calculation we obtain the results in Table 3 in Appendix A for each of the method associations retrieved as non-dominated for any of the profiles. The resulting lookup table, with the method associations with maximum C_i value for the implemented operator highlighted with a black line can be seen in Fig. 3.18. Based on this optimization approach, we end up with 29 optimal association rules based on the selected C_i, which are listed in Table 3 in Appendix A.

2.4.2 Identification of Multiple-profile Association Rules

We can group some of the rules into multiple-profile decision making rules. To do so, we explore the lookup table in Fig. 3.18. Differential profiles are optimally retrieved by as many method associations as rows we can find containing a highlighted method association (19 rows in total). The summarization of these rules by using the C_i measurement allows us to identify sets of differential profiles being retrieved with maximum C_i value a method associations, (i.e., differential profiles in the antecedent share a common method association with maximum C_i in the individual profile evaluation). That is the case of differential profiles #26, #28 and #29 which are retrieved with maximum C_i values by $M_1 \cup M_2$ (see Fig. 3.18). Their association rules can be written as

- $R_l : IF\ [DP_{26}(P_{T26}, P_{C26}, G_{26}); DP_{28}(P_{T28}, P_{C28}, G_{28}); DP_{29}(P_{T29}, P_{C29}, G_{29})]\ THEN\ M_1 \cup M_2$

If we were interested in retrieving a set containing genes from differential profiles which do not have a common consequent with maximum C_i from the rules they have associated, we need to reevaluate the behavior of the method associations since optimality in the non-dominance relation can not be guaranteed when combining consequents. We retrieve a set of 14 different differential profiles, say #2, #4, #5, #9, #11, #13, #15, #18, #20, #21, #22, #23, #25, #29. We can see from the lookup table (Fig. 3.18) that there are no rules for these 14 profiles with common method associations in the consequent. Note that in this case, we want to identify a the set of genes in our dataset belonging to any of the 14 differential profiles listed. Therefore, we reevaluate the behavior of the lattice in retrieving this set of profiles. There are 24 non-dominated solutions, (i.e., 24 optimal method associations to retrieve the set of 14 differential profile). The results are shown in Table 3.3.

Method Associations	Specificity	Sensitivity	Cost
M_2	0.67952	0.780683	0.9
$M_2 \cup M_9$	0.666859	0.83878	0.8
$M_2 \cup M_{10}$	0.668801	0.834423	0.8
$M_2 \cup M_6 \cup M_9$	0.65564	0.877996	0.7
$M_2 \cup M_6 \cup M_{10}$	0.654781	0.880174	0.7
$M_2 \cup M_9 \cup M_{10}$	0.664234	0.859114	0.7
$M_2 \cup M_3 \cup M_8 \cup M_{10}$	0.651828	0.893246	0.6
$M_2 \cup M_3 \cup M_9 \cup M_{10}$	0.658096	0.873638	0.6
$M_2 \cup M_6 \cup M_8 \cup M_{10}$	0.647299	0.922295	0.6
$M_2 \cup M_6 \cup M_9 \cup M_{10}$	0.654071	0.89252	0.6
$M_2 \cup M_8 \cup M_9 \cup M_{10}$	0.65142	0.899782	0.6
$M_2 \cup M_3 \cup M_6 \cup M_8 \cup M_{10}$	0.645551	0.932462	0.5
$M_2 \cup M_3 \cup M_8 \cup M_9 \cup M_{10}$	0.650052	0.909223	0.5
$M_2 \cup M_6 \cup M_8 \cup M_9 \cup M_{10}$	0.646434	0.928105	0.5
$M_2 \cup M_3 \cup M_6 \cup M_8 \cup M_9 \cup M_{10}$	0.645177	0.937545	0.4
$M_3 \cup M_4 \cup M_5 \cup M_6 \cup M_8 \cup M_9$	0.640526	0.954975	0.4
$M_1 \cup M_3 \cup M_4 \cup M_5 \cup M_6 \cup$ $M_8 \cup M_9$	0.640389	0.955701	0.3
$M_2 \cup M_3 \cup M_4 \cup M_5 \cup M_6 \cup$ $M_8 \cup M_{10}$	0.639793	0.985476	0.3
$M_2 \cup M_3 \cup M_6 \cup M_7 \cup M_8 \cup$ $M_9 \cup M_{10}$	0.642152	0.944808	0.3
$M_3 \cup M_4 \cup M_5 \cup M_6 \cup M_8 \cup$ $M_9 \cup M_{10}$	0.64	0.964415	0.3
$M_1 \cup M_3 \cup M_4 \cup M_5 \cup M_6 \cup$ $M_8 \cup M_9 \cup M_{10}$	0.639865	0.965142	0.2
$M_2 \cup M_3 \cup M_4 \cup M_5 \cup M_6 \cup$ $M_8 \cup M_9 \cup M_{10}$	0.639529	0.986928	0.2
$M_3 \cup M_4 \cup M_5 \cup M_6 \cup M_7 \cup$ $M_8 \cup M_9 \cup M_{10}$	0.639829	0.975309	0.2
$M_2 \cup M_3 \cup M_4 \cup M_5 \cup M_6 \cup$ $M_7 \cup M_8 \cup M_9 \cup M_{10}$	0.638149	0.991285	0.1

Table 3.3: Objectives values for the non-dominated solutions in the lattice of potential methods associations to retrieve a set of 14 differential profiles (#2, #4, #5, #9, #11, #13, #15, #18, #20, #21, #22, #23, #25, #29).

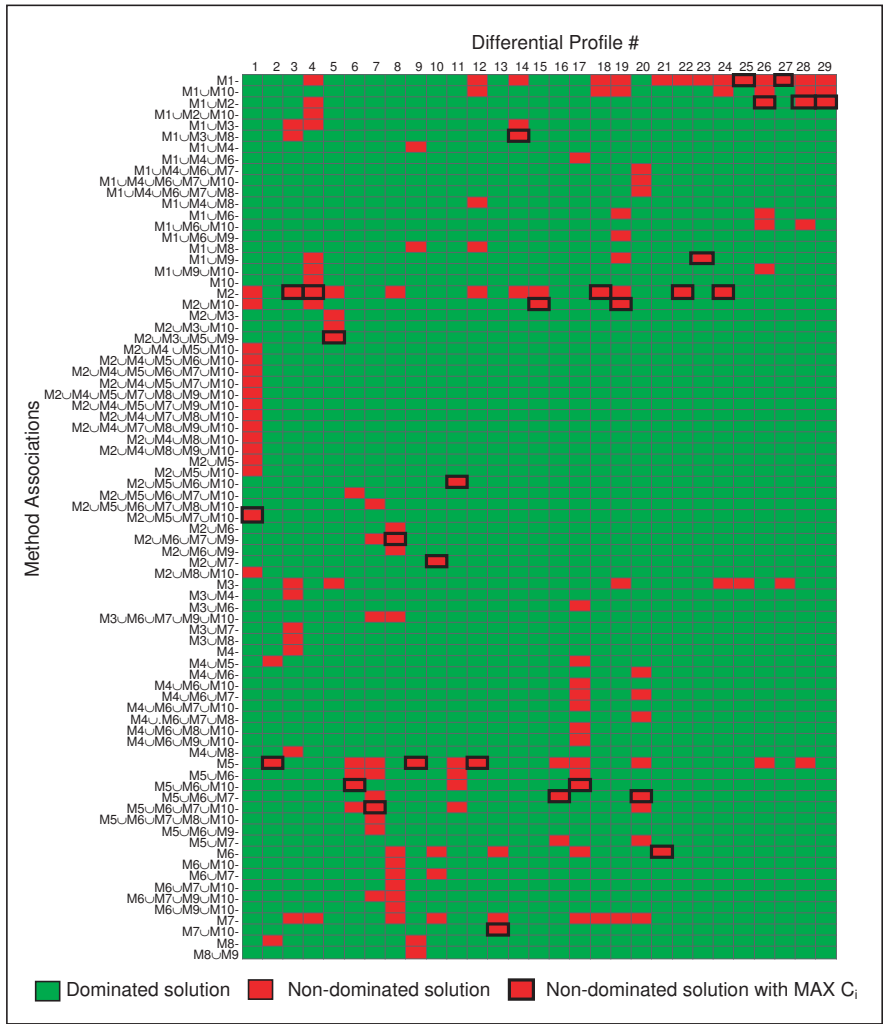


Figure 3.18: Lookup table with the 29 optimal rules

2.5 Hierarchical Association Rules

We can summarize and reduce the rule base complexity by clustering their scope. This provides a minimal set of rules with maximum coverage of antecedents (differential profiles). We apply double hierarchical clustering to the lookup table in order to obtain intersections areas between method associations and

sets of differential profiles. The results can be seen in Fig. 3.19

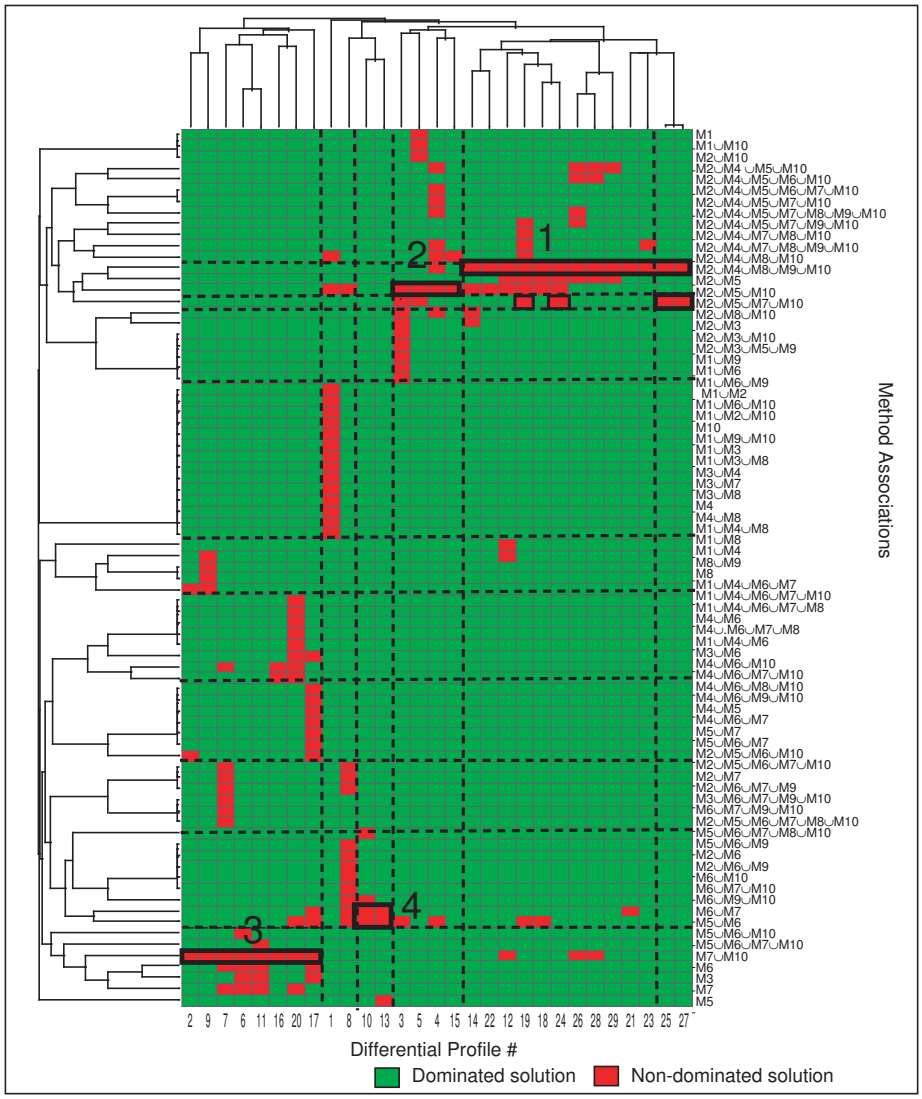


Figure 3.19: Results of applying hierarchical clustering to the relations between differential profiles and method associations using the ∪ operator. Groups of profiles that can be retrieved with a single method association are denoted with numbers.

From the dendrograms we can extract some conclusions relating the method

associations and the differential profiles, or sets of differential profiles they can optimally retrieve. Differential profiles #2, #6, #7, #9, #11, #16, #17 and #20 can be optimally retrieved by association of methods $M_7 \cup M_{10}$ (denoted as 3 in Fig. 3.19). These differential profiles move in low levels of variation between time points in the treatment group, with values fluctuating between 0 and 10000, and values in similar to the treatment profiles. They are shown in Fig. 3.20. The methods capable to retrieve the former differential profiles, M_7, ANOVA over time and treatment, and M_{10} Repeated Measures ANOVA (RANOVA) over time and treatment, are highly sensitive to information related to the variation between time points: these profiles are ignored by method focussed on finding changes between just the treatment and the control condition, since difference in levels of expression are too weak.

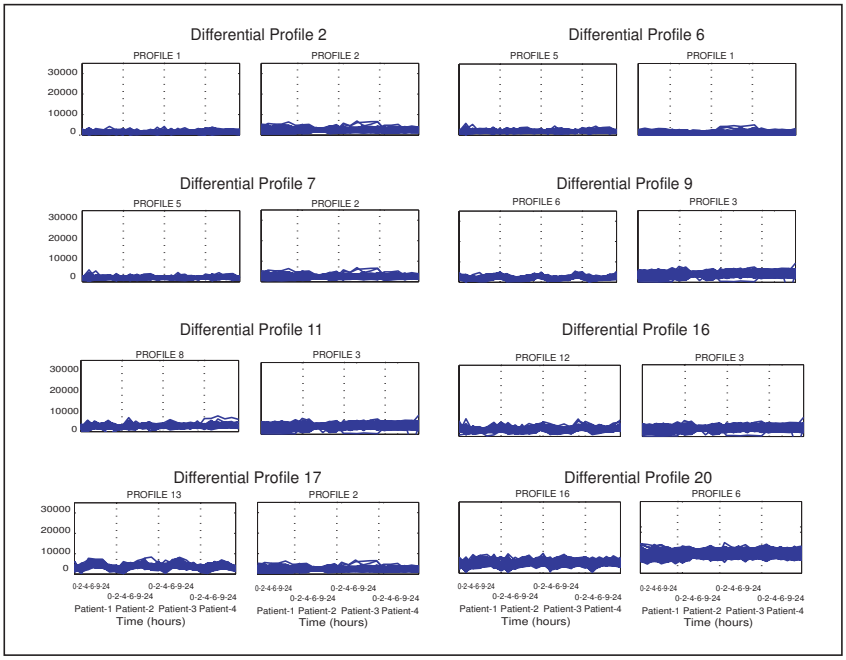


Figure 3.20: Differential profiles #2, #9, #7, #6, #11, #16, #17 and #20. These profiles have some levels of variation among time points in the treatment group, with values fluctuating between 0 and 10000, and values in control also fluctuating among those data.

We also see in Fig. 3.19 two differential profiles, #10 and #13, which are optimally retrieved either by $M_5 \cup M_6$ and by $M_6 \cup M_7$ (denoted as 4 in in Fig. 3.19). The differential profiles share expression levels in the treatment group

changing from 0 to almost 10000 at times 2 and 4, and then going back to value 0 at time 24. The values in the control group are close to 0 (see Fig. 3.21). The ANOVA family, over time (M_6) complemented by ANOVA over treatment (M_5) or ANOVA over time and treatment are optimal to retrieve this type of change.

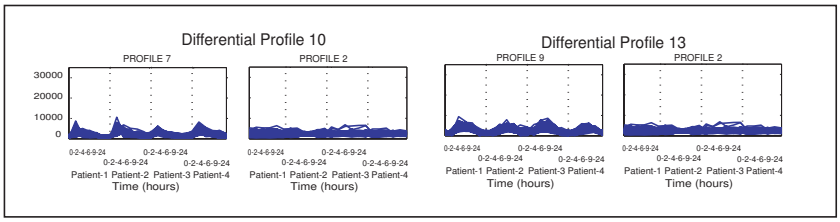


Figure 3.21: Differential profiles #10 and #13. These profiles have expression levels in the treatment group changing from 0 to almost 10000 at times 2 and 4, and then going back to value 0 at time 24.

From the hierarchical cluster image we can also see differential profiles #3, #5, #4 and #15, which are optimally recovered by method association $M_2 \cup M_5$ (denoted as 2 in in Fig. 3.19). From these differential profiles, it is noteworthy that #5 and #15 are two of the three differential profiles from the inflammation and host response to injury problem, exhibiting different behavior among patients in the treatment group, (i.e., see differences between patient one and the other three patients). The third profile with such characteristic is #22 (see Fig. 3.22), to which the method association $M_2 \cup M_5$ also results optimal. Profiles #5 and #15 exhibit different levels of expression in the samples throughout the treatment group, that is why methods which take into account the time condition, in particular Student's T-test considering time, complemented by ANOVA over treatment, M_5, are capable to retrieve them. The same situation applies to differential profile #22, which also exhibits a different behavior of the samples in the treatment group throughout time. This profile is more easily retrieved by other associations of methods since the expression levels in the samples behaving different are higher and therefore easier to statistically detect, and therefore, grouped in another region of the clustering.

Regarding profiles #14, #22, #12, #19, #18, #24, #26, #28, #29, #21, #23, #25, #27, they are all grouped together and being optimally retrieved by method association $M_2 \cup M_4 \cup M_8 \cup M_9 \cup M_{10}$ (denoted as 1 in in Fig. 3.19). These differential profiles have in common the high level of expression fluctuation they exhibit (see Fig. 3.15). From the set of profiles we can extract a subgroup, differential profiles #19, #24, #25 and #27 which have in common decreasing their level of expression at times 2 and 4 in relation to their values

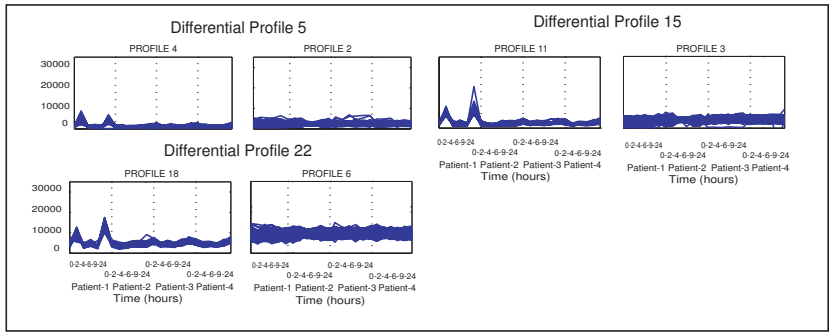


Figure 3.22: Differential profiles #5, #15 and #22. These profiles exhibit different behavior among patients in the treatment group, in particular between patient one and the other three patients.

at times 0 and 24 (see Fig. 3.23). This subgroup is optimally retrieved by a smaller association of methods, $M_2 \cup M_5 \cup M_7 \cup M_{10}$.

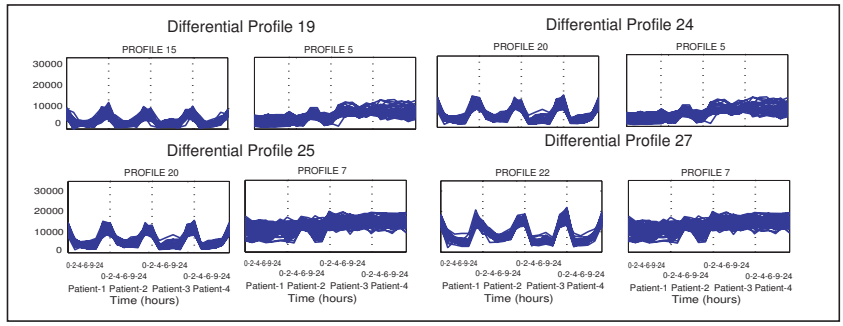


Figure 3.23: Differential profiles #19, #24, #25 and #27. These differential profiles have in common the high level of expression fluctuation they exhibit.

We can deduce some other summary rules, from Fig. 3.19 such as

- R_l : $IF\ [DP_{10}(P_{T10}, P_{C10}, G_{10}); DP_{13}(P_{T13}, P_{C13}, G_{13})]\ THEN\ M_5 \cup M_6$
- R_m : $IF\ [DP_3(P_{T3}, P_{C3}, G_3); DP_4(P_{T4}, P_{C4}, G_4); DP_5(P_{T5}, P_{C5}, G_5);$ $DP_{15}(P_{T15}, P_{C14}, G_{15})]\ THEN\ M_2 \cup M_5 \cup M_{10}$
- R_m : $IF\ [DP_{14}(P_{T14}, P_{C14}, G_{14}); DP_{18}(P_{T18}, P_{C18}, G_{18});$ $DP_{19}(P_{T19}, P_{C19}, G_{19}); DP_{21}(P_{T21}, P_{C21}, G_{21}); DP_{22}(P_{T22}, P_{C22}, G_{22});$

$DP_{23}(P_{T23}, P_{C23}, G_{23}); DP_{24}(P_{T24}, P_{C24}, G_{24}); DP_{25}(P_{T25}, P_{C25}, G_{25});$
$DP_{26}(P_{T26}, P_{C26}, G_{26}); DP_{27}(P_{T27}, P_{C27}, G_{27}); DP_{28}(P_{T28}, P_{C28}, G_{28});$
$DP_{29}(P_{T29}, P_{C29}, G_{29})]\ THEN\ M_2 \cup M_4 \cup M_8 \cup M_9 \cup M_{10}$

2.6 Comparison with Classical Application of Methods

Once we have made a general description of the results obtained by application of method associations, we compare how this technique behaves in comparison with the classical use of microarray analysis methods, that is to say, with the application of the classical microarray methods (i.e., M_1-Student's T-test, M_2-Permutation Test, ... methods M_1 to M_{10}) applied in isolation, as described in Chapter 2 Section 2.

Regarding the specificity and sensitivity objectives, the optimal association rules created by our methodology overcome the behavior of the individual application of methods in the 29 evaluated differential profiles. For example, profile #19 is optimally recovered by a set of 44 different associations of methods which are non-dominated in their specificity, sensitivity and cost values. We compare the sensitivity and specificity values with the specificity and sensitivity values of the 10 microarray analysis classical methods applied individually. We see in Fig. 3.24 how the specificity and sensitivity levels of each of the 44 optimal associations of methods are better (i.e., closer to 1) than any of the specificity and sensitivity levels of the methods applied on their own.

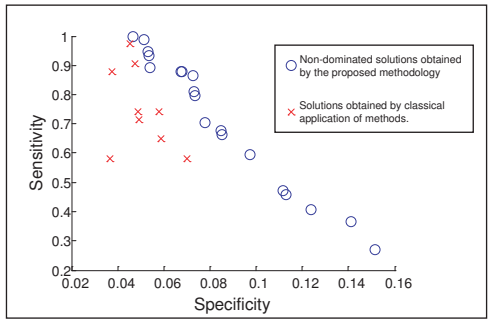


Figure 3.24: Comparison of specificity and sensitivity levels for differential profile #19.

The values obtained with our methodology for $(specificity, sensitivity)$ rank from $(0.05, 0.99)$ on the one side of the Pareto optimal front to $(0.15, 0.47)$ on the other side, with an average value of $(0.084, 0.88)$, while the values from the

methods applied in isolation rank from $(0.045, 0.78)$ to $(0.070, 0.58)$ with average value of $(0.052, 0.76)$. The set of non-dominated solutions in the three objectives (Pareto-optimal front) retrieved by our methodology for genes belonging to one of the differential profiles, #19 in particular, are shown in Fig. 3.25 .

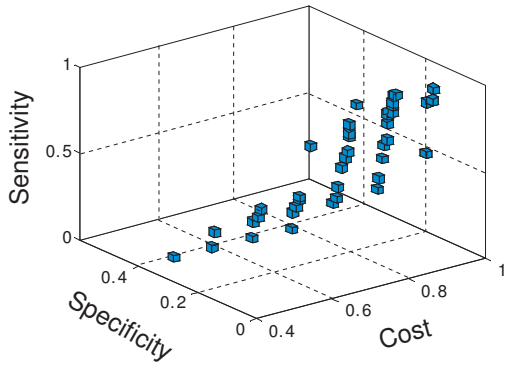


Figure 3.25: Pareto-optimal front for differential profile #19.

We also show a comparison of the behavior of our methodology against the classical application of microarray analysis methods in the retrieval of more than one differential profile simultaneously. We recall the example from Section 2.4 where profiles #2, #4, #5, #9, #11, #13, #15, #18, #20, #21, #22, #23, #25, #29 are simultaneously retrieved. Consistently, our results achieved better specificity and sensitivity than the results of applying the methods in isolation (Fig. 3.26). The values obtained with our methodology rank from $(0.638149, 0.991285)$ on the one side of the Pareto-optimal front to $(0.67952, 0.780683)$ on the other side, with an average value of $(0.65, 0.89)$, while the values from the methods applied in isolation rank from $(0.59, 0.78)$ to $(0.65, 0.48)$ with an average value of $(0.61, 0.49)$.

The application of our methodology helps to alleviate the problems exhibited by individual methods, including missing important profiles or genes belonging to such profiles, and oppositely, recovering non useful profiles. Our approach retrieves probe sets with related behavior to other probe sets with already known profiles, which might have similar functionalities (Tavazoie et al., 1999). For instance, probe set 206011_at (genes CASP1) is related both in behavior and in function (apoptosis-related cysteine peptidase) to probe sets 211367_s_at (CASP1) and 211368_s_at (CASP1) (Fig. 3.27), stated as relevant for the inflammation problem in (Calvano et al., 2005). The isolated application of classical methods such as M_1 or M_3 with the default p-value and false discovery rate would not retrieve such probe set as differentially expressed. We retrieve

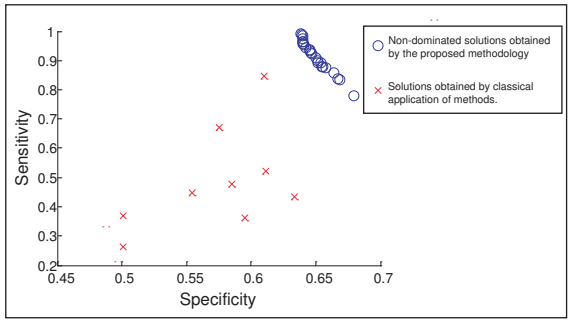


Figure 3.26: Comparison of specificity and sensitivity levels for 14 out the the 29 differential profiles.

206011_at applying the method aggregation $M_7 \cup M_10$ with values $(1, 0.25, 0.8)$ for sensitivity, specificity and cost respectively. The same situation applies to probe sets 202076_at (BIRC2) and 210538_s_at, (BIRC3) related both in behavior and in function (inhibitor of apoptosis protein 2 and 1 respectively) (Fig. 3.28). Probe set 202076_at is retrieved applying the methods $M3 \cup M6$ with values (0.35 ,0.93, 0.8).

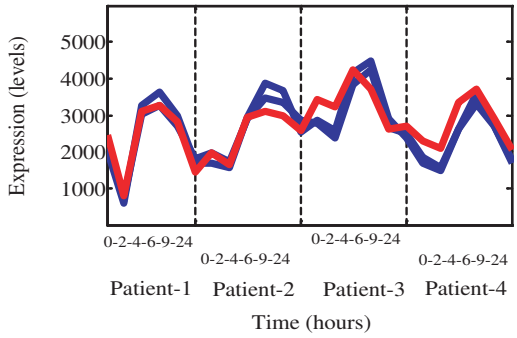


Figure 3.27: Probe sets in blue are stated as relevant for the inflammation problem in (Calvano et al., 2005). Probe sets in red are retrieved applying our methodology but not applying classical microarray analysis methods individually.

We also find a time-dependence between the probe sets related to gene CASP1 (209970_x_at, 211366_x_at, 211367_s_at and 211368_s_at) and the probe set related to BIRC2 (202076_at) (Fig. 3.29). The expression throughout time of CASP1 related probe sets is regulated by the expression of the BIRC2 probe

set. This regulation dependence is confirmed by the Ingenuity Pathway Assist tool (http://www.ingenuity.com/). This software confirms that CASP1 has been described in the literature as regulated by BIRC2, which are included in a pathway related to death receptor signaling and apoptosis signaling (Fig. 3.30).

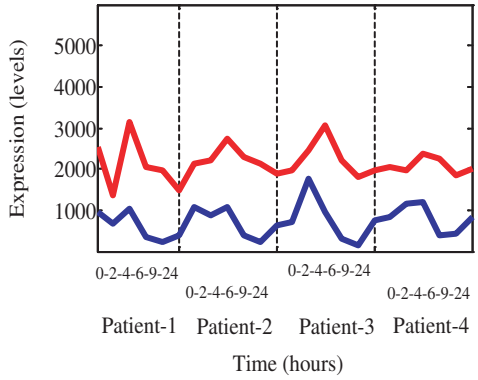


Figure 3.28: Probe sets in blue are stated as relevant for the inflammation problem in (Calvano et al., 2005). Probe sets in red are retrieved applying our methodology but not applying classical microarray analysis methods individually.

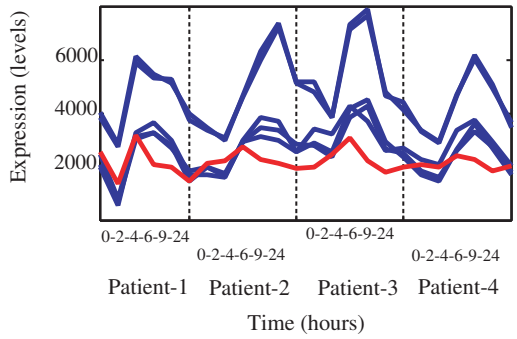


Figure 3.29: Time-dependence between the probe sets related to gene CASP1 (209970_x_at, 211366_x_at, 211367_s_at and 211368_s_at) (blue) and the probe set related to BIRC2 (202076_at) (red). The expression throughout time of CASP1 related probe sets is regulated by the expression of the BIRC2 probe set. High expression peaks in BIRC2 occur before high expression peaks of CASP1 in all four subjects.

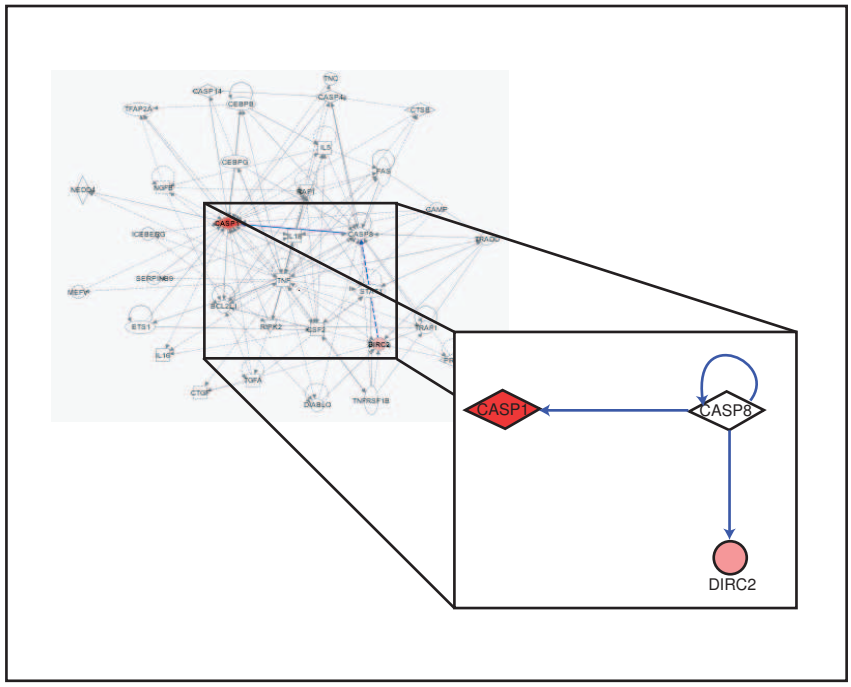


Figure 3.30: Network obtained from Ingenuity Pathway Assist where genes CASP1 and BIRC2 are included. We see how BIRC2 regulates indirectly CASP1 through CASP8. This network is related to a cell death pathway.

3 Validity of the Association Rules

The methodology proposed has the goal the of creating a set of association rules which store the relation between the methods associations and the differential profiles they are able to recover, in an attempt to combine the advantages of the microarray analysis methods to identify all significant probe sets in data from microarray experiments by identifying the differential profiles they exhibit.

The set of decision making association rules obtained based on the inflammation and host response to injury problem, (See Section 2.4) should be generally suitable for any other microarray expression data, not only for the microarray expression data they were extracted from. We can think as the inflammation problem as the training set where we learn the association rules from, and now we need a test set to validate them.

The justification of the usefulness of new microarray analysis methodologies has been on two approaches a) application to experimental datasets where the algorithms were able to recover information that was previously established by independent methods, or b) application to synthetic data sets (Mendes et al., 2003). It is to mention that carrying out microarray experiments is highly expensive, so the amount of microarray datasets available for research purposes is very reduced. Furthermore, a big percentage of the datasets obtained from application of microarray technology is not public. Therefore, *ad-hoc* microarray datasets are artificially generated and used to validate microarray related research. The second approach, which uses artificial data, has the advantage that the process that generated the data is well known and so one is able to judge the success or failure of the algorithm (Mendes, 2003). Artificial datasets generation has been widely applied both in microarray related publications (Barenco et al., 2006; Hakamada et al., 2006), as well as in other general data mining applications (Pargas et al., 1999).

In our particular case, a very complex microarray study has been carried out, a study including 8 human volunteers, with 48 samples taken throughout time with changes studied over time, treatment and patient. Due to the complexity of this study, there are not public microarray datasets which reproduce these experimental conditions. Therefore, we have created an *ad-hoc* artificial dataset to examine the validity of the decision making association rules extracted for the differential profiles. If the results obtained by application of the methodology proposed to this new dataset, (i.e., we obtain a set of rules from each of the differential profiles in the set of artificial data similar to those obtained from the inflammation problem) we will be then able to affirm that the set of rules obtained is valid for microarray data in general.

3.1 Creation of the Artificial Expression Data

The artificial set of expression data has been created based on each of the expression profiles obtained from the inflammation problem: 24 for the treatment group and 8 for the control group. A set of 100 artificial probe sets has been created for each of the profiles based on its centroid. The centroid of each profile from the inflammation dataset has been calculated as the average value of all probe sets at each of the time points of the experiment. To avoid the centroid being biased by any outlier or misclassified probe set, only those probe sets with all their time expression values lying within a distance smaller than three times the size of the standard deviation from the centroid value are taken into account (from the StatSoft Inc. Statistics Glossary, an outlier is defined as “any measurement that falls outside of three standard deviations”).

These probe sets have been created adding some noise to the profile centroids by modifying at each probe set, and at each time point, the centroid by a value randomly chosen from $[-a, +a]$ where a is defined as the maximum centroid value divided by $2sd^{-1}$. Therefore, each probe set related to a profile will be a limited random variation os such profile's centroid value at each time point. Therefore, the creation of the random probe sets for the treatment group is described as:

```
FOR EACH random_profile i FROM 1 TO 24
    FOR EACH probe_set j FROM 1 TO 100
        FOR EACH time_point k FROM 1 TO 6
            a := max (treatment_centroid[i]) / 2sd
            val := treatment_centroid[i][k] + random(-a,+a)
            probe_set[j][k] := val
```

The probe sets for the control group profiles have been created following the same algorithm. In Fig. 3.31 and Fig. 3.32 we show the probe sets created for each of the 24 and 8 expression profiles from the treatment and control group respectively. We apply to this artificial dataset each of the methodology steps proposed in Section 1 and applied to the inflammation and host response to injury dataset in Section 2. For the sake of generality, we do not consider differences among patients, as it is often the case that other microarray experiments do not have as many samples as we do in this study.

3.2 Results of the Identification of Differential Profiles in the Artificial Dataset

The randomly generated probe sets have been created based on the 24 treatment and 8 control profiles from the inflammation and host response to injury problem. The differential profiles have been created following the same profile combination as in the inflammation as host response to injury dataset (i.e., differential profile #1 is composed of treatment profile #1 and control profile #1, differential profile #2 is composed of treatment profile #1 and control profile #2, differential profile #3 is composed of treatment profile #2 and control profile #1, see Fig. 3.33 and Fig. 3.34).

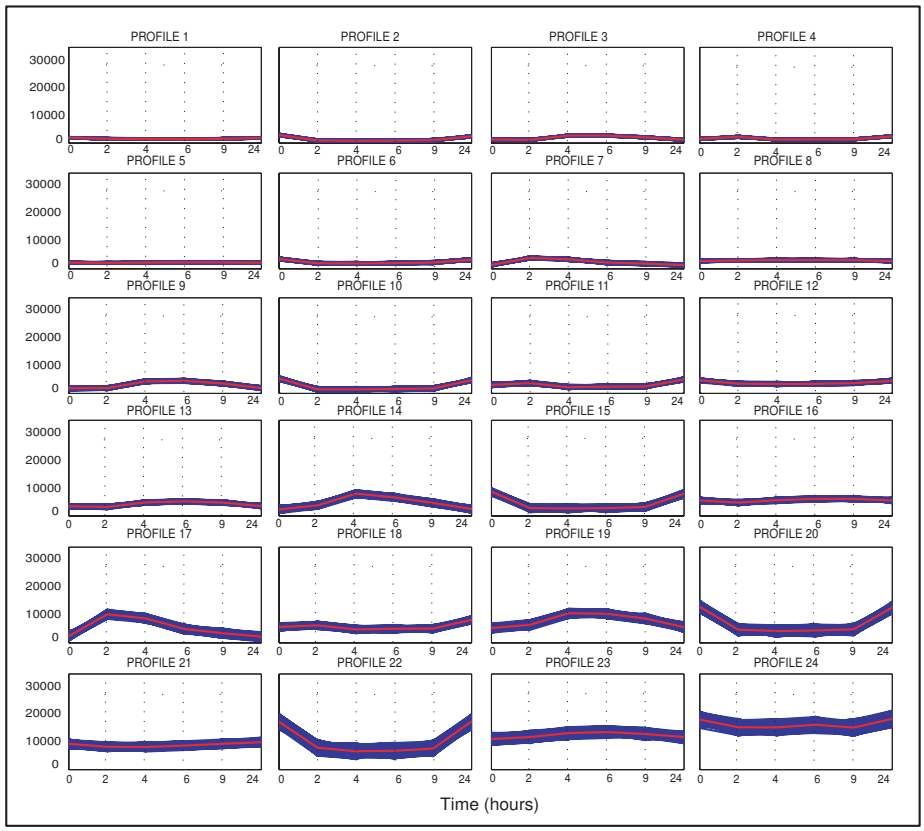


Figure 3.31: Random profiles generated for the treatment group. The centroids are highlighted in red.

3.3 Results of the Evaluation of Method Association Performance in the Artificial Dataset

In order to asses that the association rules created from the inflammation data set are generally suitable for any other microarray expression data, not only for the microarray expression data they were extracted from, we evaluate the behavior of the methods associations over the artificial set of expression data created. The evaluation is performed with the rules generated from the inflammation and host response to injury problem in Section 1.3 with the C_i resulting from weighting the three objectives: sensitivity, specificity and cost. We evaluate the behavior of the methods associations retrieved as optimal for the differential

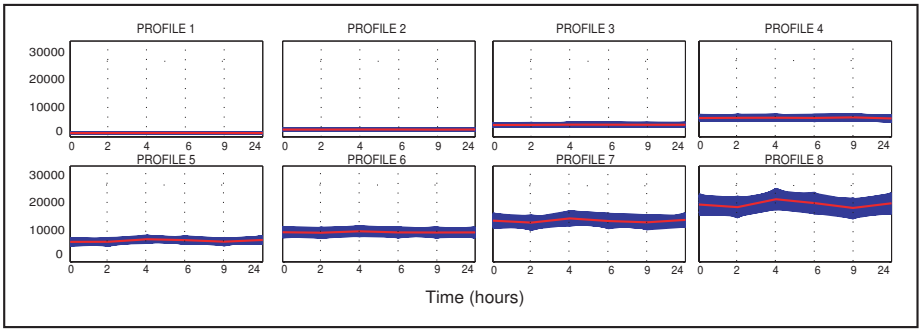


Figure 3.32: Random profiles generated for the control groups. The centroids are highlighted in red.

profiles in the inflammation problem over the differential profiles of the artificial set of data expression data.

We compare the values obtained for the training (inflammation and host response to injury problem) in Fig. 3.35 nd the test (artificially created profiles) in Fig. 3.36.

We have also used the graphical representation as with the inflammation to get a wider picture of the behavior of the methods associations based on the union operator with each of the differential profiles in Fig. 3.36. Instead of creating a lookup table, we have created a table implementing the same functional ∫ operator as to create the coefficient C_i, the weighted sum. The weights assigned have been $\omega_1 = 0.30$, $\omega_2 = 0.55$ and $\omega_3 = 0.30$. The sensitivity has been favored in the weighted sum since it is the objective that we are more interested in to get a wider picture of the behavior of the genes in our current problem.

When compare both images, we see the behavior of methods over the differential profiles is very similar. For instance, differential profiles such as #5 or #15 are easily retrieved my methods methods that take into account the time condition in both the inflammation data set and the artificial data set. Particularly, Student's T-test considering time retrieves such differential profiles, while the rest of the method associations do not perform very well. In both data sets, profiles #18 to #29 are easily retrieved by all methods. At the view of these results, we can conclude that the results obtained by application of the methodology proposed are capable to retrieve the differential profiles as described by the association rules not only in the inflammation data set but in any any mi-

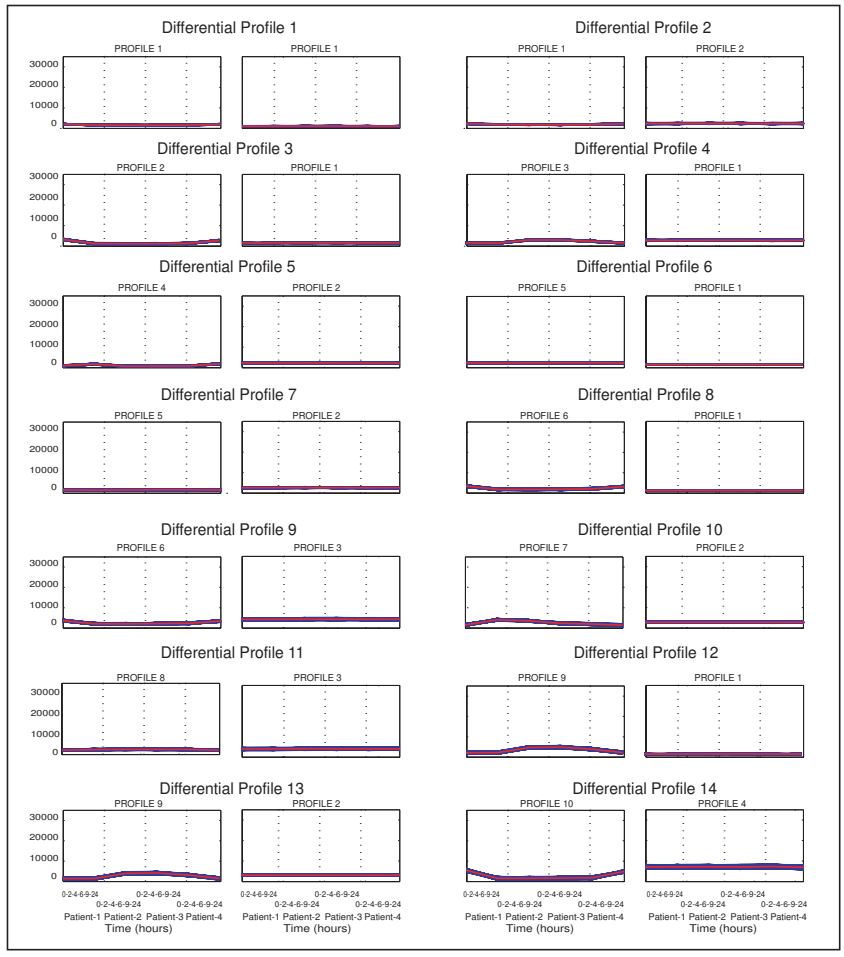


Figure 3.33: Differential profiles (P_{T_m}, P_{C_n}, G) 1 to 14 from the inflammation data problem.

croarray data and these rules are, therefore, valid for further application. The same process has been applied over the information obtained from application of the $\cap$ operator. The corresponding images can be found in Appendix A in Fig. 11 and Fig. 12.

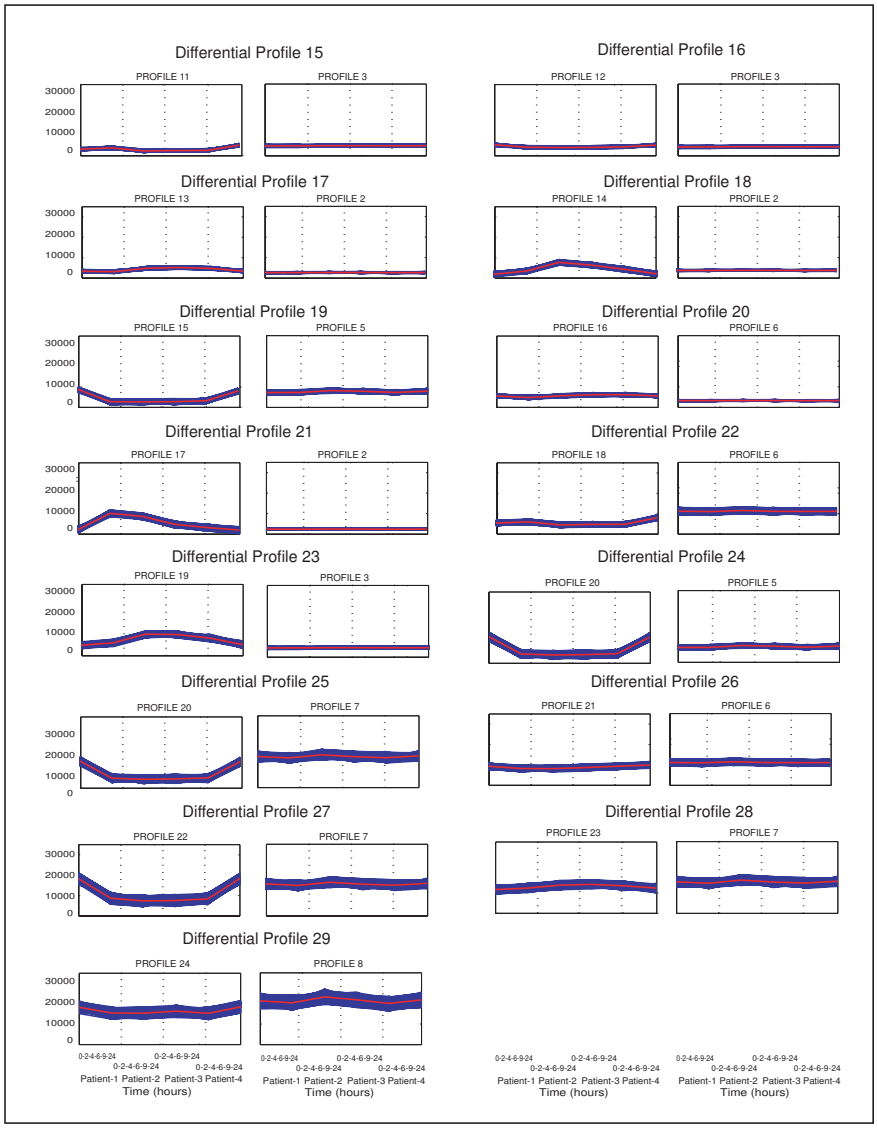


Figure 3.34: Differential profiles (P_{T_m}, P_{C_n}, G) 15 to 29 from the inflammation data problem.

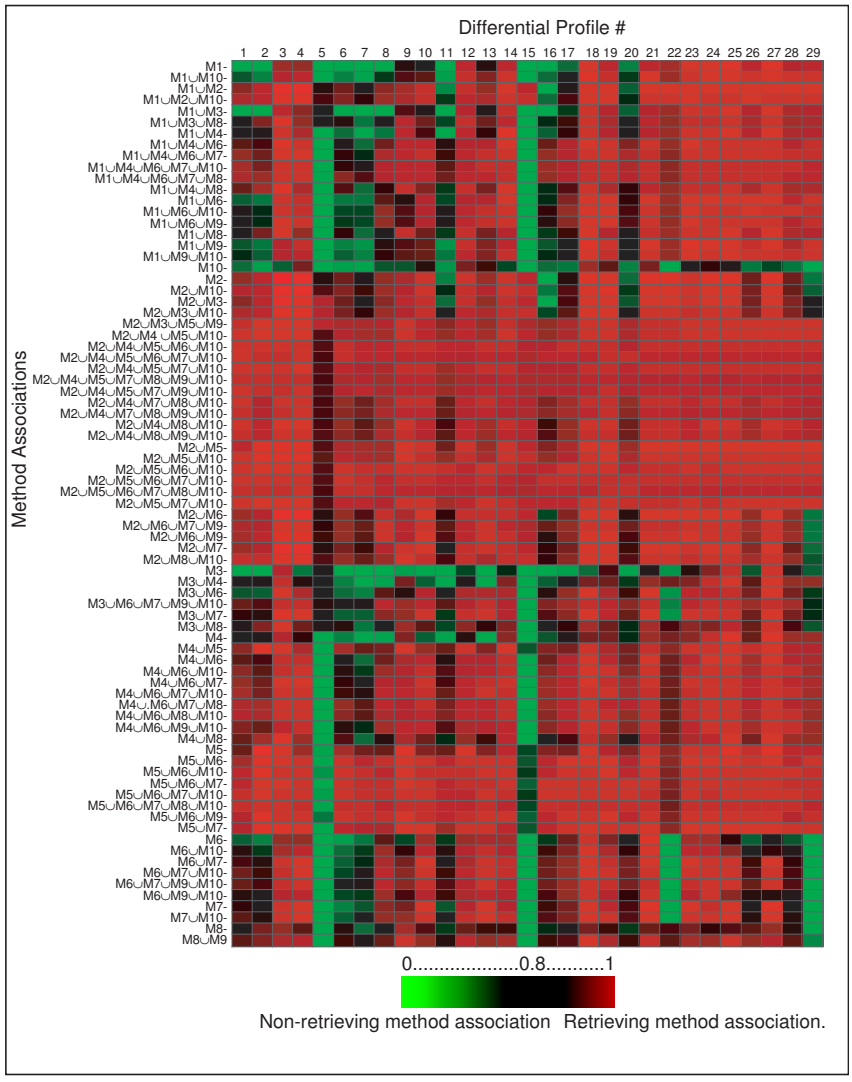


Figure 3.35: Behavior of the optimal associations of methods using the ∪ operator in retrieving the differential profiles from the inflammation and host response to injury problem.

4 Concluding Remarks

In this chapter we proposed a methodology which tries to alleviate the problem caused by classical microarray analysis methods not being capable to extract

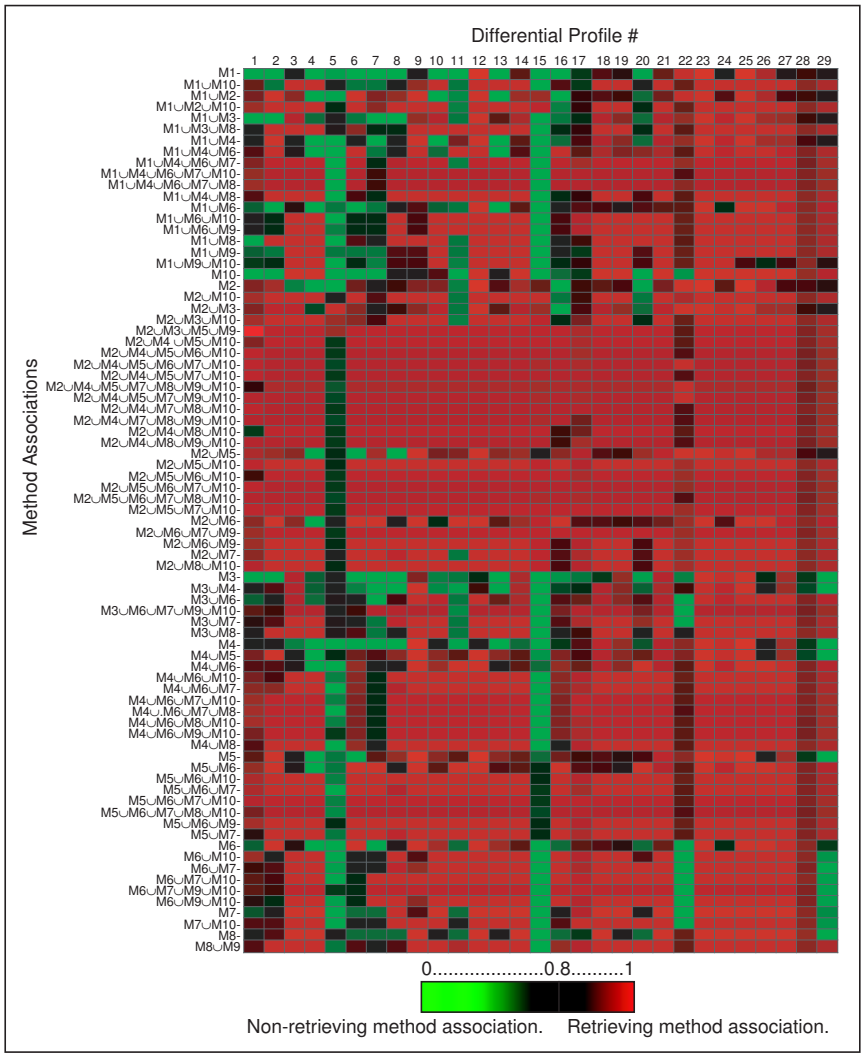


Figure 3.36: Behavior of the optimal associations of methods using the ∪ operator in retrieving the differential profiles from the artificial data set.

on their own all the information present in microarray data sets (see Chapter 3). This methodology can be described as a conceptual clustering approach (Cheeseman and Oldford, 1994; Cook et al., 2001b; Zwir et al., 2005b; Zwir et al., 2005c), devoted to identify optimal associations among microarray analysis methods in an effort to identify gene expression profiles from microar-

ray data sets. In this approach, we created set of association rules which relate microarray analysis method associations to differential profiles (i.e., sets of genes with coordinate changes in RNA abundance in all experimental conditions, including treatment, control and patients) they are able to recover.

As a first step, we identified from the inflammation problem (*see* Capter 2 Section 1) a set of differential profiles (P_{T_m}, P_{C_n}, G), whose biological relevance has been partially explained by means of mining several biological databases and will be extensively described in the following chapter (see Chapter 4. The mining has proved that the differential profiles contain genes (or probe sets) cohesive in their biological functions.

Once the differential profiles were extracted, we identified method associations capable to retrieve these profiles using the $\cup$ and $\cap$ set operators. We organized them in a lattice, which constitutes our space of potential hypotheses for relationships between profiles and methods. We evaluated these relationships using multiobjective optimization techniques based on three criteria: specificity, sensitivity and cost. The results showed that methods associations are required to identify differential profiles that exhibit changes among time, treatment and control and patients. Indeed, their performance highly overcomes the individual microarray analysis methods. We also saw how some the probe sets not retrieved by the classical methods are indeed relevant for the problem under study.

We encoded the former relationships between differential profiles and association of methods into decision making association rules. The application of these rules generates recommendations for optimal method associations capable of identifying a desired set of differential profiles. These rules present some noteworthy characteristics, such as the capacity to combine them under certain conditions without lost of optimality in the non-dominance relation. Indeed, they can be organized at different levels of granularity and thus, provide a framework of rules with distinct levels of accuracy and interpretability. We successfully tested the ability of the learned association rules to recover other microarray expression data by using an artificial dataset.

All of these reveal that the proposed methodology can be generalized for other gene expression experiments. Indeed, it can be used for combining the salient characteristics of different machine learning methods, as organized multiclassifiers, to identify the behavior of a system in different fields and application (Guigo and Consortium., 2007). Finally, the obtained knowledge can be fused with additional information to validate and increase the confidence of the obtained rule base (see Chapter 4) and incorporate different new types of profiles (see Chapter 5).

Chapter 4

Biological Significance of the Differential Profiles Obtained from the Inflammation and Host Response to Injury Problem

Genes sharing the same expression pattern are likely to be involved in the same regulatory process. Though in theory it is a big step from simple correlation analysis to gene interaction networks, several papers indicate that the clustering of gene expression data does result in groups of genes that have related functions (D'haeseleer et al., 2000; Eisen et al., 1998).

We address here the question of detecting the level of relation between expression patterns and biological function by mining several biological databases in order to asses how cohesive the differential profiles obtained from the inflammation and host response to injury problem are. Four different biological databases have been used (1) *OMIM* (Online Mendelian Inheritance in Man), (Hamosh et al., 2005), a database of human genes and genetic disorders, (2) *KEGG* (Kyoto Encyclopedia of Genes and Genomes), (Kanehisa et al., 2002), a resource for linking genomes to biological systems and environments, (3) the *Gene Ontology* project (Consortium, 2000), a controlled vocabulary to de-

scribe gene and gene product attributes in any organism and (4) *GeneBank* (Benson et al., 2007),the NIH genetic sequence database, an annotated collection of all publicly available DNA sequences.

The use of techniques of automatic management of biological information in the study of the coherence of gene groups has only recently been addressed (Oliveros et al., 2000). For the automatic mining of the mentioned databases, we will use already existing algorithms as well as new algorithms developed by us as part of this PhD. work. Throughout this chapter we will describe how the mining of the biological databases has been approached: mining of the *OMIM* and *KEGG* databases, which contain plain data, in relation to the genes in the differential profiles obtained from the inflammation and host response to injury problem will be approached using the *Apriori* algorithm (Agrawal and Shafer, 1996b) a classic algorithm for learning association rules. The *Gene Ontology* project, a structural database, will be mined using an algorithm termed *EMO-CC* (Evolutionary Multi-Objective Conceptual Clustering), proposed by us in Romero-Zaliz *et al.,* (2007), which retrieves meaningful substructures from network databases using multi-objective and multi-modal optimization techniques. The information obtained from the *GeneBank* related to the genes in the differential profiles from the inflammation and host response to injury problem will be mined applying a new program, satDNA analyzer, developed by us part of this PhD. work (Navajas-Pérez et al., 2007). The program automatizes the analysis of satellite-DNA sequences from aligned DNA sequence data.

We show in Fig. 4.1 a description of the proposed methodology including the biological assessment of the differential profiles obtained from the inflammation and host response to injury problem.

1 Mining OMIM and KEGG

We address the question of the detection of the level of relation between expression patterns and biological function by mining into *OMIM* (Online Mendelian Inheritance in Man), (Hamosh et al., 2005), a database of human genes and genetic disorders and *KEGG* (Kyoto Encyclopedia of Genes and Genomes), (Kanehisa et al., 2002), a resource for linking genomes to biological systems and environments. *OMIM* has been chosen for being a database only containing information specific to the human genome, and therefore it will provide human-specific information in relation to the inflammation and host response to injury problem treated throughout this work. The election of *KEGG* has been based on the fact that is one of the more widely used and complete open

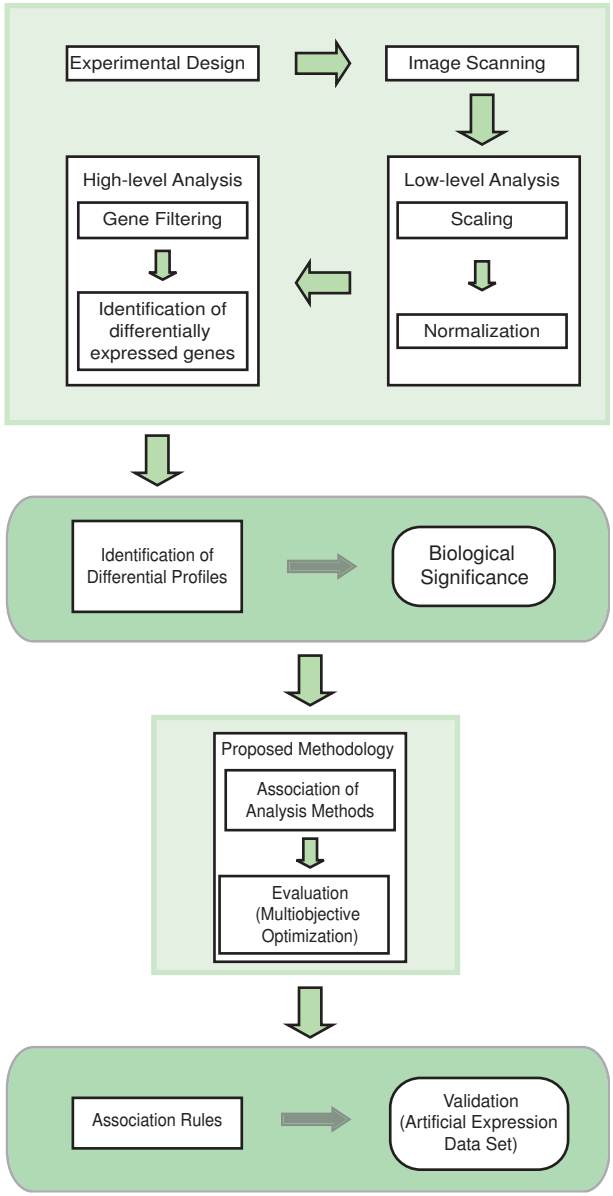


Figure 4.1: Schema of the methodology proposed including the assessment of biological significance of the differential profiles

source metabolic pathway search tools. *KEGG* is a collection of manually drawn pathway maps representing knowledge on the molecular interaction and reaction networks for metabolism, genetic information processing, environmental information processing and cellular processes.

Both databases have been accessed applying software tools developed at the HUSAR group at the Deutsches Krebsforschungszentrum (DKFZ), Division of Molecular Biophysics (del Val et al., 2006). Unigene identification codes (Schuler et al., 1996a) have been obtained for each of the probe sets in the differential profiles from the inflammation and host response to injury problem by mapping the annotation files associated to the GeneChips® HG-U133A v2.0, Affymetrix Inc. Unigene is an *NCBI* database of the transcriptome, where each entry is a set of transcripts that appear to stem from the same transcription locus (i.e. gene or expressed pseudogene). Information on protein similarities, gene expression, cDNA clones, and genomic location is included with each entry.

Plain data (flat fields without any structure underlying them) has been retrieved from both databases in relation to each of the differential profiles. The *Apriori* algorithm (Agrawal and Shafer, 1996b) has been applied to find frequent subsets of database descriptions from the information retrieved. The *Apriori* algorithm is a classic algorithm for learning association rules which attempts to find subsets common to at least a minimum number C (the cutoff, or confidence threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation, and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

As a result of applying the Apriori algorithm, common subsets of database annotations have been found for each of the differential profiles. The subsets retrieved are combination of *OMIM* and *KEGG* codes which are shared for the probe sets in certain differential profiles (i.e., the probe sets in a differential profiles share common *OMIM* and *KEGG* information). For instance, probe sets grouped as differential profile #19 appear in the frequent item sets with *KEGG* pathways $KEGG03010$, $KEGG72766$ and $KEGG74160$, meaning that probe sets in differential profile #19 are related to such pathways. These pathways are associated to *apoptosis*, *cell cycle* and the *ribosome*, terms very related to the inflammation and host response to injury problem under study. Probe sets in differential profile #19 also appear in the frequent item sets with codes from the *OMIM* database related to *ribosomal proteins* $OM180460 and OM603636$, *apoptosis* $OM604170$ and *cytokine signaling* (604170), terms also related to the inflammatiion and host response to injury problem. In Table 1 we show some frequent subsets obtained by the *Apriori* algortihm.

DP	OMIM and KEGG codes
#1	OM17687 OM17687 KEGG04510 KEGG04910 KEGG04810
#3	OM10291 OM10291 KEGG00193 KEGG00190 KEGG15869
#4	OM13563 OM13563 KEGG04360 KEGG04514 KEGG04512
#9	OM16436 OM16436 KEGG00193 KEGG00190 KEGG15869
#10	OM60831 OM20780 KEGG00330 KEGG00220 KEGG71291
#11	OM60099 OM60099 KEGG04020 KEGG04540 KEGG10958
#12	OM13122 OM60304 KEGG00230 KEGG00240 KEGG15869
#17	OM10747 OM10747 KEGG04060 KEGG04630 KEGG04650
#19	OM60370 OM60370 KEGG03010 KEGG74160 KEGG72766

Table 4.1: Frequent itemsets retrieved by the Apriori algorithm mining the OMIM and KEGG databases in the relation to the differential profiles from inflammation and host response to injury problem.

2 Mining the Gene Ontology Project

The GO project (Consortium, 2000) stores one of the most powerful characterizations of genes. It uses three structured vocabularies (i.e., ontologies) to describe gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner (Consortium, 2000). The GO terms are organized as hierarchical networks, where each level corresponds to a different specificity definition of such terms (i.e., higher level terms are more general than lower level terms, Fig. 4.2). From the computational point of view, these networks are organized as structures termed DAGs, which are one way routed graphs that can be represented as trees.

Current tools and techniques devoted to examine the content of large databases are often hampered by their inability to support searches based on criteria that are meaningful to their users. These shortcomings are particularly evident in data banks storing representations of structural data such as biological networks. Conceptual clustering techniques have demonstrated to be appropriate for uncovering relationships between features that characterize objects in structural data. However, typical conceptual clustering approaches normally recover the most obvious relations, but fail to discover the less frequent but more informative underlying data associations. This is mostly caused by biasing the searches in the feature space of the investigated problem, constraining them to predefined areas of potential solutions and specific levels of details, and restraining the evaluation criteria for the data association to a single quality measure. The combination of evolutionary algorithms with multi-objective and multi-modal optimization techniques constitutes a suitable tool for removing

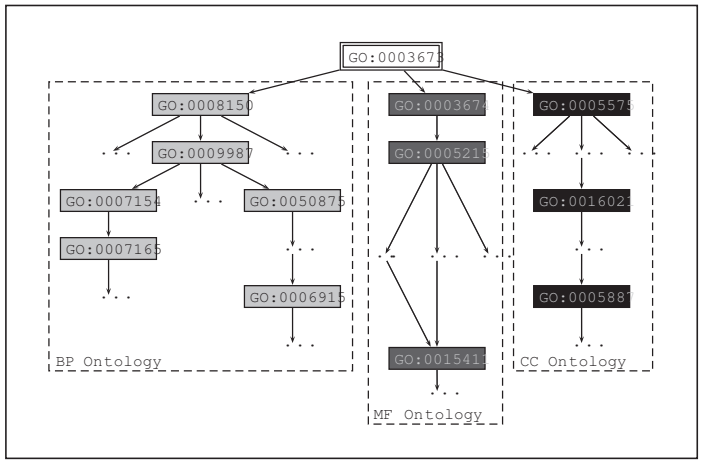


Figure 4.2: The GO project ontology. The GO database is composed of three sub-ontologies, which are shown at different colors starting from the root node GO:0003673: Biological Process (BP), Molecular Function (MF) and Cellular Component (CC).

these biases.

We make use of a novel conceptual clustering methodology termed *Evolutionary Multi-Objective Conceptual Clustering (EMO-CC)*, relying on the NSGA-II multi-objective (MO) genetic algorithm, that focuses primarily on the discovery of objects identified by their most representative features lying in the set of all optimal solutions of a multi-objective optimization problem. We apply this methodology to identify conceptual models in structural databases generated from gene ontologies. These models can explain and predict phenotypes in the inflammation and host response to injury problem, similar to models provided by gene expression or other genetic markers. We compare the results obtained in our methodology with other conceptual clustering. The analysis of these results reveals that our approach uncovers cohesive clusters, even those comprising a small number of observations explained by several features, which allows describing objects and their interactions from different perspectives and at different levels of detail. This approach provides novel annotations that are often concealed by methods that emphasize most frequent descriptions.

2.1 Methodology Description

EMO-CC (Evolutionary Multi-Objective Conceptual Clustering) retrieves meaningful substructures from network databases using multi-objective and multi-modal optimization techniques (Romero-Zaliz et al., 2007). We apply *EMO-CC* to the *Gene Ontology* database (i.e., the GO Project, (Ashburner et al., 2000)), which consists of three structural networks of terms describing gene product features. *EMO-CC* recovers optimal substructures containing genes sharing a common set of terms, which are defined at different levels of specificity and correspond to different networks, producing novel annotations.

The increased availability of repositories containing representations of complex objects in spatial databases, such as satellite maps, or temporal databases, including microarray time series, regulatory networks or metabolic pathways, permits access to vast amounts of data where these objects may be observed (Siripurapu et al., 2005; Nikitin et al., 2003; Consortium, 2000). However, the underlying object representations used in these databases are typically based on computational convenience of database implementers and their tendency to increase the amount of stored data (Schuler et al., 1996b). Current tools and techniques devoted to examine the contents of these large databases are hampered by their inability to support searches based on criteria that are meaningful to the users of those repositories. In particular, and in spite of the recent renewed interest in knowledge discovery techniques (or data mining), there is a dearth of data analysis methods intended to facilitate the understanding of the represented objects and related systems by their most representative features and those relationships derived from these features (i.e., structural data (Cook et al., 2001a)). Plain databases cannot deal with this structural information. For example, images often stored in spatial databases are composed of small pieces of geometrical objects (e.g., triangles or squares) that encode complex relationships between them, including nested or composite relative locations (e.g., square on triangle, Fig. 4.3 (a1-a2)). This type of relationships normally exceeds the simple presence/absence of the underlying elements (e.g., triangle and square). Indeed, plain data, as data at the *OMIM* and *KEGG* databases, are difficult to generalize into more abstract concepts (e.g., object on triangle) resulting from frequent patterns found in the database (Fig. 4.3 (d2)).

Structural data, in contrast to plain data, can be viewed as a graph containing nodes representing objects. Sub-graph partitions of the dataset are termed *substructures* (Cook et al., 2001a) (Fig. 4.3 (b-d)). Each object in a substructure is described by its most representative features, which are encoded as nodes linked to other nodes by edges corresponding to their relationships. Conceptual clustering techniques have been successfully applied to structural data to uncover concepts that explain underlying objects by searching through a predefined

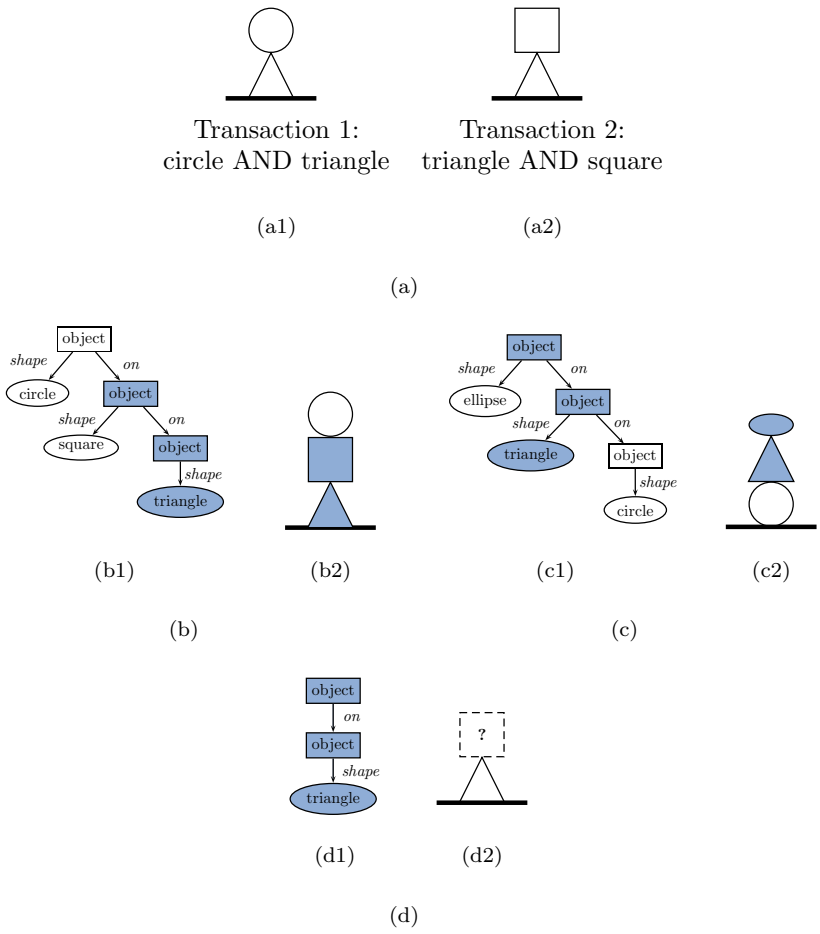


Figure 4.3: Differentiating plain and structural databases: example of geometrical observations. (a) Plain codification of two observations as typical transactions encoding a presence/absence relationship between data. (b-c) Structural codification also encoding positional relationships. (b1-c1) Tree-graphs corresponding to computational representations in a structural database. (b2-c2) Geometrical interpretation of the represented observations. The color-coded parts of the trees show repeated instances that generate substructures. (d) A generalized substructure learned from (b-c) that cannot be encoded by (a).

space of potential hypotheses (i.e., substructures that represent associations of features) for those that best fits the training examples (Mitchell, 1997).

The formulation of conceptual clustering as a search problem, in a graph-based structure, would result, however, in the generation of many substructures with small extent, as it is easier to explain or model smaller data subsets than those that constitute a significant portion of the dataset. For this reason, any successful methodology should also consider additional criteria to extract better defined concepts based on the complexity of the substructure being explained, the number of retrieved substructures, and their diversity (Cook et al., 2001a; Ruspini and Zwir, 2001; Zwir et al., 2002; Romero-Zaliz et al., 2004). These are conflicting criteria that can be approached as an optimization problem, close in spirit to Minimum Description Length (MDL) methods (Rissanen, 1989), which are based on the aggregation of the various objectives into a global measure of cluster quality. The basic challenge with this approach is the potential bias caused by weighting the objectives (Ruspini and Zwir, 2001; Ruspini and Zwir, 1999), which always derives in the convergence to solutions corresponding to single or limited regions of the search space. This problem is noteworthy because typical data mining approaches, particularly in computational biology, tend to emphasize consensus or most frequent patterns (Zwir et al., 2005a). These consensus patterns often conceal rather than reveal novel and useful knowledge about the problem, retrieving only already known or irrelevant information that discourages the use of computational methods (McCue et al., 2001; Martinez-Antonio and Collado-Vides, 2003). Consequently, there is a need of new methods that can provide even less frequent but more descriptive substructures that reflect problem descriptions from different angles (Zwir et al., 2005d).

The described technique is a conceptual clustering methodology termed *EMO-CC* for *Evolutionary Multi-Objective Conceptual Clustering* that uses multi-objective and multi-modal optimization techniques to retrieve meaningful substructures from structural databases. The EMO-CC methodology uses an efficient search process based on evolutionary algorithms (EA) (Back et al., 1997; Deb, 2001; Coello-Coello et al., 2002) relying on the NSGA-II algorithm (Deb et al., 2000), which inspects large data spaces that otherwise would be intractable. Indeed, it explores hierarchically organized databases, which can contain data defined at different levels of specificity. EMO-CC identifies optimal clusters corresponding to different substructures lying in the Pareto optimal frontier (Deb, 2001; Ruspini and Zwir, 2001). This frontier is composed of a collection of multi-objective optima in the sense that their solutions are not worse than any other substructure for the objectives being considered (i.e., non-dominated) (Deb, 2001). This approach is less biased than aggregating various objectives into a weighted function. The clusters obtained by EMO-CC are

composed of solutions belonging to different neighborhoods, where each cluster represents a local optimum in a multi-modal problem. The methodology optimizes the number of substructures being retrieved based on a flexible compression of the database and provides annotations for the uncovered substructures. Finally, EMO-CC applies an unsupervised classification approach to predict new members of previously discovered substructures.

We apply EMO-CC to the discovery of meaningful substructures containing genes sharing common sets of features (i.e., GO terms) in the Gene Ontology project (GO) database (Consortium, 2000), which is composed of biological processes, cellular components and molecular functions defined at different levels of specificity. These substructures can explain/predict gene expression profiles. We consider gene profiles that reflect differences in gene expression over time, treatment and patient, corresponding to the inflammation and host response to injury problem (Calvano et al., 2005).

2.1.1 Methodology Preliminaries

In this section, we provide the methodological and problem background used in this section. First, we briefly supply a general framework for conceptual clustering algorithms, and introduce two methods used to mine structural databases. These methods include the SUBDUE conceptual clustering method (Jonyer et al., 2001) and the APRIORI unsupervised method (Agrawal and Shafer, 1996a). Second, we describe the GO project and its structural database. These two methods are selected to be compared with our approach. We also provide a brief survey of evolutionary and multi-objective optimization. Finally, we characterize the multi-objective optimization problem and define a set of metrics used to evaluate the quality of the results obtained by EMO-CC in comparison with the other methods.

EMO-CC is defined as a conceptual clustering methodology. Cluster analysis, or simply clustering, is a data mining technique often used to identify various groupings or taxonomies in databases (Duda et al., 2000). Most existing methods for clustering are designed for plain feature-value data. However, sometimes we need to represent structural data that not only contain descriptions of individual observations, but also relationships between these observations. Therefore, mining structural databases entails addressing both the uncertainty of which observations should be placed together, as well as which distinct relationships among features best characterize different sets of observations. This is more problematic since, *a priori*, we do not know which features are meaningful for a given relationship. Typical clustering techniques (Der and Everitt, 1996) are not designed to deal with this, even when combined with global feature ex-

traction methods such as principal component analysis or stepwise descendant methods (Liu and Motoda, 1988; Yeung and Ruzzo, 2001).

In contrast, conceptual clustering techniques have been successfully applied to structural databases to uncover concepts that are embedded in subsets of structural data or substructures (Cook et al., 2001a). Consequently, conceptual learning can be formulated as the problem of searching through a predefined space of potential hypotheses (i.e., substructures or associations of features and observations) for those that best fit the training examples.

While most machine learning techniques applied directly or indirectly to structural databases exhibit methodological differences, they share a five-steps framework, even though they use distinct metrics, heuristics or probability interpretations (Cheeseman and Oldfors, 1994; Cook et al., 2001a):

- **Database representation.** Structural data can be viewed as a graph containing nodes representing features, linked to other nodes by edges corresponding to their relations. A substructure consists of a sub-graph of structural data, which represents an object of a concept embedded in the data (Cook et al., 2001a). These data can be efficiently organized by taking advantage of a naturally occurring structure over the feature space, which consists of a general to specific ordering of possible substructures (i.e., a direct acyclic graph (DAG) (Aho et al., 1983)).
- **Structure Learning.** This process consists of searching through the DAG space for potential substructures, and returning either the best one found or an optimal sample of them. If the number of substructures is super-exponential in the number of nodes, different heuristic methods can be applied for this learning process (e.g., greedy (Chickering, 2003); hill climbing (Chickering, 2003); genetic algorithms (Larranaga et al., 1996)).
- **Cluster evaluation.** The substructure quality is measured by optimizing several criteria, including complexity, where harboring more features always increases the inferential power; support, where a large coverage of the dataset produces good generality; and diversity, where minimal overlapping between clusters generates more distinct clusters and descriptions from different angles. The basic challenge with this approach consists of fixing the potential bias and inflexibility caused by combining these criteria in a weighted sum formula (Ruspini and Zwir, 2001; Rissanen, 1989).
- **Database compression.** The database compression provides simpler representations of the objects in a database. This procedure is often done by selecting the best substructures and replacing their instances by single vertices. However, it may be the case that these summarized substructures

need to be decompressed or re-compressed when they are combined with different or independent data sources (Jonyer et al., 2001).

- **Inference.** New observations can be predicted from previously learned substructures by using classifiers that optimize their matching based on distance (Bezdek, 1998a) or probabilistic metrics (Mitchell, 1997). When designed for labeled data, the approach is referred to as supervised learning (as opposed to unsupervised learning) (Mitchell, 1997).

Here we expalin two different methods, one originally designed to perform conceptual clustering and another, adapted to work with structural databases.

SUBDUE. This method (Jonyer et al., 2001) is a typical example of a conceptual clustering approach that finds repeated substructures in databases represented as graphs. SUBDUE starts by looking for the substructure that best compresses the graph using the MDL principle (Rissanen, 1989), which states that the best description of a dataset is the one that minimizes the description length of the entire dataset. After finding the first substructure, SUBDUE compresses the graph and can iterate to repeat the same process. SUBDUE uses a computationally-constrained beam search strategy to find substructures. The algorithm starts with a single vertex as the initial substructure and at each iteration expands it by adding instances exploring every possible edge to generate new substructures that will recursively be considered for expansion.

APRIORI. This method (Agrawal and Shafer, 1996a) uses a classic algorithm for learning association rules. It is designed to operate on databases containing transactions (e.g., collections of items bought by customers, or items of a website access). The algorithm attempts to find subsets of items (e.g., sets of retail transactions of each listing individual items purchased) shared by at least a minimum number of observations. This approach is similar to the computation of biclusters, as it allows simultaneous clustering of features and transactions (Grothaus et al., 2006). APRIORI uses a bottom-up approach, where frequent subsets are extended one item at a time, and candidate sets of items are evaluated by their supporting observations. The algorithm terminates when no further successful extensions are found. APRIORI uses breadth-first search and a hash tree structure to count candidate sets of items efficiently. It generates candidate sets of length k from sets of item of length $k-1$. Then, it prunes the candidates which have an infrequent sub-pattern (Agrawal and Shafer, 1996a). This algorithm was originally designed to work with plain data, and here adapted to manage structural databases.

2.2 Evolutionary and multi-objective optimization

Evolutionary algorithms are often used to solve knowledge discovery or data mining problems. Several evolutionary algorithms (EAs) have been successfully applied in classical clustering problems including hard and fuzzy c-means functional optimization (Hall et al., 1999) and the estimation of the optimal number of clusters (Pan and Cheng, 2007). Moreover, genetic algorithms in combination with multi-objective optimization techniques have been used for selecting features in an unsupervised fashion (Handl and Knowles, 2007) and for developing multi-classifiers (Morita et al., 2003). Linguistic and association rules also incorporated evolutionary techniques for their optimization and searching processes (Delima and Yen, 2005; Alatas et al., 2008). Indeed, biclustering techniques, often used in bioinformatics (Prelic et al., 2006), use an appropriate combination of EAs and multi-objective optimization (Mitra and Banka, 2006).

We incorporate some of the former features successfully applied to knowledge discovery to develop a novel evolutionary method focused on conceptual clustering data.

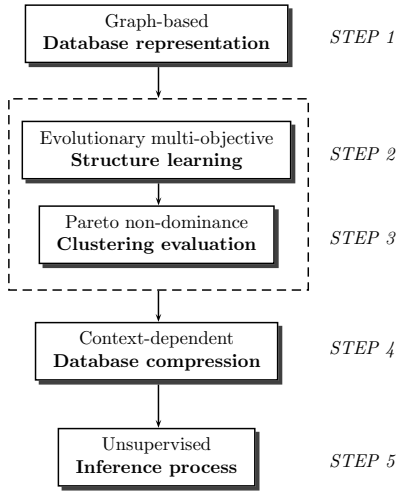


Figure 4.4: The EMO-CC methodology. The different steps of EMO-CC are developed based on the typical phases of a conceptual clustering method. The dashed box represents the searching and evaluating iterative process carried out by the multi-objective evolutionary algorithm.

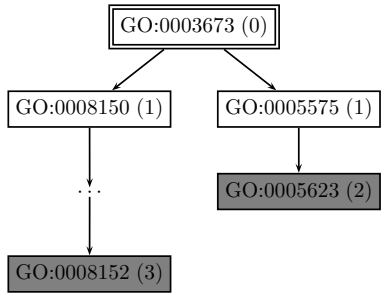
2.3 Graph-based database representation

The database representation used for the GO domain can be viewed as a database containing different features, where each feature has nested values denoting descriptions at different levels of specificity. Therefore, the identification of which relationships among features best characterize different sets of observations have to consider, not only the process of grouping distinct type of features (e.g., biological process GO:0007165 and GO:0050785, representing a signal transduction process and an advanced glycation end-product receptor activity, respectively, and cellular component GO:0016021, representing an integral to membrane situation), but also defining at which level of specificity they have to be represented. This is even more problematic since several values of the same type of feature may be useful for describing a set of observations, and thus, represented in a substructure (e.g., biological process GO:0007165 (level 4) and GO:0050785 (level 3)). Consequently, to address the problem of the multi-level definition of a feature we re-define an instance as the particular subset of values that constitutes a prefix tree[1] of a database observation. Then, an instance of a substructure occurs in an observation of the database if a sub-graph of the prefix tree that represents that instance matches with the observation tree. The substructure tree contains tagged nodes with the type of feature (e.g., biological process), its corresponding value (e.g., GO:0007165), and the edges representing relationships between features (e.g., is_a).

We use the GO database and compatibilize the terms with descriptions provided by Affymetrix for the GeneChips® HG-U133A v2.0, where each observation of the database has the following features:

- *Name:* Affymetrix identifier for each gene in HG-U133A v2.0 set of arrays.
- *Biological process:* List of biological processes where a gene product is involved, which are indexed by a list of GO codes (e.g., GO:0007067 (mitosis), GO:0008152 (metabolic process)). The processes are broad biological goals that are accomplished by ordered assemblies of molecular functions.
- *Molecular function:* List of biological functions of gene products, which are indexed by a list of GO codes (e.g., GO:0030246 (carbohydrate binding), GO:0016887 (ATPase activity)). These functions are tasks performed by individual gene products.
- *Cellular component:* List of cellular components indicating location of gene products, which are indexed by a list of GO codes (e.g., GO:0005634

[1] Tree t' is a prefix tree of t if t can be obtained from t' by appending zero or more sub-trees to some of the nodes in t'. Notice that any tree t is a prefix of itself.



(a)

207929_at
202442_at
213475_s_at
214467_at
219889_at
203508_at
201087_at
209864_at
200625_s_at

(b)

Figure 4.5: An example of a chromosome representing a substructure ($specificity = 0.6769$, $support = 0.0051$). (a) A tree representation of a substructure, where the gray boxes are the most specific GO terms, and the levels of the terms in the GO hierarchy are shown in parenthesis. (b) The list of genes that corresponds to the substructure.

(nucleus), GO:0019012 (virion)). These components are sub-cellular structures, locations, and macromolecular complexes.

2.4 Multi-objective GP structure learning

The chromosome representation used in the GO domain is a tree-like structure (Fig. 4.5). Each node of this tree corresponds to a GO term, and each edge corresponds to a `is_a` or `part_of` relationship.

The complexity of the substructures in the GO domain is not linearly dependent on its size. This happens because the GO ontology is composed of terms that can be located at different levels in the hierarchy. For example, a substructure (substructure #1) is less specific than another substructure (substructure #2), if the leaf nodes from the former belong to a lower level (level 4) than the latter (level 5) (Fig. 4.6). However, by calculating the complexity as the number of edges plus nodes of each substructure, the first substructure reaches a higher evaluation value (i.e., $complexity = 8$) than the second (i.e., $complexity = 7$). Thus, we redefine the complexity as *specificity*, extending the original objective by including, not only the size of the substructure measured by the number of nodes and edges, but also the accuracy of the substructure in modelling the covered instances:

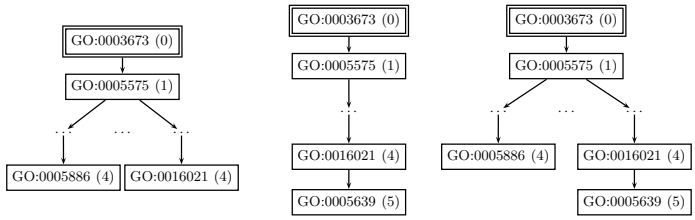


Figure 4.6: Relationship between substructures and observations. (a) *substructure #1* and (b) *substructure #2* both represent an observation (c). In this example, *substructure #2* is more specific than *substructure #1* and, therefore, more complex, since the leaf nodes from the former belong to level 5, while those of the latter belong to level 4. The double frame box corresponds to the root of the GO, the boxes indicate cellular component terms, and the number in parenthesis correspond to the level of the nodes in the GO hierarchy.

$$Specificity(s) = \frac{\sum_{i=1}^{k} \left(1 - \sum_{u=1}^{l} \frac{dist(node_{ui}, s)}{level(node_{ui})}\right)}{k} \tag{4.1}$$

where k is the number of instances occurring in substructure s, l is the number of leaf-nodes in the *i-th* instance occurring in substructure s, $node_{ui}$ is a leaf-node of the *i-th* instance occurring in substructure s. The distance *dist* between a node and a substructure is calculated as the number of edges between the given node and a its closest ancestor in the GO hierarchy appearing in the given substructure. The *level* of a node is calculated as the length of the shortest path

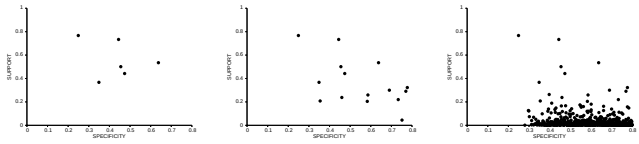


Figure 4.7: The Pareto fronts obtained by different methods. Each dot represents a solution with the support given by its value on the y axis, and the specificity given by its value on the x axis. Non-dominated solutions reported by: (a) APRIORI, (b) SUBDUE, and (c) EMO-CC.

to the root node. For a perfect match (i.e., $specificity = 1$), all nodes of the instance must appear in the substructure and their distances to the substructure must be zero.

3 Experiments and Analysis of Results

The structural database used for the GO domain is composed of 2155 significantly expressed genes, extracted from the set of the total genes available in a GeneChip (i.e., approximately 22000), and their GO associated terms. The population of the EA is initialized by 50% of randomly chosen subtrees from the database, and by another 50% of random trees. This randomization procedure is needed to avoid the potential bias introduced in the search process using only a subset of GO terms instead of the complete GO database.

We execute EMO-CC ten times with different seeds and a set of parameters that maximizes the computational performance (Table 3). We analyze the sensitivity of the parameters, increasing the population (e.g., from 200 to 800) and changing the operator probabilities (e.g., crossover from 0.6 to 0.9 and mutation from 0.1 to 0.3). The similar results obtained by this analysis suggests that the NSGA-II has a robust behavior. Then, we used the average of the ten runs to report the results evaluated by the metrics $\mathcal{M}_2^*$, $\mathcal{M}_3^*$, $\mathcal{C}$ and $\mathcal{ND}$.

Parameter	Value
Population Size	200
Number of Evaluations	20000
Crossover probability	0.6
Mutation probability	0.2

Table 4.2: Parameters for the GO domain

In the following subsections we show the experimental results obtained by EMO-CC in the inflammatory response problem. In the first subsection we compare EMO-CC with two other methods: the conceptual clustering method SUBDUE (Jonyer et al., 2001), and the APRIORI unsupervised method (Agrawal et al., 1993), which is adapted for using structural data. In the second subsection, we perform a context-dependent database compression of the learned substructures that can explain gene expression.

3.1 Pareto non-dominance clustering evaluation

Since both APRIORI and SUBDUE methods are not MO algorithms, we remove from the final set of solutions of both methods those solutions that are dominated, to provide a comparable set of substructures. For APRIORI, we also transform the original structural database into a plain repository by adding all parent terms for each GO term used in the biological application. The results for a single run are reported. We show the union of the results obtained by SUBDUE from three runs, each one using a different optimization criteria, including: support (i.e., the number of instances occurring in a substructure), complexity (i.e., the size of the substructure calculated as the number of bits needed to encode the adjacency matrix corresponding to the graph (Jonyer et al., 2001)), and a weighted sum metric that combines the latter two (i.e., MDL (Rissanen, 1989)), which is the default option of SUBDUE. The results obtained by APRIORI and SUBDUE are compared with each of the Pareto sets found by EMO-CC, when using the MO evaluation metrics.

The substructures recovered by EMO-CC obtain a better coverage of the Pareto front extent than SUBDUE and APRIORI.

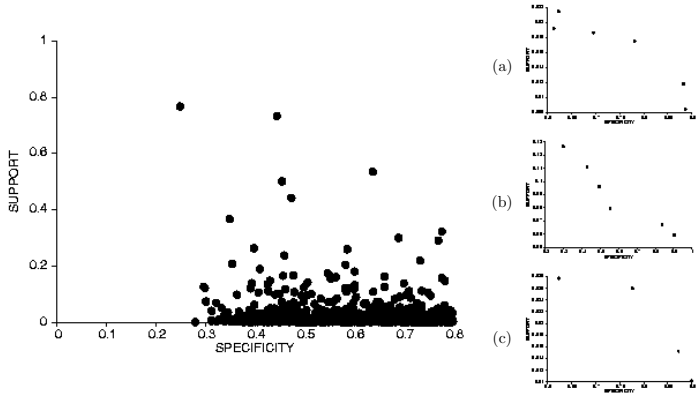


Figure 4.8: The non-dominated solutions obtained by EMO-CC. (a-c) Each subplot shows non-dominated solutions in different neighborhoods, which do not compete with each other, as a consequence of the multi-modal policy followed by EMO-CC.

The obtained results reveal that there is no solution found by EMO-CC that is dominated by APRIORI, and only one solution obtained by SUBDUE dominates a solution belonging to the EMO-CC Pareto set. Moreover, EMO-CC discovers more non-dominated solutions than both APRIORI and SUBDUE.

mboxEMO-CC retrieves almost all solutions identified by the other methods and covers a wide set of all optimal solutions in the GO domain. Moreover, both APRIORI and SUBDUE obtain a limited number of non-dominated solutions in comparison with the EMO-CC methodology (Fig. 4.7. Besides, EMO-CC extracts more diverse solutions, in the objective space, than those found by APRIORI and SUBDUE. Particularly, our approach retrieves substructures of the Pareto optimal front containing few instances but harboring several features (i.e., cohesive substructures), which were undetected by the other methods. Moreover, EMO-CC finds diverse solutions in the variable space due to the niching strategy used in the non-dominance measure.

The examination of the results obtained by APRIORI and SUBDUE suggests that their deficiencies can be attributed to (i) the linearization of the database in the APRIORI method, which constraints the data representation; (ii) the thresholds used in APRIORI, which discard substructures with few members, even if they cohesively share several features; and (iii) the inflexibility caused by weighting the evaluation objectives in SUBDUE (i.e., complexity and support) into a single function, which can constrain the set of solutions to a single or limited region of the search space.

3.2 Context-dependent database compression using gene expression profiles

(a) (b)

Figure 4.9: Class #13 differential expression profile encodes genes with different behavior between treatment and control, with a similar pattern among patients. (a) Gene expression corresponding to treated patients. (b) Gene expression from patients belonging to the control group.

We use 24 gene expression profiles (Fig. 4.12), which constitute the independent classes used for validating the substructures detected by the three methods

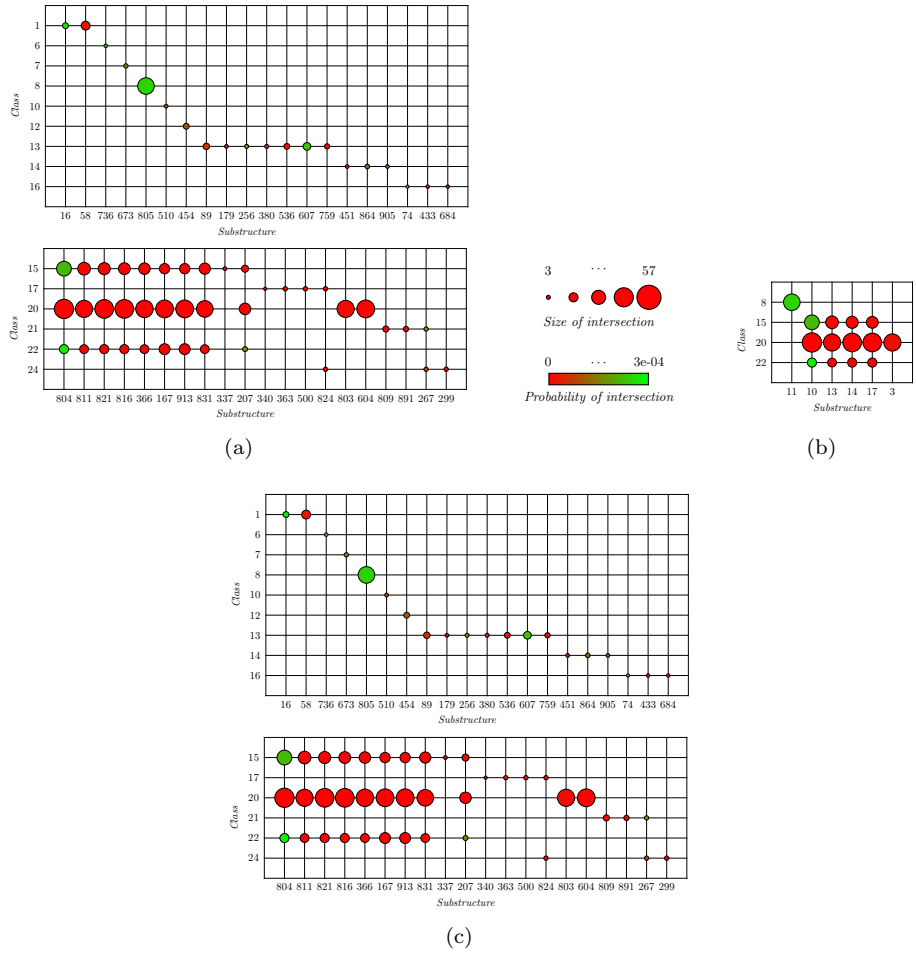


Figure 4.10: Description of gene expression profiles that are explained by GO substructures. Each intersection is represented by a circle, where the size corresponds to the number of elements in common between a differential profile and a substructure, and the color illustrates the probability of intersection (green:low, red:high). (a) APRIORI. (b) SUBDUE. (c) A subset of all EMO-CC intersections.

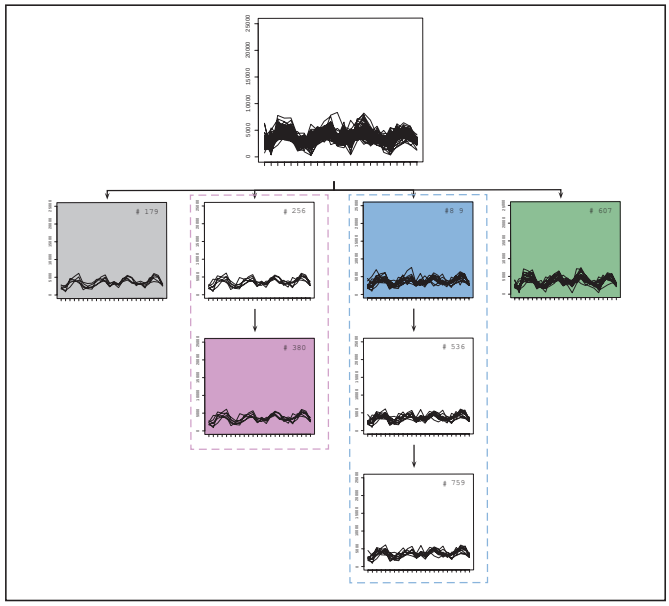


Figure 4.11: Compressed substructures that explain differential profile #13 expression profile. Differential profile #13 is explained by 7 substructures (color-coded sub-graphs show compression of substructures. These substructures are arranged by parental order in the GO database and compressed, dissecting similar expression patterns based on independent information provided by GO.

Table 4.3: Differential Profile #13 and substructure intersections

Substructure	Size	Intersection	Probability of Intersection
179	7	5	2.20×10^{-6}
536	69	12	1.52×10^{-5}
759	42	10	1.43×10^{-6}
256	22	6	1.91×10^{-4}
89	104	14	5.79×10^{-5}
380	18	6	5.43×10^{-5}
607	179	18	2.37×10^{-4}

previously described, or, in other words, which can be explained by these substructures. For example, differential profile #13 constitutes a differential gene expression profile that changes between treatment and control gene expression (Fig. 4.9). This differential profile is described by several substructures identified by EMO-CC, including substructure #89, #179, and #256, at different coincidence levels represented by the PI between differential profiles and substructures (Fig. 4.10 (c) and Table 4.3, $PI < 3.1 \times 10^{-4}$). Substructure #89 describes differential profile #13 based on a cell communication biological process located at the integral to the plasma membrane or, in a more general case, at the integral to membrane cellular component. A slightly different description is provided by substructure #256, which includes a cellular physiological process. A different example is given by substructure #179, which describes an apoptosis process (i.e., a form of programed cell death) located at the integral to plasma membrane. Significantly, these descriptions are based on different types of features (e.g., biological process and cellular components) that belong to different levels of the GO hierarchy (e.g., level 6 or level 4). These diverse substructures are optimal in the sense that they belong to the Pareto optimal set composed of specific and sensitive descriptions (Fig. 4.7).

We compare the performance of EMO-CC for extracting biologically valid substructures with APRIORI and SUBDUE. We have already seen that EMO-CC subsumes those solutions obtained by the other methods and provides novel and diverse optimal solutions (i.e., belonging to the Pareto optimal set) by the evaluation of several quantitative metrics. A qualitative evaluation of these methods reveals that EMO-CC obtains more specific substructures than the other methods for those substructures discovered in common. Moreover, the matching among substructures retrieved by EMO-CC and the independently obtained differential profiles derived from the expression profiles is better than the one achieved by the other methods. For example, substructure #5 identified by APRIORI matches with differential profile #15 with a PI of 2.1738×10^{-4}, while the corresponding PI for substructure #811 retrieved by EMO-CC is 6.9854×10^{-6} (Fig. 4.10).

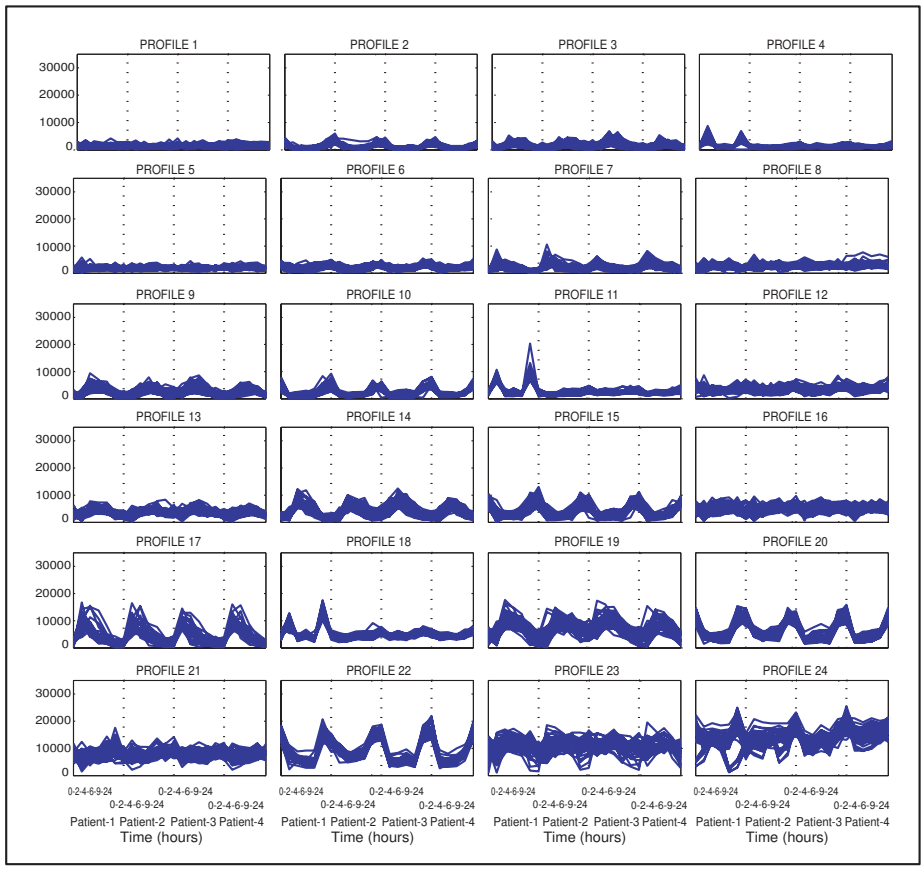


Figure 4.12: The 24 profiles P_T extracted from the treatment group.

We compress those substructures that explain the same expression profile to provide a summarized description of this phenomenon. The 24 expression profiles can be explained by 45 substructures of GO terms. For example, substructures #89 and #256, which explain differential profile #13 are compressed because they are indistinguishable for this differential profile. However, substructure #179 describes it from a very different point of view and it is preserved as a diverse solution. This compression is dynamic because substructures are re-grouped in a context-dependent fashion, where the context corresponds to an explained differential profile, and a different classification can produce a distinct substructure association (e.g., substructures #89 and #256 are indistinguishable for differential profile #13, while it may not be the case for other differential profile of microarray or clinical experiments). An emergent property of current

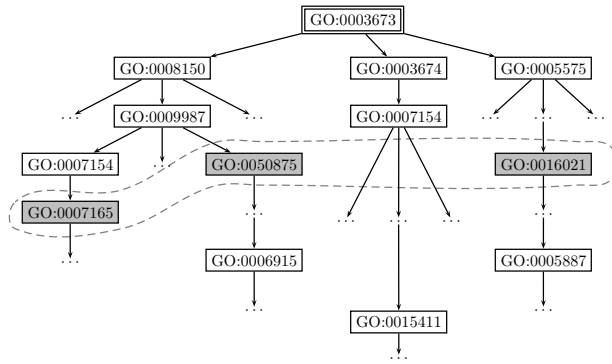


Figure 4.13: Example of a novel annotation uncovered by EMO-CC (dashed lines) based on substructure #380. The tree represents the GO hierarchy with the three sub-ontologies as the main branches (GO:0008150 "Biological Process", GO:0003674 "Molecular Function" and GO:0005575 "Cellular Component"). This annotation include GO terms from different sub-ontologies and defined at different levels of specificity.

explanations provided by the substructures retrieved by EMO-CC consists of their usefulness for differentiating even subtle expression patterns (Fig. 4.11). Notably, this classification is performed based on external information provided by the GO database, instead of the levels of expression.

The substructures identified by EMO-CC can be considered new annotations (Fig. 4.13). These annotations include different types of features defined at distinct hierarchically-organized levels of specificity, which can be used to uncover new members to the underlying substructures based on the similarity with the corresponding GO terms. Consequently, this guideline can be used to indirectly classify new members of an expression differential profile, as we will see in the next section.

Finally, we validate the GO substructures obtained by EMO-CC using a high-quality hand-curated database termed Ingenuity Pathways Knowledge Base (Systems,), which is, at the moment, a gold-standard for metabolic pathways. We queried this database with the web-based entry tool developed by Ingenuity Pathways Analysis (IPA) (Systems,). For example, by using the list of genes from differential profile #13, the best description identified by IPA (score 45, focus genes 21) functionally corresponds to an inflammatory network *Inflammatory Disease*. Moreover, *Inflammatory Disease* is the prevalent function of this network with p-values between $1.15 \times 10^{-5} - 8.83 \times 10^{3}$, suggesting that differential profile #13 and the EMO-CC substructures that explain it

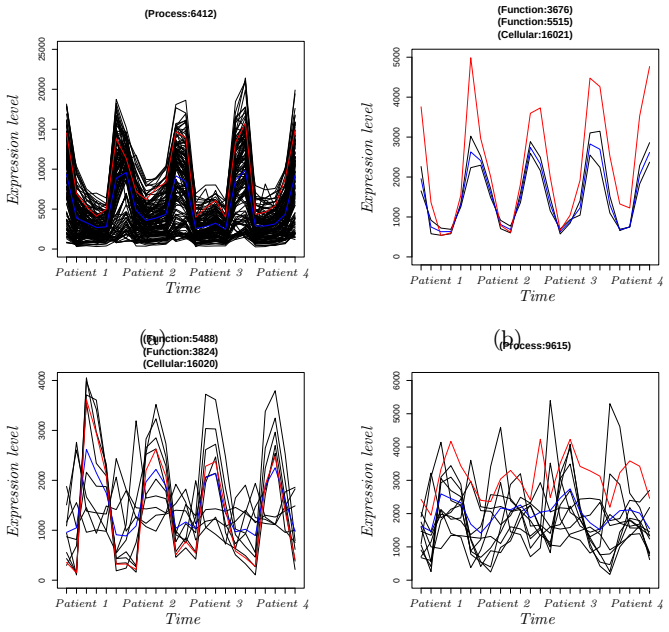


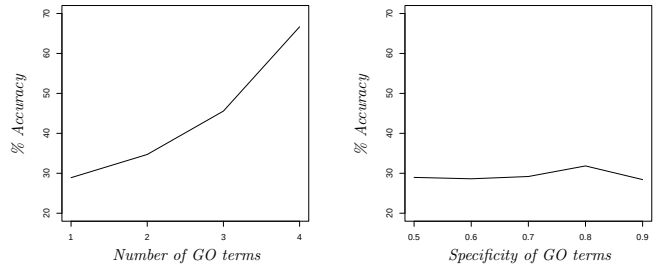
Figure 4.14: The EMO-CC inference process. The new observation classified by EMO-CC is color-coded in red within the inferred substructure, while the centroid of the substructure is color-coded in blue. Expression of the substructures that classify (a) gene `203107_x_at`, (b) gene `208982_at`, (c) gene `216316_x_at`, and (d) gene `211676_x_at`.

constitute a meaningful biological association.

3.3 Unsupervised classifier inference process

The EMO-CC methodology classifies new instances by their similarity with one or more substructures using a k-nearest neighbor unsupervised classifier. We evaluate the performance of the proposed inference process by the following procedure: (1) we divide our original gene dataset in two subsets: *training data* and *test data*, with 80% and 20% of the original dataset, respectively, selected randomly without reposition (Hand et al., 2001) to the training data only; (2) for each gene in the test set we use its GO annotation to calculate its membership to the set of the previously identified substructures in (1) and select the substructure with the highest membership value as the best predic-

tion; and (3) we test the accuracy of the inference process by: (3.1) identifying the expression differential profile explained by the selected substructure; (3.2) calculating its centroid as a weighted average of the expression values of its members (Bezdek, 1998a); and (3.3) computing the Pearson correlation (PC) (Applegate and Crewson, 2002) between the expression of the predicted gene and the centroid of (3.2).



(a) (b)

Figure 4.15: Performance of the EMO-CC inference process evaluated by considering substructures with different number of terms defined at distinct specificity levels. (a) Accuracy of the inference process evaluating the test set, where substructures contain 1 to 4 GO terms. (b) Accuracy of the EMO-CC inference process evaluating the test set, where substructures contain only one term with specificity levels from 0.5 to 0.9.

We illustrate this process by: (1) evaluating the gene `203107_x_at` from the test set (Fig. 4.14 (a)); (2) calculating its membership to the set of the previously identified substructures. Since the obtained substructures are not disjoint, a given observation may belong to more than one substructure (e.g., probe set `203107_x_at` has a membership degree greater than zero in substructure #2 (0.24), #8 (0.25), #16 (0.63), #28 (0.68), #33 (0.70), #34 (0.76) and #127 (0.91)). Therefore, we select the maximum value among the different memberships, classifying the target probe set into substructure #127. Then, we test the accuracy of the predictions by (3.2) calculating the centroid corresponding to substructure #127 (Fig. 4.14 (a), which is a cohesive profile with very similar expression pattern of its members. Afterwards, (3.3) we calculate the correlation between the gene `203107_x_at` and the former centroid ($PC > 0.6$) and evaluate the prediction as a positive matching. Similar results are observed with other genes in the test set (Fig. 4.14 (b-d)).

We evaluate the complete test set by considering substructures with at least n GO terms, where n ranges between 1 and 4. Our results indicate that 70% of the successful predictions can be achieved by using four GO terms (Fig. 4.15 (a)),

showing that the performance increases as the number of GO terms increases. However, this monotonic process is not conserved when the specificity of a given substructure is improved. For example, by increasing the specificity values of the former substructures from 0.5 to 0.9, we cannot observe an improvement in the prediction performance (Fig. 4.15 (b)). These results suggest that approaches that widely explore GO database in the complete feature space (i.e. all GO terms from biological process, molecular function and cellular component) can be appropriate for describing and predicting gene expression patterns.

The proposed testing process indicates a strategy to predict gene expression patterns based on an independent source of data such as GO terms. However, several classification errors result from ambiguous annotation terms or too general categories, as well as, missing information in the GO database rather than misclassifications (Tanay et al., 2004). Many of these problems will be solved when the GO database becomes more accurately curated.

4 Mining Genetic Sequences Databases

Repetitive DNA sequences form a substantial fraction of the genomes of many eukaryotes (Dover and Flavell, 1982). The repeats vary in complexity from simple oligonucleotides to complex sequences of several kilobases. When repeats with G+C or A+T contents significantly different from that of the main component DNA (mcDNA) are arranged in tandem in the genome, their altered density allows them to be physically separated from the mcDNA by density gradient centrifugation in solutions of cesium salts. Because such repeated DNAs can be partitioned in an "orbit" of their own in a centrifuge, they are sometimes referred to as satellite DNAs (stDNAs) (Charlesworth et al., 1994). In many cases these sequences seem to be maintained solely by their ability to replicate within the genome (the "selfish DNA" hypothesis (Orgel and Crick, 1980)). Far from conferring benefits, their behavior can sometimes result in a fitness loss to the host. Some human genetic diseases are known to be caused in this way, including mutations due to insertions of transposable elements, to chromosomal rearrangements induced by recombination between repeated sequences (Wallace et al., 1991). It has often been proposed that the repetitive sequences are functionally important for the host organism (Britten and Davidson, 1969), or are maintained because their mutagenic activities contribute to the long-term evolutionary potential of the population (Nevers and Saedler, 1977).

Therefore, we think of the study of DNA sequences databases in general, and repetitive DNAs in particular, as a very important field for research to complement the information acquired from microarray experiments, along with

the databases described in previous sections.

To properly handle the analysis of this repetitive DNAs, in particular the satellite DNA, we have developed a software package called satDNA Analyzer for the analysis of satellite-DNA sequences from aligned DNA sequence data. It allows fast and easy analysis of patterns of variation at each nucleotide position considered independently amongst all units of a given satellite-DNA family when comparing sets of sequences belonging to two different species. The program classifies each site as monomorphic or polymorphic, discriminates shared from non-shared polymorphisms and classifies each non-shared polymorphism according to the model proposed by Strachan (1985) in six different stages of transition during the spread of a variant repeat unit toward its fixation. Briefly described, class 1 site represents complete homogeneity between two species, whereas classes 2 to 4 represent intermediate stages in which one of the species shows polymorphism. The frequency of the new nucleotide variant at the site considered is low in stage 2 and intermediate in stage 3, while class 4 comprises sites in which a mutation has replaced the progenitor base in most members of repetitive family in the other species (almost fully homogenized site). Class 5 represents diagnostic sites in which a new variant is fully homogenized and fixed in all members of one of the species while the other species retain the progenitor nucleotide. Class 6 represents an additional step over stage 5 (new variants appear in some of the members of the repetitive family at a site fully divergent between two species). The program has been implemented according to the following structure:

```
FOR EACH set_of_sequences
  calculate intra_species classification
  calculate intra_species measures (average consensus sequences,
                                    average base pair contents,...)
FOR EACH pair_of_set_of_sequences
  calculate inter_species classification
  calculate inter_species measures (average consensus sequences,
                                    average base pair contends,...)
create output file
create output file excluding shared polymorphisms
```

The program implements several other utilities for satellite-DNA analysis evolution such as the design of the average consensus sequences, the average base pair contents, the distribution of variant sites, the transition to transversion rate, and different estimates of intra and inter-specific variation. Aprioristic hypotheses on factors influencing the molecular drive process and the rates and biases of concerted evolution can be tested with this program. Additionally,

satDNA Analyzer generates an output file containing an alignment to be used for further evolutionary analysis by using different phylogenetic softwares. The novelty of this feature is that it optionally discards the shared polymorphisms for the analysis, which as shown in (Navajas-Pérez, 2005), can interfere with the results when analyzing closely related species. The program generates an HTML output that can be seen in Fig. 4.16.

5 Concluding Remarks

In this chapter we have approached several biological databases using automatic mining techniques, some of them already existing and some of them developed by us. We mined into the *OMIM* and *KEGG* databases, which contain plain data, using the *Apriori* algorithm (Agrawal and Shafer, 1996b) a classic algorithm for learning association rules. The *Gene Ontology* project, a structural database, was mined using an algorithm termed *EMO-CC* (Evolutionary Multi-Objective Conceptual Clustering), proposed by us in Romero-Zaliz *et al.,* (Romero-Zaliz et al., 2007), which retrieves meaningful substructures from network databases using multi-objective and multi-modal optimization techniques (Romero-Zaliz et al., 2007). Sequence information from the *GeneBank* database related to the genes in the differential profiles from the inflammation and host response to injury problem has been studied applying a new program, satDNA analyzer, developed by us part of this PhD. work (Navajas-Pérez et al., 2007). The program automatizes the analysis of satellite-DNA sequences from aligned DNA sequence data.

Mining of *OMIM* and *KEGG* showed the biological functional cohesiveness of the differential profiles obtained from the inflammation and host response to injury problem. For instance, the probe sets grouped as profile #19 appear in the frequent item sets with *KEGG* pathways 03010, 72766 and 4160. These pathways are related to *apoptosis*, *cell cycle* and the *ribosome*. Probe sets in differential profile #19 also appear in the frequent item sets with codes from the *OMIM* database related to *ribosomal proteins* (180460*and*603636), *apoptosis* (604170) and *cytokine signaling* (604170). All these biological terms are highly related to the cellular processes related to the inflammation and host response to injury problem.

Mining of the *Gene Ontology* project also provided annotations which co-expressed probe sets and can be used to make predictions by using an independent source of information. The coincidence indexes between the differential profiles and the groups of annotations retrieved from the *EMO-CC* algorithm from genes related to the inflammation and host response to injury problem

satDNA Analyzer v1.2 - Universidad de Granada

Summary for species *S* and species *J*

Strachan Type I	490
Strachan Type II (from 0% to 25%)	210
Strachan Type III (from 25% to 50%)	22
Strachan Type IV (from 50% to 100%)	28
Strachan Type V	48
Strachan Type VI	20
Shared Polymorphism	21
Shared Polymorphisms due to indels	2
Irrelevant Positions	36
Irrelevant Positions due to indels	134
Non-shared Polymorphism	498 [210 + 22 + 28 + 48 + 20 + 36 + 134]
TOTAL	1011

```
S16  T-ATATCGCGACATAATGTATTGGATATTGTAGTATATATA--GTATCTAATAGGTAGAA
S11  T-ATATCGCGACATAATGTATTGGATATTGTAGTATATATA--GTATCTAATAGGTAGAA
S45  T-ATATCGCGACATAATGTATTGGATATTGTAGTATATATA--GTATCTAATAGGTAGAA
S9   T-ATATCGCGACATAATGTACTGGATATTGTAGTATATATA--GTATTTAATAGGTAGAA
S3   T-ATATCGCGACATAATGTATTGCATATTGTAGTATATATA--GTATCTAATAGGTAGAA
S17  T-ATATCGCGACATAATGTACTTGATATTGTAGTATATATA--GTATCTAATAGGTAGAA

J42  T-ATATTGCGACAGAATATACTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
J82  T-ATATTGCGACATAATATACTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
J7   TTATATTGCGACATAATATACTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
J66  T-ATATTGCGACATAATATACTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
J83  T-ATATTGCGACATAATATACTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
J111 T-ATATTGCGACATAATATCCTCAATGTTATAGTAAATATA--GTCTCTAATAGTTAGAA
```

Legend: Strachan Type I, Type II, Type III, Type IV, Type V, Type VI;
Shared polymorphism, Shared polymorphism due to indels, Irrelevant position;

Number of Nucleotide Differences between Species S and J (JC)

	J42	J82	J7	J66	J83	J111	J81	J23	J145	J21	J110	J72	J15	J24	J26	J76	J114	J211	J85	J78
S12	0.1899	0.1955	0.1935	0.1899	0.1899	0.1866	0.1825	0.1854	0.1972	0.1911	0.1911	0.1897	0.1868	0.1823	0.1823	0.1813	0.1768	0.1856	0.1859	0.1783
S25	0.1842	0.1911	0.1878	0.1842	0.1842	0.1809	0.1768	0.1797	0.1928	0.1854	0.1854	0.1882	0.1811	0.178	0.1766	0.1757	0.1726	0.1799	0.1816	0.1726
S29	0.1928	0.1998	0.1964	0.1928	0.1928	0.1894	0.1854	0.1882	0.1986	0.194	0.1926	0.1969	0.184	0.1837	0.1823	0.1785	0.1783	0.1885	0.1902	0.1811
S53	0.1773	0.1842	0.1809	0.1773	0.1773	0.174	0.17	0.1728	0.1859	0.1799	0.1785	0.1828	0.1757	0.1726	0.1712	0.1716	0.1714	0.1787	0.1818	0.1728
S14	0.1818	0.1859	0.1825	0.1789	0.1789	0.1757	0.1716	0.1759	0.1875	0.1801	0.1816	0.183	0.1716	0.17	0.1714	0.1704	0.173	0.1804	0.182	0.1759
S20	0.1847	0.1887	0.1854	0.1818	0.1818	0.1785	0.1745	0.1787	0.1904	0.183	0.1844	0.183	0.1745	0.1728	0.1742	0.1733	0.173	0.1804	0.1849	0.1787
S28	0.1875	0.1916	0.1882	0.1847	0.1847	0.1813	0.1773	0.1816	0.1933	0.1844	0.1873	0.1887	0.1759	0.1728	0.1771	0.1747	0.1787	0.1861	0.1878	0.1816
S57	0.1818	0.1859	0.1854	0.1818	0.1818	0.1785	0.1745	0.1787	0.1875	0.1816	0.183	0.1873	0.1759	0.1742	0.1728	0.1747	0.1745	0.1847	0.1849	0.1801
S19	0.1832	0.1873	0.184	0.1804	0.1804	0.1771	0.173	0.1773	0.189	0.1787	0.183	0.1844	0.1759	0.1714	0.1728	0.1719	0.1745	0.1818	0.1835	0.1773
S10	0.1847	0.1887	0.1824	0.1819	0.1819	0.1786	0.1746	0.1788	0.1904	0.1859	0.1845	0.1831	0.1733	0.173	0.1744	0.1735	0.1746	0.1819	0.1849	0.1788
S1	0.176	0.18	0.1768	0.1733	0.1733	0.17	0.1661	0.1703	0.1817	0.1772	0.1758	0.1772	0.1703	0.1645	0.1659	0.1649	0.1675	0.1746	0.1763	0.1703
S62	0.1863	0.189	0.1873	0.1834	0.1819	0.1803	0.1762	0.1805	0.1951	0.1907	0.1892	0.1861	0.1819	0.1774	0.1788	0.1776	0.1895	0.1878	0.1895	0.1831
S27	0.1831	0.1857	0.1838	0.1802	0.1802	0.177	0.173	0.1772	0.1887	0.1828	0.1828	0.1828	0.173	0.1687	0.17	0.1691	0.1689	0.176	0.1777	0.1716
S16	0.1744	0.1812	0.178	0.1744	0.1744	0.1712	0.1673	0.1673	0.1828	0.177	0.1756	0.177	0.1673	0.1657	0.1643	0.1634	0.1687	0.1758	0.1774	0.17
S11	0.1746	0.1814	0.1782	0.1746	0.1746	0.1714	0.1675	0.1689	0.1831	0.1786	0.1772	0.1786	0.1675	0.1659	0.1645	0.1636	0.1689	0.176	0.1777	0.1716
S45	0.1747	0.1816	0.1783	0.1747	0.1747	0.1714	0.1674	0.1716	0.1832	0.1787	0.1773	0.1787	0.1716	0.1686	0.1672	0.1662	0.1688	0.1761	0.1777	0.1702
S9	0.1733	0.1801	0.1768	0.1733	0.1733	0.17	0.166	0.1702	0.1818	0.1773	0.1745	0.1773	0.1702	0.1658	0.1644	0.1648	0.1674	0.1747	0.1763	0.1702
S3	0.1751	0.1791	0.1758	0.1709	0.1709	0.1691	0.1651	0.1693	0.1793	0.1749	0.1735	0.1749	0.1679	0.1649	0.1649	0.164	0.1638	0.1723	0.1739	0.1679
S17	0.1788	0.1828	0.1782	0.176	0.176	0.1728	0.1689	0.173	0.1831	0.1786	0.1772	0.18	0.173	0.17	0.1687	0.1677	0.1703	0.176	0.1777	0.173

Average Number of Nucleotide Differences: S and J (K): 142.1
Average Number of Nucleotide Substitution per Site between Species S and J (Dxy): 0.1587
Average Number of Nucleotide Substitutions between Species S and J (Dxy(JC)): 0.1783
Number of Net Nucleotide Substitutions per Site between Species S and J (Da): 0.1135
Number of Net Nucleotide Substitutions per Site between Species S and J (Da(JC)): 0.1315
The expected λ/T value is: 2.837e-08
The expected λ/T value(JC) is: 3.288e-08

Figure 4.16: Example of the output of the satDNA software.

is significative, as seen in Fig. 4.10. For instance, one of the differential profiles, #13, is highly associated (p-values for the coincidence between $1.15x10^{-5}$ and $8.83x10^{-3}$) to several inflammation related terms, such as the *inflammation process* annotation.

We also saw how the satDNA Analyzer software package automatizes the analysis of satellite-DNA sequences from aligned DNA sequence data. It allows fast and easy analysis of patterns of variation at each nucleotide position considered independently amongst all units of a given satellite-DNA family when comparing sets of sequences belonging to two different species, as well as implementing several other utilities for satellite-DNA analysis evolution such as the design of the average consensus sequences, the average base pair contents, the distribution of variant sites, the transition to transversion rate, and different estimates of intra and inter-specific variation. Aprioristic hypotheses on factors influencing the molecular drive process and the rates and biases of concerted evolution can be tested with this program.

Chapter 5

Modeling Genetic Networks

Gene expression is determined by protein-protein interactions among regulatory proteins and with RNA polymerase(s), and protein-DNA interactions of these transacting factors with cis-acting DNA sequences in the promoters of regulated genes (Kaern, 2003). These interactions define complex genetic networks, whose designs have motivated researchers to draw direct analogies with established techniques in electrical engineering (Guet et al., 2002; Hasty et al., 2001). As with the construction of electrical circuits, the gene circuit approach uses mathematical and computational tools in the construction and posterior analysis of a proposed network diagram. The qualitative agreement between model and experiment in a series of studies depends both on the design of the network topology, which most of the times includes uncertain connections between genes, as well as on the dynamic behavior of the network, which is affected by the ambiguity inherent to the biological processes (e.g., monomer or dimmer binding of promoters, enzymes having kinase and/or phosphatase activities, etc.) and the mathematical models used to represent them (e.g., Boolean or continuous models; reverse or forward algorithms) (van Someren et al., 2002). Moreover, the number of genes considered in the networks is usually large compared to the number of the available measurements (e.g., time-point expression). Therefore, more than one possible model may be consistent with the subjacent data (Wahde et al., 2001). Finally, the data always contains a substantial amount of noise (Li and Wong, 2001b; Li and Wong, 2001a; McAdams and Shapiro, 2003) provided by the systematic

variability of the experiments (Nadon and Shoemaker, 2002), which in addition to previous problems, makes it difficult to deduce the implications of the underlying logic of genetic networks through experimental techniques alone.

Systems biology research arises at this point as the field to explore the life regulation processes in a cohesive way making use of the new technologies. Proteins have a main role in the regulation of genes (Rice and Stolovitzky, 2004), but unfortunately, for the vast majority or biological datasets available, there is no information about the level of protein activity. Therefore, we use the expression level of the genes as an indicator of the activity of proteins they generate. Gene networks represent these gene interactions. A gene network can be described as a set of nodes which usually represent genes, proteins or other biochemical entities. Node interaction is represented with edges corresponding to biologic relations.

There is a wide range of models available to build genetic networks up. One of the differences between such models is whether they represent static or dynamic relations. Static modeling explains causal interactions by searching for mutual dependencies between the gene expression profiles of different genes (van Someren et al., 2002). Clustering techniques are widely applied for static genetic network, since they group genes that exhibit similar expression levels. In dynamic modeling, the expression of a node A in the network at time t_{+1} can be given as the result of the expression of the nodes in the network with edges related to A at time t (van Someren et al., 2002). The understanding of the relations helps to describe all the relations occurring in a given organism. Temporal studies are becoming widely used in biomedical research. In fact, over 30% of published expression data sets are time series (Simon et al., 2005).

The question arises as which network model is the most appropriate given a set of data. In this work we compare the behavior of static vs. dynamic modeling. We have used an static network building method, (K-means clustering (Duda and Hart, 1973)) and dynamic network models (a Boolean method, described in (D'Onia et al., 2003)) and implemented in (Velarde, 2006) and a graphic Gaussian method (GGM) (Schäfer and Strimmer, 2005).

We now describe in detail each of these three genetic network build up methods and show the results obtained by application of them to a problem derived from inflammation and host response to injury (Calvano et al., 2005). In Fig. 5.1 we show the schema of the proposed methodology including the genetic network build up step. The information already obtained in previous steps of the methodology proposed will be fused with information obtained from the genetic networks obtained.

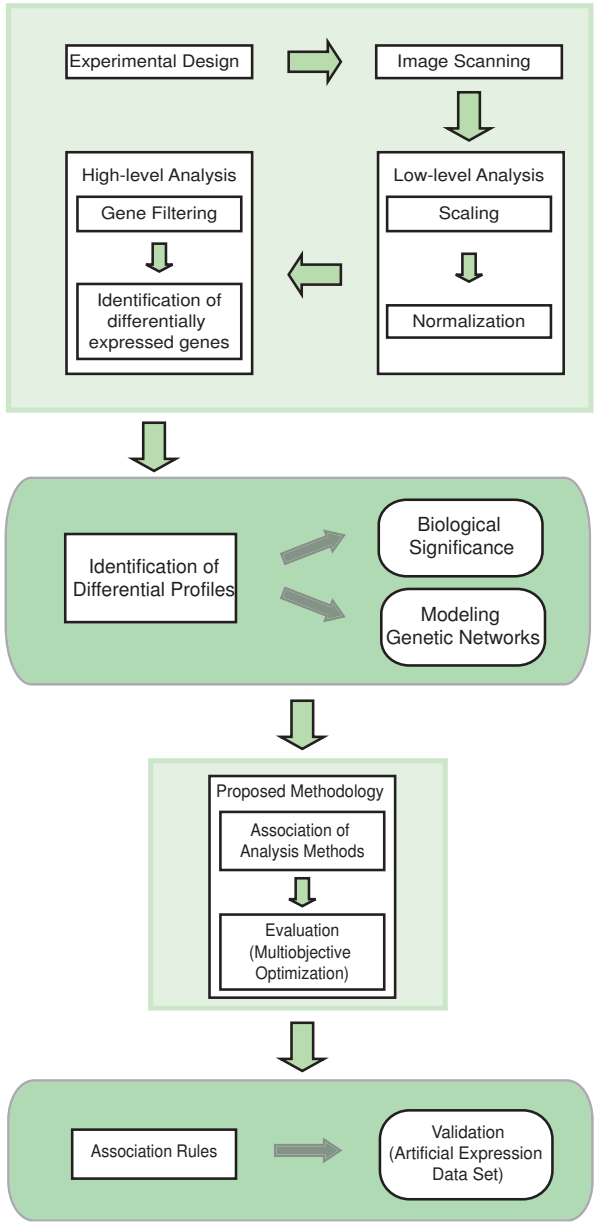


Figure 5.1: Schema of the methodology proposed including modeling of genetic networks

1 Genetic Network Construction

We have applied both static and dynamic models to a set of data obtained from an inflammation and host response to injury problem. Static modeling explains causal interactions by searching for mutual dependencies between the gene expression profiles of different genes (van Someren et al., 2002). The relation found by static methods might not only be similar behavior throughout time (direct correlation), but an inverse correlation (two genes having exactly opposite profiles over time), or a proximity on the expression values (distance measures such as Euclidean Distance or City block distance). Dynamic modeling retrieves temporal dependencies among genes, i.e., it detects dependencies of a gene at time t_{+1} related to some other(s) gene at time t (see Fig. 5.2).

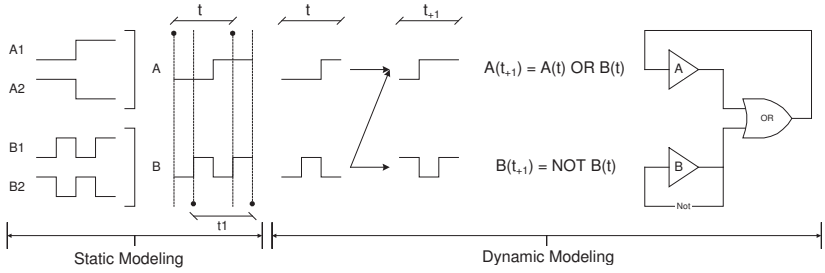


Figure 5.2: The static modeling captures the relation (inverse correlation) between A_1 and A_2 (profile A) and between B_1 and B_1 (profile B). However, it does not capture the relation between A and B describing profile A at time t_{+1} as dependent on the behavior of profiles A and B at time t. This relation is only captured by the dynamic model.

Clustering techniques are widely applied for static genetic network. We have used a classic clustering algorithm based on Euclidean distance, the K-means (Duda and Hart, 1973) which is a very popular clustering algorithm widely used on data from microarray experiments (A. et al., 2006). Two dynamic methods have been applied as well: a Boolean method, described in (D'Onia et al., 2003) and implemented in (Velarde, 2006) and a graphic Gaussian method (GGM) (Schäfer and Strimmer, 2005). These two methods have been chosen as representation of discrete and continuous models respectively, the two big families in which dynamic models can be divided (van Someren et al., 2002). We now describe each of them.

1.1 Static Models

Classification of gene expression patterns to explore shared functions and regulation can be accomplished using clustering methods (D'haeseleer et al., 2000). In one of the phases of our methodology, we have already applied a clustering algorithm based on Euclidean distance, the K-means algorithm (Duda and Hart, 1973) (see Section 1.1). The number of resulting clusters k is estimated by application of the *Davies-Bouldin* validity index (Davies and Bouldin, 1979). This index detects compact representations of K-means partitions (Bezdek, 1998b) by choosing the cluster size c that minimizes the following formula through different number of clusters (i.e., $c = 2$ to $c = \sqrt{a}$):

$$DB(U, \overline{V}, X) = (\frac{1}{c}) \sum_{i=1}^{c} [max_{j.j=1} \left\{ \frac{\alpha_i + \alpha_j}{\|\overline{v_i} - \overline{v_j}\|} \right\}] \tag{5.1}$$

where the dataset is partitioned as: $X = U_{i=1}^{c} X_i; |X_i| = n_i; X_i \cap X_j = \emptyset$ for $i \neq j; \| \|$ is the Euclidean norm; each centroid is defined as: $\overline{v}_i = \sum_{x \in X_i} \frac{x}{n}$ for each X_i; the total cluster centroids are calculated as: $\overline{V} = \{\overline{v}_1, \ldots, \overline{v}_c\}$ and $\alpha_i = \sum x \in X_i \frac{\|x - \overline{v}_i\|}{|X_i|}$.

From the partitions obtained from application of the K-means algorithm to our dataset we obtain some non-overlapping clustering of data for the treatment and for the control experimental groups. This groupings are the entry to build up the dynamic models. From each the groupings obtained we will keep the centroid, as representation of each cluster, since the genetic network building algorithms can not deal with the number of probe sets we are working with. For generality purposes, the centroid will summarize information of all patients from an experimental group. The representation used is graphically represented in Fig. 5.3

1.2 Dynamic Models: Boolean Networks

A Boolean network is composed by a set of nodes n which represent genes, proteins or other biochemical entities. These nodes can take on/off values. The net is determined by a set of at maximum n boolean functions, each of them having the state of k specific nodes as input, where k depends on each node. Therefore, each node has its own boolean function which determines the next

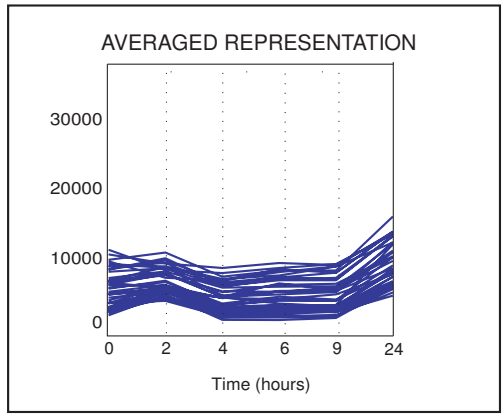


Figure 5.3: Averaged graphical representation

NOT A = A NAND A
A AND B = (A NAND B) NAND (A NAND B)
A OR B = (A NAND A) NAND (B NAND B)

Table 5.1: Boolean functions obtained only using the $NAND$ function.

state (state at time t_{+1}) based on the actual state (state at time t) of the input nodes. The changes in the net are assumed to occur at discrete time intervals.

The algorithm applied to build the Boolean network with our data is the GeneYapay (D'Onia et al., 2003). It performs an exhaustive search of boolean functions over the data, where a number of nodes, less or equal than k, univocally determine the output of some other gene. All possible subsets of $1, 2, ..., k$ elements are visited calculating the number of inconsistencies of the boolean functions in relation to the output value of each gene. The algorithm stops the search for each node when a subset of nodes is found which defines the expression profile. The implementation applied (Velarde, 2006) only uses the $NAND$ function since all other Boolean function -AND, OR, NOT- can be expressed using $NAND$ (see Table 5.1).

Boolean building network algorithms use discrete data which take two possible values: on or off, i.e., 1 or 0. Therefore, the set of profiles to be used needs to be transformed to fit the binary scheme. They are scaled in the $[0, 1]$ interval according to the maximum value scored in the expression level of such profile throughout the six time points stored. The individual scaling has been used instead of a global one (scaling the 24 profiles according to the global maximum)

since the profiles fluctuate in different levels of expression. For instance, profile #1 takes values between 1224.2 and 1724.4, while profile #24 changes between 13632 and 16436. If we scaled all values together, the variations between the expression values in profile #1 would result to small to be traceable, although they could be significative.

Once the values are scaled in the $[0, 1]$ interval we have assigned them [0] or [1] values. The simplest approach is to establish a threshold value, for instance 0.5, and set each of the time points to either [0] or [1] depending whether they are over/under the threshold. The obvious problem with this approach are the "border values", such as 0.45 or 0.55. These will be set [0] and [1] respectively, but they are so close to each other that they should take the same value. Our approach consists in setting the value based on the proximity to the expression level in the previous time point, which solves the previously described problem and captures the behavior of the profile over time. The scheme used to set the values is:

$$\begin{Bmatrix} if(\mid t - t_{+1} \mid \leq \delta) & \text{then} & t_{+1} = t \\ if(\mid t - t_{+1} \mid > \delta) & \text{then} & \begin{Bmatrix} if(t - t_{+1} \leq 0) & \text{then} & t_{+1} = 0 \\ if(t - t_{+1} > 0) & \text{then} & t_{+1} = 1 \end{Bmatrix} \end{Bmatrix} \quad (5.2)$$

where t_{+1} is the gene value to be set and t is the gene value in the previous time point.

1.3 Dynamic Models: Graphic Gaussian Network

The graphical Gaussian models were first proposed by Kishino and Waddell (2000) for the association structure among genes. GGMs are similar to Bayesian networks in that they allow to distinguish direct from indirect interactions (i.e. whether gene A acts on gene B directly or through a third gene C). As any graphical model, they also provide a notion of conditional independence of two genes. However, in contrast to Bayesian networks, GGMs contain only undirected rather than directed edges. This makes graphical Gaussian interaction modeling on the one hand conceptually simpler, and on the other hand more widely applicable (e.g. there are no problems with feedback loops as in Bayesian networks).

The GGM applied in this work has been developed by Schäfer and Strimmer (2005), and is based on (1) improved (regularized) small-sample point estimates

of partial correlation, (2) an exact test of edge inclusion with adaptive estimation of the degree of freedom and (3) a heuristic network search based on false discovery rate multiple testing.

2 Results

High-throughput techniques provide great amounts of data that need to be processed before using them to build genetic networks up. The first step is the identification of genes relevant for the problem under study. We have applied the methodology described in Chapter 3. The proliferation of related microarray studies by independent groups, and therefore, different methods, has lead to the natural step of combination of results (Guigo and Consortium., 2007). Thus, a battery of analysis methods has been applied (Student's T-Tests (Li and Wong, 2003), Permutation Tests (Tusher and Chu, 2001), Analysis of Variance (Park et al., 2003) and Repeated Measures ANOVA (Der and Everitt, 2001)). A total of 2155 genes have been identified as relevant for the problem under study. For this particular problem the number of genes retrieved is very high compared to other microarray experiments, since the problem under study, inflammation and host response to injury, is a process that affects the human system in a global manner, hence altering the behavior of a large number of genes (Calvano et al., 2005).

At the view of these, we decide to use the expression profiles of the genes obtained as significant from the treatment group as the input for the genetic network building algorithms, since the number of genes involved in the problem is unfeasible for both building and analyzing the genetic networks. The set of profiles used is the one obtained from the static model applied, the K-means algorithm.

2.1 Static Models

We apply a clustering method, the K-means algorithm, as described in Section 1.1. We have identified 24 expression profiles (see Chapter 3) (see Fig. 5.4). These profiles have been proved as functionally cohesive by fusing the information from the expression profiles with information obtained from mining into biological databases Chapter 4. For instance, the majority of the genes exhibiting profile #22 are related to the inflammatory response (GO:0006954) and are annotated as intracellular (GO:0005622). Another sample is profile #16, with

genes sharing the *apoptosis* (GO:0006915) and *integral to plasma membrane* (GO:0005887) annotations.

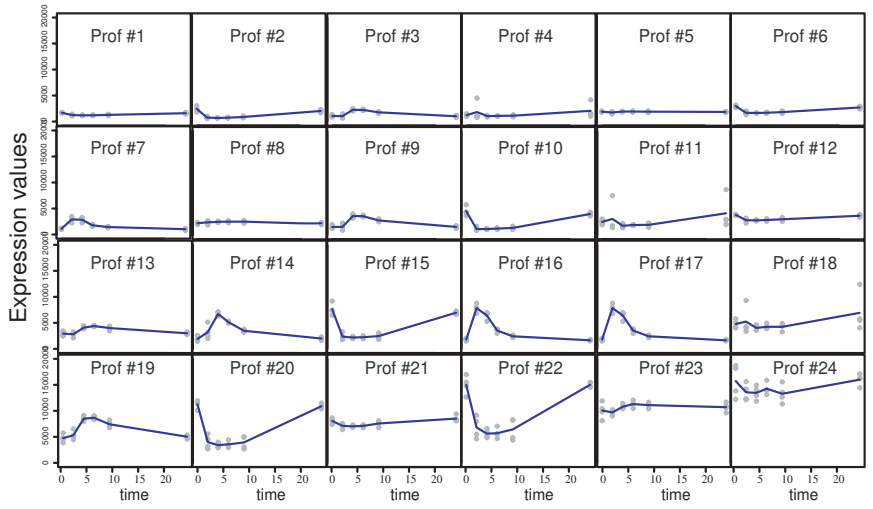


Figure 5.4: Set of 24 expression profiles obtained from the inflammation and host response to injury problem.

The groupings obtained using this method (i.e., gene expression profiles), are expected to be functionally cohesive since genes sharing the same expression profiles are likely to be involved in the same regulatory process (D'haeseleer et al., 2000). . This has been already proved in Chapter 4 by mining into several biological databases and fusing the information obtained with information from the differential profiles obtained from the inflammation and host response to injury problem.

2.2 Dynamic Models: Boolean Network

Boolean building network algorithms use discrete data which take two possible values: on or off, i.e., 1 or 0. Therefore, the set of 24 differential profiles obtained in the inflammation and host response to injury problem (Calvano et al., 2005) needs to be transformed to fit the binary scheme. The expression levels before scaling are shown in Table 5.2 column (A), and (B) we show the obtained boolean values for the 24 profiles in our problem.

Once the values are scaled in the $[0, 1]$ interval we have assigned them [0]

Profiles	CONTINUOUS VALUES (A)						BOOLEAN VALUES (B)					
	T_0	T_2	T_4	T_6	T_9	T_{24}	T_0	T_2	T_4	T_6	T_9	T_{24}
#1	1724.4	1316.4	1224.2	1236.9	1327.5	1666.1	1	0	0	0	0	1
#2	2546.2	734.44	700.28	737.51	867.44	2107.8	1	0	0	0	0	1
#3	1108.8	1027.9	2403.2	2376.1	1843.3	1069.6	0	0	1	1	0	0
#4	1323.6	2001.9	1089.4	1139.8	1192.7	2230.8	0	1	0	0	0	1
#5	1933.1	1829.8	1970.5	1983.6	1966.4	1907.5	1	0	1	1	1	1
#6	3146.1	1694.2	1669.1	1746.3	1889.8	2872.3	1	0	0	0	0	1
#7	1265.8	3551.7	3079.1	2008.1	1656.4	1160.3	0	1	1	0	0	0
#8	2396.3	2577.6	2721.5	2726.6	2712.1	2412.9	0	1	1	1	1	0
#9	1614.2	1619.1	3756.4	3972.6	3116.5	1676.8	0	0	1	1	1	0
#10	4844.2	1278.3	1248.4	1316.9	1468.1	4240.1	1	0	0	0	0	1
#11	2730.3	3351.4	1921.3	2114.9	2146.3	4459.3	0	1	0	0	0	1
#12	4176.1	2984.1	2974.1	3068.7	3265.5	4021.8	1	0	0	0	0	1
#13	3022.8	2898.1	4262.2	4666.1	4329.1	3150.8	0	0	1	1	1	0
#14	2117.6	3289.7	7298.8	5871.3	4036.8	2229.4	0	0	1	1	0	0
#15	7849.5	2328.1	2297.4	2450.1	2738.6	7171.7	1	0	0	0	0	1
#16	4836.6	4220.5	5085.4	5398.3	5356.3	4829.7	1	0	1	1	1	0
#17	1950.7	9001.6	7946.1	4268.8	2804.1	1787.1	0	1	1	0	0	0
#18	5238.2	5734.5	4445.8	4654.6	4665.7	7584.4	1	0	0	0	0	1
#19	4935.7	5335.4	9034.5	9171.1	7858.1	5285.3	0	0	1	1	1	0
#20	11615	4161.2	3578.6	3760.8	4149.9	11344	1	0	0	0	0	1
#21	8358.3	7308.8	7244.2	7652.2	8139.2	8913.8	1	0	0	0	0	1
#22	15442	7021.5	5798.9	5918.8	6632.3	15605	1	0	0	0	0	1
#23	10473	10132	11396	11871	11531	10980	0	0	1	1	1	1
#24	16095	13749	13632	14364	13741	16436	1	0	0	1	0	1

Table 5.2: Continuous and Boolean values obtained for each of the 24 profiles in the data set.

or[1] values. The simplest approach is to establish a threshold value, for instance 0.5, and set each of the time points to either [0] or [1] depending whether they are over/under the threshold. The obvious problem with this approach are the "border values", such as 0.45 or 0.55. These will be set [0] and [1] respectively, but they are so close to each other that they should take the same value. Our approach consists in setting the value based on the proximity to the expression level in the previous time point, which solves the previously described problem and captures the behavior of the profile over time. The scheme used to set the values is:

$$\begin{cases} if(\mid t - t_{+1} \mid \leq \delta) & \text{then} & t_{+1} = t \\ if(\mid t - t_{+1} \mid > \delta) & \text{then} & \begin{cases} if(t - t_{+1} \leq 0) & \text{then} & t_{+1} = 0 \\ if(t - t_{+1} > 0) & \text{then} & t_{+1} = 1 \end{cases} \end{cases} \quad (5.3)$$

where t_{+1} is the gene value to be set and t is the gene value in the previous time point.

The resulting boolean network, obtained by application of the algorithm described in (D'Onia et al., 2003)) and implemented in (Velarde, 2006), is shown in Fig. 5.5. This net is the result of an exhaustive search of boolean functions over the data which univocally determines the output of the other genes. We see that some nodes represent more than one expression profile. This is due to the processing the data has to undergo. The scaling of the data to the $[0, 1]$ interval, makes profiles at different levels of expression end up sharing a common boolean profile. A sample of this in our particular problem are profiles #9, #13 and #19. These three expression profiles share similar behavior throughout time at different levels of expression (see Fig. 5.6).

The net shows valuable information about relation between profiles. For instance, the relation established between profiles #7 and #17 with profiles #3 and #14 is confirmed when searching in *KEGG* (Kanehisa et al., 2002), a metabolic pathway database. Genes exhibiting profiles #7 and #17 are in the same pathway and regulate genes exhibiting profile #14 (See Fig. 5.7). That is the case of gene *IL1RN* (prof. #17, Interleukin-1 receptor antagonist protein precursor), related to the immune response (GO:0006955) and gene *IL1R2* (prof. #14, Interleukin-1 receptor type II), also related to the immune response.

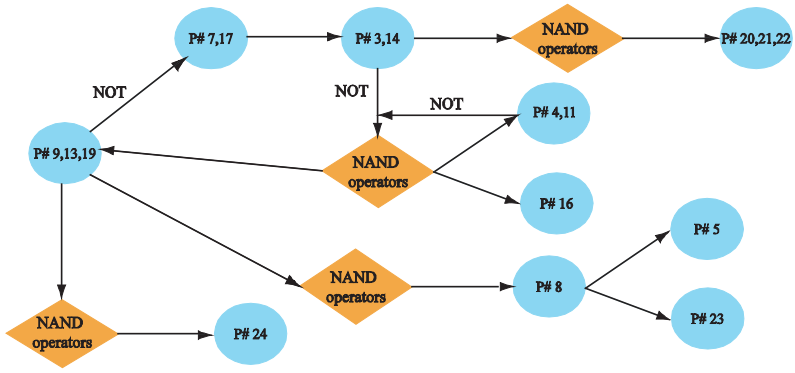


Figure 5.5: Genetic network obtained using the Boolean model. The round nodes represent the gene expression profiles (groups of genes with a common behavior) and the diamond shape nodes represent the Boolean function based on the $NAND$ operator. Note that some nodes represent more than one expression profile.

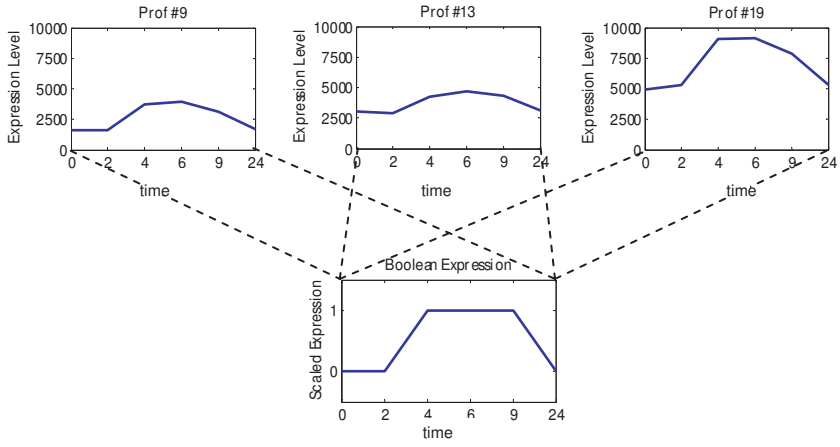


Figure 5.6: Profiles at different levels of expression but sharing a common behavior throughout time, and therefore sharing the same Boolean profile.

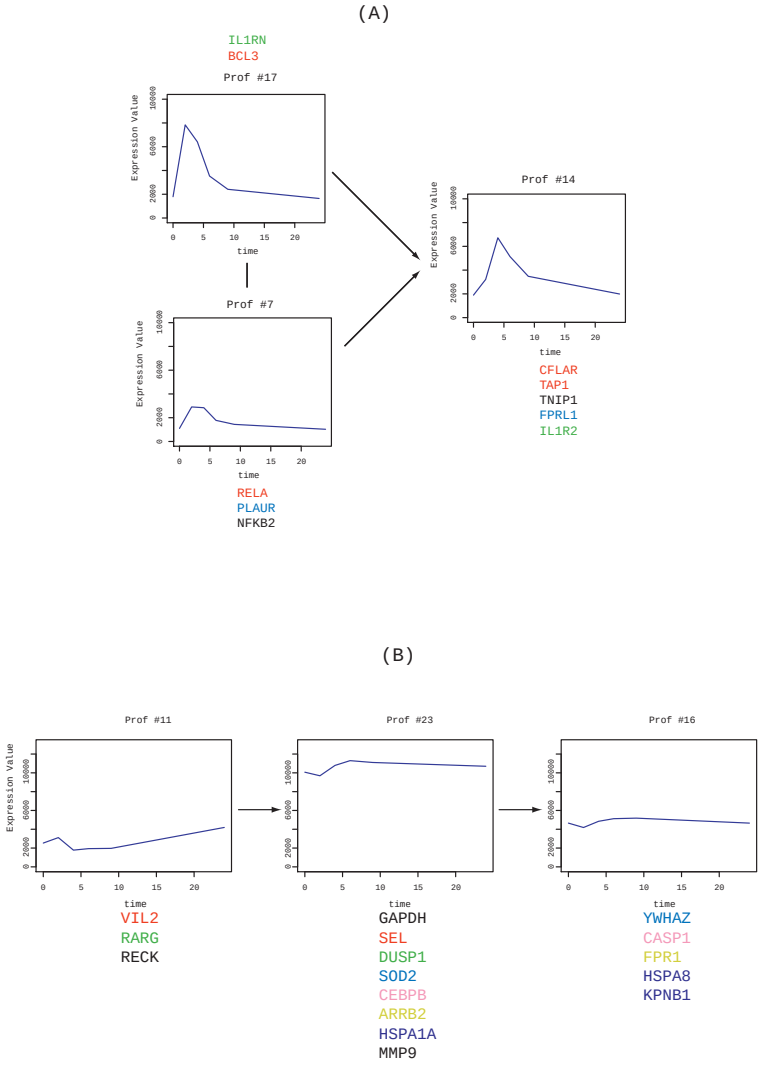


Figure 5.7: Gene relations detected by the network building algorithms and confirmed in the KEGG database. (A) has been found by both the Boolean algorithms and GGM while (B) has only been found by GGM. Each gene regulates genes with the same color.

2.3 Dynamic Models: Graphical Gaussian Model

We have applied a Graphic Gaussian Model algorithm (Schäfer and Strimmer, 2005), which takes as input continuous data that can be in longitudinal format (R. and K., 2006), very convenient for microarray time course experiments since it deals with repeated measurements, irregular sampling, and unequal temporal spacing of the time points. To select the edges, and thus the nodes, we have used the local false discovery rate *(fdr)* (expected proportion of false positives among the proposed edges), an empirical bayes estimator of the false discovery rate (Efron, 2005). An edge is considered present or significant if its local *fdr* is smaller than 0.2 (Efron, 2005). Three independent networks are found (see Fig. 5.8). In Fig. 5.8, network (B) confirms the information provided by the boolean network about profiles #7, #14 and #17. In Fig. 5.8 network (A) there is a relation established between profiles #11, #23 and #16 that is confirmed when searching in *KEGG* (see Fig. 5.7(B)). That is the case of gene *RACK* (Reversion-inducing cysteine-rich protein with Kazal motifs), which exhibits profile #11 and is related to gene *MMP9* (Matrix metalloproteinase-9), which exhibits profile #23. Both genes are related to the inflammation problem. Another relation is found between a gene exhibiting profile #23, *CEBPB* (CCAAT/enhancer-binding protein beta), related to the immune response (GO:0006955) and to the; inflammatory response (GO:0006954) and a gene exhibiting profile #16, *CASP1* (Caspase-1) related to apoptosis (GO:0006915).

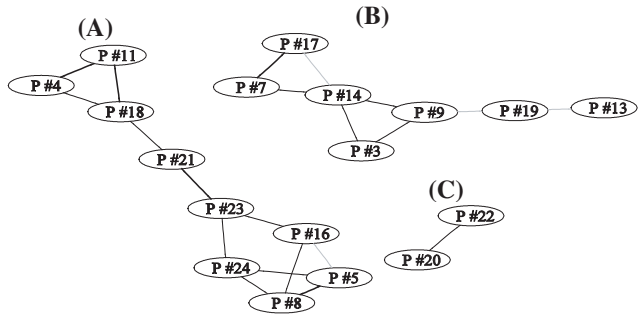


Figure 5.8: Three independent networks found by the GGM algorithm.

The GGM net we see that profiles #5, #8 and #23 are related since they are in the same subnet, but the boolean network specifically describes the behavior of those profiles: #8 determines the behavior of both #5 and #23 (see Fig. 5.9)

since the behavior shown by profile #8 is shifted over time in profiles #5 and #23. This kind of information is only available in network models which strongly stress the temporal dependencies, as it is the case with boolean networks.

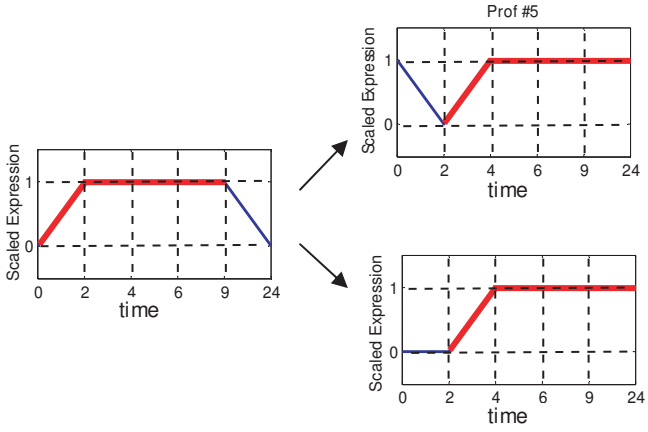


Figure 5.9: Time relations found by the Boolean algorithm. Profile #8 determines the behavior of profiles #6 and #23.

However, boolean algorithms lack the capacity to distinguish among expression profiles with similar behavior throughout time at different levels of expression (see Fig. 5.6). For instance, the boolean algorithm considers profiles #9, #13 and #19 as only one node. GGM uses continuous values solving this problem and taking advantage of the diversity or the data, but in spite of this it misses some information. The network (C) in Fig. 5.8 provided by GGM covers profiles #20 and #22. In the boolean network they are considered as one single profile along with #21, since their boolean representation is the same. GGM has not been able to capture the similarity between these three profiles, only between two of them, #20 and #22. However, the boolean model considers them as the same node, so any temporal relation between them is impossible to capture. In fact, when searching in *KEGG* (Kanehisa et al., 2002), a metabolic pathway database, we see that one of the genes that exhibit profile #20 is *NFKB2* (nuclear factor of kappa light polypeptide gene enhancer in B-cells 2) and one of the genes exhibiting profile #22 is *TNIP1* (TNFAIP3-interacting protein 1). When searching for information about these two genes, which are related in their behavior, we see they are also functionally related since *TNIP1* interacts with zinc finger protein A20/TNFAIP3 and inhibits TNF-induced NF-kappa-B-dependent gene expression (*NFKB2*). This valuable information is only prone to be found with network models such as GGM which permit the representation of temporal dependencies among strongly correlated profiles.

3 Concluding Remarks

In this chapter we have applied both static and dynamic methods for the analysis of a data set derived from the inflammation and the host response to injury (Calvano et al., 2005). The static method has been the K-means clustering algorithm, and the dynamic methods have been a discrete one, boolean model described in (D'Onia et al., 2003) and implemented by (Velarde, 2006), and a continuous one, Graphic Gaussian Model developed by (Schäfer and Strimmer, 2005).

We have already described some of the findings these methods have made on the dataset: the static method is capable of grouping the genes based on their behavior throughout time and these groupings are cohesive in biological functionality. The dynamic models provide temporal relations between the genes, or in this case, between the profiles they exhibit, organizing them in regulatory networks which have been validated using the *KEGG* database. These temporal relations would not have been found only applying static models.

When comparing the two dynamic models, we see that they cross-validate in general their results (i.e., the profiles involved and the relations between those profiles are concordant with one another). The boolean algorithm and GGM show different and complementary information about the problem under study. In a GGM network the relation between nodes is based on the levels of correlation but the time dependency is not so clearly pointed out as in boolean networks.

However, boolean algorithms lack the capacity to distinguish among expression profiles with similar behavior throughout time at different levels of expression (see Fig. 5.6). From the work performed we conclude that static models provide very valuable information but a step further is needed to get a deeper knowledge of the problem under study. Dynamic models provide information of the temporal dependencies in the data what is very valuable especially for time-course experiments, which are becoming very popular used in biomedical research. Dynamic discrete models miss valuable information when discretizing the data, while the continuous models do not suffer this problem. However, dynamic continuous models are not capable to find some of the dependencies that discrete model discover and vice versa. Therefore, they are complementary methods and it is a recommendable practice to apply both models to experiments to extract the maximum information possible from it.

Concluding Remarks

In the last part of this work we present the results obtained in this PhD. work and some conclusions derived from them, as well as some future works and a list of published papers about the topics in this thesis.

I Results and Conclusion

In this work we have proposed a methodology for the broad and complete analysis of gene expression data coming from microarray experiments. Our methodology is an integrated resource that enables researchers to extract results from experiments carried out applying the microarray technology. We have provided a computational framework for identifying reliable targets providing statistically meaningful results and characterizing novel expression patterns. This environment also fuses genetic information from different sources including experimental knowledge and biological databases. This integrated analysis suite allows us to sort the information acquired by functional groups, gene interaction through time, metabolic pathways, disease associations and DNA sequence information.

The research work has been performed based on data acquired from an experiment carried out over an inflammation and host response to injury problem. Inflammation is a hallmark of many human diseases. Understanding the inflammation process is critical because the body uses inflammation to protect itself from infection or injury (e.g., crushes, massive bleeding, or a serious burn) which, in extreme cases (e.g., car accidents or gun shootings), can lead to massive organ malfunction and death. According to the U.S. Centers for Disease

Control's National Center for Health Statistics, unintentional injury is the leading cause of death for people ages 1 to 35. The host response to trauma and burns is a collection of biological and pathological processes that depends critically upon the regulation of the human immuno-inflammatory response. This study, in part carried out at the Cellular Injury and Adaptation Laboratory, Washington University School of Medicine, is a piece of a Large-scale Collaborative Research Project sponsored by the National Institute of General Medical Sciences (www.gluegrant.org)

Along with the relevance of the biological problem under study, analysis of the set of gene expression profiles obtained from this experiment has been complex, given the number of samples taken and variance due to treatment, time, and subject phenotype. Therefore, we believe this problem is typical and informative as a RNA microarray case study.

Some of the objectives achieved throughout the performance of this work have been:

- We have created a methodology which performs a broad and complete analysis of gene expression data from microarray experiments. This methodology has successfully extracted all reliable targets from microarray experiments providing statistically meaningful results by means of combining the advantages of several microarray analysis methods. The gene information has been extracted based on the differential profiles that genes exhibit over time, treatment, patient, or other experimental conditions. Classical microarray analysis methods are not capable to extract all the information present in microarray data sets (as seen in Chapter 2), therefore we propose a methodology to overcome this problem. Decision making association rules have been created based on this information. These rules present some noteworthy characteristics, such as the capacity to combine them under certain conditions without lost of optimality in the non-dominance relation. We also provide a set of summary rules at different levels of granularity.

- We have functionally annotated the gene expression information, grouped in differential profiles, obtained from microarray experiments, with data from biological databases. These annotations have been achieved through mining of the databases and fusing the knowledge retrieved. The mining has been performed with already existing algorithms as well as new algorithms developed by us as part of this PhD. work. The databases used throughout this work have been human gene and diseases, biological pathways, gene product and DNA sequence. Information obtained from this databases has allowed us to, on the one hand, asses the cohesiveness of the

differential profiles obtained from our experiments, and on the other hand, perform a deeper research of the experiment being studied, providing us with cohesive groups of genes grouped not only by their expression profiles but also by the annotations related to their function and the biological processes where they are involved.

- Comparison of gene network creation methods and fusion of the information given by these genetic networks with the already known problem related information. We have compared the behavior of static vs. dynamic modeling. On the one hand, static modeling searches for relations between the expression levels of genes throughout time. The relation found by static methods might not only be similar behavior throughout time (direct correlation), but an inverse correlation (two genes having exactly opposite profiles over time), a proximity on the expression values (distance measures such as Euclidean Distance or City block distance). On the other hand, dynamic modeling retrieves temporal dependencies among genes, i.e., it detects dependencies of a gene at time t_{+1} related to some other(s) gene at time t.We have made a study of the performance of these two models over a real problem, the immuno-inflammatory response problem. The gene networks, created based on the differential profiles obtained from the problem being studied, have provided us with information about the regulation process underlying the genes retrieved as significant from the experiment.

II Future Works

From a biological point of view, some of the works to perform are:

- Fusing the already known information with clinical data to study the cohesiveness of the differential profiles. This information is related to measures taken from the patient such as fever, headaches, cough and other related symptoms throughout the resolution of the experiment carried out. This information is now available and most likely will throw some light into the inflammation and host response to injury problem.

- Mining into SNPs databases. SNP (single nucleotide polymorphism), are DNA sequence variations occurring when a single nucleotide (A, T, C, or G) in the genome (or other shared sequence) differs between members of a species (or between paired chromosomes in an individual). For example, two sequenced DNA fragments from different individuals, AAGCCTA to AAGCTTA, contain a difference in a single nucleotide. In this case we say

that there are two alleles : C and T. Almost all common SNPs have only two alleles.SNPs are being studied in many human diseases, as schizophrenia, and it seems to be a good idea to perform a preliminary study on the inflammation and host response to injury. SNPs microarray technology has been developed to detect polymorphisms within a population

From a computational point of view, some of the works to perform are:

- Extend the methodology to include no only conventional statistical microarray analysis methods but other conceptually based methods, such as Bayesian statistical models (Smyth, 2004) or Kolmogorov-Smirnov-style (Subramanian et al., 2005) statistics.
- Scale the methodology to be capable to perform with large sets of data implementing some optimization techniques such as evolutionary computation.
- Application of this global methodology, integration of different available methods, to work with data different from gene expression, such as gene sequences, following the way opened by the ENCODE project (Guigo and Consortium., 2007).

III Publications Derived from this Thesis

To conclude we must mention that the main parts of this work have been published and/or submitted in different international and national journals and conferences, as well as a book chapter. They are:

International Journals

- Rocío C. Romero-Zaliz, Cristina Rubio-Escudero, J. Perren Cobb, Francisco Herrera, Oscar Cordon and Igor Zwir. A multi-objective evolutionary conceptual clustering methodology for gene annotation within structural databases: A case of study on the Gene Ontology database. *IEEE Transactions on Evolutionary Computation.* In press.
- Navajas-Pérez R, Rubio-Escudero C, Aznarte JL, Rejón MR, Garrido-Ramos MA. SatDNA Analyzer: a computing tool for satellite-DNA evolutionary analysis. *Bioinformatics.* 2007 ;23(6):767-8.

Book Chapters

- Learning robust dynamic networks in prokaryotes by Gene Expression Networks Iterative Explorer (GENIE). "Studies in Computational Intelligence", Springer-Verlag.

Lecture Notes

- Rafael Navajas-Pérez, Manuel Ruiz Rejón and Manuel Garrido-Ramos, José Luis Aznarte and Cristina Rubio-Escudero. (2007) satDNA Analyzer 1.2 as a valuable computing tool for evolutionary analysis of satellite-DNA families: revisiting Y-linked satellite-DNA sequences of Rumex (Polygonaceae).
- Cristina Rubio-Escudero, Oscar Harari, Oscar Cordón, Igor Zwir. (2007) Modeling genetic networks: from static to dynamic models.
- Oscar Harari, Cristina Rubio-Escudero and Igor Zwir. (2007) Targeting Differentially Co-regulated Genes by Multiobjective and Multimodal Optimization.
- Cristina Rubio-Escudero, Coral del Val, Oscar Cordón, Igor Zwir. (2006) Decision Making Association Rules for Recognition of Differential Gene Expression Profiles.
- Oscar Harari, Rocio Romero-Záliz, Cristina Rubio-Escudero, Igor Zwir. (2006) Fusion of domain knowledge for dynamic learning in transcriptional.
- Rocio Romero-Záliz, Cristina Rubio-Escudero, Oscar Cordón, Oscar Harari, Coral del Val, Igor Zwir. (2006) Mining Structural Databases: An Evolutionary Multi-Objective Conceptual Clustering Methodology.
- Cristina Rubio-Escudero, Rocio Romero-Záliz, Oscar Cordón, Oscar Harari, Coral del Val, Igor Zwir. (2006) Optimal Selection of Microarray Analysis Methods using a Conceptual Clustering Algorithm.

International Conference Contributions

- Navajas-Pérez R, Ruiz Rejón M, Garrido Ramos MA, Aznarte JL, Rubio-Escudero C. Sat DNA Analyzer, The First Computing Solution For Satellite-DNA Evolutionary Analysis. 5th Asia Pacific Bioinformatics Conference. Honk-Kong.

- Romero, R.C., Cordón, O., Rubio-Escudero, C., Zwir, I. A Multiobjective Evolutionary Fuzzy System for Promoter Discovery in E. coli. I International Workshop on Genetic Fuzzy System, Granada, Spain, March 2005.
- Rubio-Escudero, C., Cordón., O., Zwir, I. Identifying meaningful temporal gene expression patterns. Affymetrix Annual Group Meeting, Edinburgh, May 2004.

I Appendix A

In this Appendix we include some Tables and Figures which are referenced throughout the text of the main chapters.

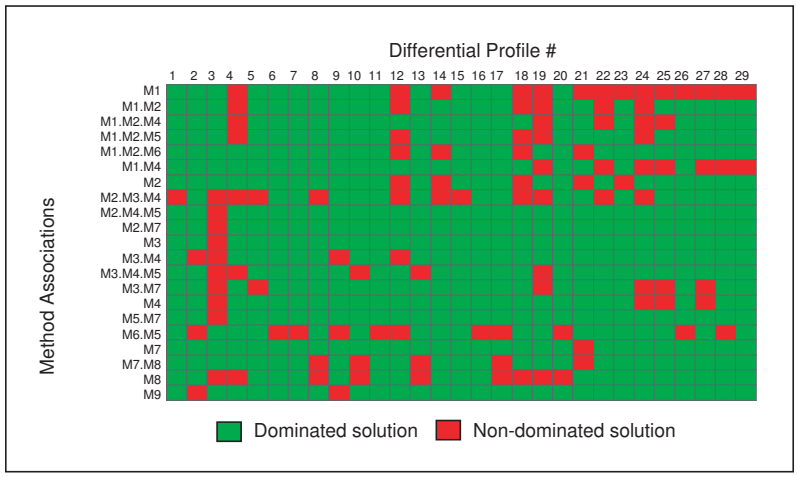


Figure 10: Lookup table for the ∩ operator in retrieving the differential profiles from the artificial data set. Red cells denote that the method association in the row is optimal to retrieve the differential profile in the column

Profile	Best Method association
#1	$M_2 \cup M_5 \cup M_7 \cup M_{10}$
#2	M_5
#3	M_2
#4	M_2
#5	$M_2 \cup M_3 \cup M_5 \cup M_9$
#6	$M_5 \cup M_6 \cup M_{10}$
#7	$M_5 \cup M_6 \cup M_7 \cup M_{10}$
#8	$M_2 \cup M_6 \cup M_9$
#9	M_5
#10	$M_2 \cup M_7$
#11	$M_2 \cup M_5 \cup M_6 \cup M_{10}$
#12	M_5
#13	$M_7 \cup M_{10}$
#14	$M_1 \cup M_4$
#15	$M_2 \cup M_{10}$
#16	$M_5 \cup M_6 \cup M_7$
#17	$M_5 \cup M_6$
#18	M_2
#19	$M_2 \cup M_{10}$
#20	$M_5 \cup M_6 \cup M_7$
#21	M_6
#22	M_2
#23	$M_1 \cup M_9$
#24	M_2
#25	M_1
#26	$M_1 \cup M_2$
#27	M_1
#28	$\mathrm{M}_1 \cup M_2$
#29	$M_1 \cup M_2$

Table 3: Method association for single profile association rules.

Profile	# Optimal Rules	Best Specificity	Best Sensitivity	Best Cost
#1	1	0.234562	0.654327	0.9
#2	3	0.178923	0.724536	0.9
#3	10	0.2156784	0.674589	0.9
#4	6	0.3678953	1	0.567894
#5	2	0.156234	0.823547	0.9
#6	1	0.267845	0.764891	0.9
#7	1	0.256912	0.786542	0.9
#8	3	0.134623	0.756432	0.9
#9	3	0.167892	0.875267	0.9
#10	3	0.208653	567460	0.9
#11	1	0.278901	0.746235	0.9
#12	7	0.37	0.457867	0.9
#13	3	0.167834	0.897645	0.9
#14	4	0.127891	0.765446	0.9
#15	1	0.231457	0.718954	0.9
#16	1	0.167891	0.632891	0.9
#17	3	0.190652	0.567312	0.9
#18	7	0.286349	0.897632	0.9
#19	9	0.356712	0.467321	0.9
#20	2	0.119834	0.754312	0.9
#21	5	0.143289	0.745678	0.9
#22	5	0.096234	0.654332	0.9
#23	2	0.256798	0.654234	0.9
#24	8	0.157345	0.578324	0.9
#25	5	0.207456	0.689319	0.9
#26	2	0.123457	0.785392	0.9
#27	4	0.186435	0.734591	0.9
#28	3	0.174672	0.653419	0.9
#29	2	0.134891	0.605681	0.9

Table 4: Number of Optimal Rules for each differential profile $\cap$ operator and summary of the specificity, sensitivity and cost values obtained. If a certain differential profile, say #15, is optimally retrieved by 3 different method associations M_2, $M_2 \cup M_{10}$ and $M_2 \cup M_4$, with specificity, sensitivity and cost values $(0.0158028, 0.925926, 0.9)$, $(0.0151339, 0.962963, 0.8)$ and $(0.0157112, 0.955674, 0.8)$. For this particular differential profile, the table shows the best specificity value obtained, (0.0158028) from M_2, the best sensitivity value obtained (0.962963) from $M_2 \cup M_4$, and the best cost value obtained, (0.9) from M_2. The best cost value will always be 0.9 representing a non-dominated solution by application of only one method.

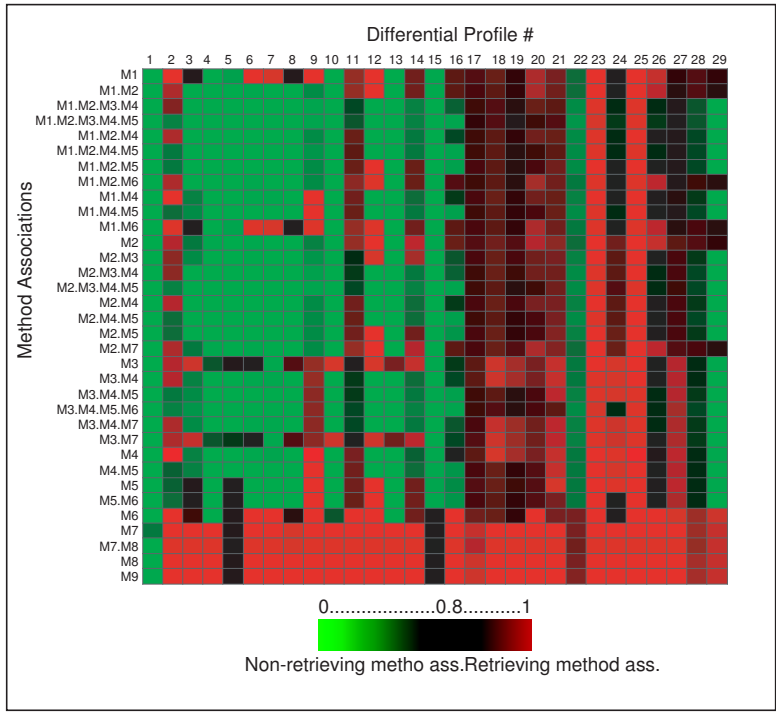


Figure 11: Behavior of the optimal associations of methods using the $\cap$ operator in retrieving the differential profiles from the inflammation and host response to injury problem..

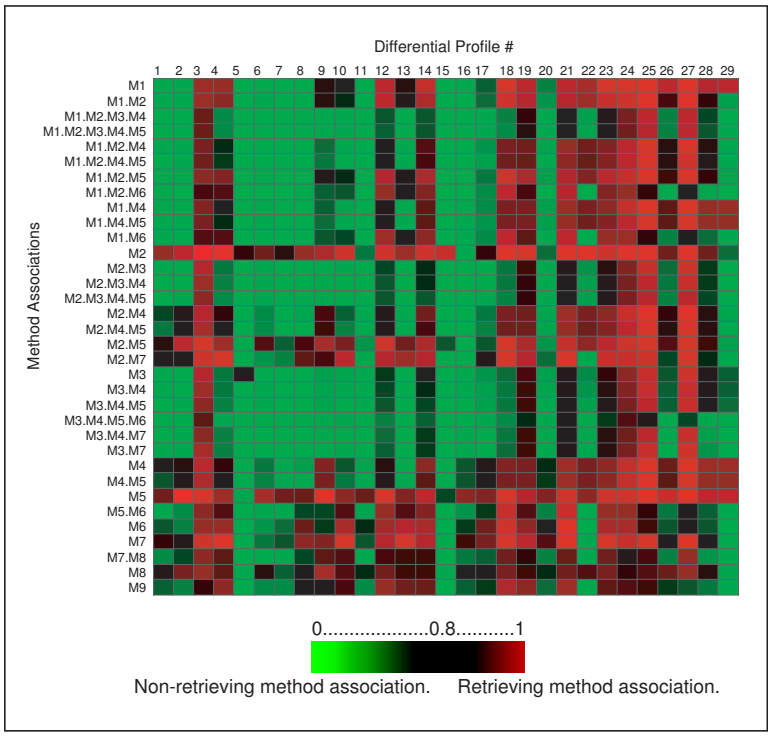


Figure 12: Behavior of the optimal associations of methods using the $\cap$ operator in retrieving the differential profiles from the artificial data set.

Bibliography

A., G., A., B., A., C., and L., M.: 2006, *Genet Res* **88(1)**, 27

Adriaans, P. and Zantinge, D.: 1996, *Data mining*, Addison-Wesley

Agrawal, R., Imielinski, T., and Swami, A.: 1993, in P. Buneman and S. Jajodia (eds.), *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp 207–216, Washington, D.C.

Agrawal, R. and Shafer, J.: 1996a, *IEEE Transactions on Knowledge and Data Engineering* **8**, 962

Agrawal, R. and Shafer, J. C.: 1996b, *Ieee Transactions on Knowledge and Data Engineering* **8(6)**, 962, Times Cited: 76 Article English Cited References Count: 12 Wc057

Aho, A., Hopcroft, J., John, E., and Ullman, J.: 1983, *Data Structures and Algorithms*, Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley

Alatas, B., Akin, E., and Karci, A.: 2008, *Applied Soft Computing Journal* **8(1)**, 646

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P.: 2003, *Biología Molecular de la Célula, Cuarta Edición*, Omega

Alizadeh, A. A., Eisen, M., Davis, R., Ma, C., and Losses, I.: 200, *Nature* **403(503)**, 511

Applegate, K. E. and Crewson, P. E.: 2002, *Radiology*

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, 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, *Nat Genet* **25(1)**, 25, 1061-4036 Journal Article

Attwood, T. and Parry-Smith, D.: 2002, *Introducción a la Bioinformática*, Prentice Hall

Back, T., Fogel, D., and Michalewicz, Z. (eds.): 1997, *Handbook of Evolutionary*

Computation, IOP Publishing Ltd., Bristol, UK

Barenco, M., Stark, J., Brewer, D., Tomescu, D., Callard, R., and Hubank, M.: 2006, *BioMed Central* **9(7)**, 251

Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M., and Yakhini, Z.: 2000, *Computer Biol.* **7(559)**, 583

Benson, D., Karsch-Mizrachi, I., Lipman, D., Ostell, J., and Wheeler, D.: 2007, *Nucleic Acids Res* **35**, D21

Bezdek, J.: 1998a, in E. Ruspini, P. Bonissone, and W. Pedrycz (eds.), *Handbook of Fuzzy Computation*, pp f6.1:1–f6.6:19, Institute of Physics Press

Bezdek, J. C.: 1998b, in W. Pedrycz, P. P. Bonissone, and E. H. Ruspini (eds.), *Handbook of Fuzzy Computation*, pp F6.1.1–F6.6.20, Institute of Physics, Bristol, editors in chief: Enrique H. Ruspini, Piero P. Bonissone and Witold Pedrycz ill.

Bone, R. C., Balk, R., Cerra, F., Dellinger, R., Fein, A., Knaus, W., RM, R. S., and W.J.Sibbald: 1992, *Chest* **101**, 1644

Britten, R. and Davidson, E.: 1969, *Science* **165**, 349

Brown, P. and Botstein, D.: 1999, *Nature Genet.* **21(Suppl.)**, 33

Calvano, S. E., Xiao, W., Richards, D. R., Felciano, R. M., Baker, H. V., Cho, R. J., Chen, R. O., Brownstein, B. H., Cobb, J. P., Tschoeke, S. K., Miller-Graziano, C., Moldawer, L. L., Mindrinos, M. N., Davis, R. W., Tompkins, R. G., Lowry, S. F., and Large Scale Collab Res Program, I. A.: 2005, *Nature*

Chankong, V. and Haimes, Y. Y.: 1983, *Multiobjective decision making theory and methodology*, North-Holland

Charlesworth, B., Sniegowski, P., and Stephan, W.: 1994, *Nature* **671**, 125

Cheeseman, P. and Oldford, R. W.: 1994, *Selecting models from data : artificial intelligence and statistics IV*, Springer-Verlag, New York, p. Cheeseman, R.W. Oldford (eds.).ill.

Cheeseman, P. and Oldfors, R. W.: 1994, *Selecting models from data*, Springer-Verlag

Chickering, D. M.: 2003, *Journal of Machine Learning Research* **3**, 507

Cho, R., Campbell, M., Winzeler, E., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T., Gabrielian, A., Landsman, D., Lockhart, D., and Davis, R.

Coello-Coello, C., Veldhuizen, D. V., and Lamont, G.: 2002, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Genetic Algorithms and Evolutionary Computation, Kluwer

Cohen, J.: 1992, *Current Directions in Psychological Science* **1(3)**, 98

Collins, F., Morgan, M., and Patrinos, A.: 2003, *Science* **300(5617)**, 286

Consortium, T. G. O.: 2000, *Nature Genet.* **25**, 25

Cook, D., Holder, L., Su, S., Maglothin, R., and Jonyer, I.: 2001a, *IEEE Engineering in Medicine and Biology, special issue on Advances in Genomics* **4(20)**, 67

Cook, D. J., Holder, L. B., Su, S., Maglothin, R., and Jonyer, I.: 2001b, *IEEE Eng Med Biol Mag* **20(4)**, 67, 0739-5175 Journal Article

Cooper, G. F. and Herskovits, E.: 1992, *Machine Learning* **9(4)**, 309, Times Cited: 226 Article English Cited References Count: 71 Jq511

Cordón, O., del Jesus, M., and Herrera, F.: 1999, *International Journal of Approximate Reasoning* **20**, 21

Coussens, L. M. and Werb, Z.: 2002, *Nature* **420**, 860

Davies, D. L. and Bouldin, W.: 1979, *IEEE PAMI* **1**, 224

Deb, K.: 2001, *Multi-objective optimization using evolutionary algorithms*, Wiley-Interscience series in systems and optimization, John Wiley and Sons, Chichester ; New York, 1st edition, Kalyanmoy Deb. ill. ; 26 cm.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T.: 2000, Proceedings of the Parallel Problem Solving from Nature VI Conference

Deb, K. and Reddy, A. R.: 2003, *BioSystems* **72(1-2)**, 111

del Val, C., Kuryshev, V., Glatting, K., Ernst, P., Hotz-Wagenblatt, A., Poustka, A., Suhai, S., and Wiemann, S.: 2006, *BMC Bioinformatics* **7**, 473

Delima, P. and Yen, G.: 2005, *ISA Transactions* **44(2)**, 315

Der, G. and Everitt, B.: 1996, *A handbook of statistical analyses using SAS*, CHAPMAN-HALL

Der, G. and Everitt, B.: 2001, *Handbook of Statistical Analyses using SAS*, Chapman and Hall/CRC

D'haeseleer, P., Liang, S., and Somogyi, R.: 2000, *Bioinformatics* **16(8)**, 707

D'Onia, D., Tam, L., Cobb, J., and Zwir, I.: 2003, in *Proceedings of the 3rd International Conference on Systems Biology (ICSB)*, pp 284–285, St. Louis, USA

Dover, G. and Flavell, A.: 1982, *Genome Evolution*, Academic, London

Duda, R. and Hart, P.: 1973, *Pattern Classification and Scene Analysis.*

Duda, R., Hart, P., and Stork, D.: 2000, *Pattern Classification (2nd Edition)*, Wiley-Interscience

Durbin, R., Eddy, S., Krogh, A., and Mitchison, G.: 1998, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press

Efron, B.: 2005, *Local false discovery rates*

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D.: 1998, *Proc Natl Acad Sci U S A* **95(25)**, 14863, 0027-8424Journal Article

Evangelos, T.: 2000, *Multi-criteria decision making methods: a comparative study*, Kluwer Academic Publishers, Dordrecht

Freitas, A.: 2002, *Data mining and knowledge discovery with evolutionary algorithms*, Springer, Heidelberg, Germany.

Fuhrman, S., Cunningham, M., Wen, X., Zweiger, G., Seilhamer, J., and Somogyi, R.: 2000, *Biosystems* **5**, 5

Galitski, T., Saldanha, A., Styles, C., Lander, E., and Fink, G.: 1999, *Science* **285**, 251

Gao, X. and Song, P.: 2005, *BMC Bioinformatics* **21**, 6

Gregory, W.: 2005, *Bioinformatics* **6**, 380

Grothaus, G., Mufti, A., and Murali, T.: 2006, *Algorithms Molecular Biology* 1:15

Guet, C. C., Elowitz, M. B., Hsing, W., and Leibler, S.: 2002, *Science* **296(5572)**, 1466, 1095-9203 Journal Article

Guigo, R. and Consortium., T. E. P.: 2007, *Nature* **447**, doi:10.1038/nature05874

Hakamada, K., Okamoto, M., and Hanai, T.: 2006, *Bioinformatics* **22(7)**, 843

Hall, L., Ozyurt, I., and Bezdek, J.: 1999, *IEEE Transactions on Evolutionary Computation* **3(2)**, 103

Halmos, P.: 1960, *Naive set theory*, Princeton, NJ: D. Van Nostrand Company

Hamosh, A., Scott, A., Amberger, J., Bocchini, C., and McKusick, V.: 2005, *Nucleic Acids. Res.* **33**, D514

Hand, D., Mannila, H., and Smyth, P.: 2001, *Principles of Data Mining*, MIT Press, HAN d3 01:1 1.Ex

Handl, J. and Knowles, J.: 2007, *IEEE Transactions on Evolutionary Computation* **11(1)**, 56

Hasty, J., McMillen, D., Isaacs, F., and Collins, J. J.: 2001, *Nat Rev Genet* **2(4)**, 268, 1471-0056 Journal Article Review Review, Tutorial

Herrera, F., Cano, J., and Lozano, M.: 2007, *Data and Knowledge Engineering* **60**, 90

Herrera, F., Herrera-Viedma, E., and Verdegay, J.: 1997, in Y. Kacprzyk and J. (eds.), *The Ordered Weighted Averaging Operators: Theory, Methodology and Applications*, p. 207, Kluwer Academic

Herrera, F., Lozano, M., and Verdegay, J.: 1994, *Fuzzy Systems and A.I.-Reports and Letters* **3**, 39

Inza, I., Larrañaga, P., Blanco, R., and Cerrolaza, A.: 2004, *Artif Intell Med.* **31(2)**, 91

Jonyer, I., Cook, D. J., and Holder, L. B.: 2001, *Journal of Machine Learning Research* **2**, 19

Kaern, M.: 2003, *Regulatory dynamics in engineered gene networks*

Kanehisa, M., Goto, S., Kawashima, S., and Nakaya, A.: 2002, *Nucleic Acids Res* **30(1)**, 42

Kanz, C., Aldebert, P., Althorpe, N., Baker, W., Baldwin, A., Bates, K., Browne, P., van den Broek, A., Castro, M., Cochrane, G., Duggan, K., Eberhardt, R., Faruque, N., Gamble, J., Garcá-Diez, F., Harte, N., Kulikova, T., Lin, Q., Lombard, V., Lopez, R., Mancuso, R., McHale, M., Nardone, F., Silventoinen, V., Sobhany, S., Stoehr, P., Tuli, M., Tzouvara, K., Vaughan, R., Wu, D., Zhu, W., and Apweiler, R.: 2005, *Nucleics Acids Research* **33(suppl_1)**, D29

Kishino, H. and Waddell, P.: 2000, *Genome Informatics* **11**, 83

Larranaga, P., Poza, M., Yurramendi, Y., Murga, R. H., and Kuijpers, C. M. H.: 1996, *IEEE Journal on Pattern Analysis and Machine Intelligence* **18**, 912

Li, C. and Wong, W.: 2003, *The analysis of gene expression data: methods and software*, Springer

Li, C. and Wong, W. H.: 2001a, *Proc Natl Acad Sci U S A* **98(1)**, 31, 21065211 0027-8424 Journal Article

Li, C. and Wong, W. H.: 2001b, *Genome Biology* **2(8)**, research0032.1

Liu, H. and Motoda, H.: 1988, *Feature Selection for Knowledge Discovery and Data Mining*, Kluwer Academic Publishers, Dordrecht

Lockhart, D. and Winzeler, E.: 2000, *Nature* **405(6788)**, 827

Martinez-Antonio, A. and Collado-Vides, J.: 2003, *Curr Opin Microbiol* **6**, 482

Marton, M., Bennett, J. D. H., Iyer, V., Meyer, M., Roberts, C., Stoughton, R., Burchard, J., Slade, D., Dai, H., Bassett, D., Hartwell, L., Brown, P., and Friend, S.: 1998, *Nat Med* **4(11)**, 1235

McAdams, H. H. and Shapiro, L.: 2003, *Science* **301(5641)**, 1874, 1095-9203 Journal Article Review Review, Tutorial

McCue, L., Thompson, W., Carmack, C., Ryan, M. P., Liu, J. S., Derbyshire, V., and Lawrence, C. E.: 2001, *Nucleic Acids Res* **29**, 774

Mendes, P.: 2003, *Comput. Appl. Biosci.* **9**, 563

Mendes, P., Wei, S., and Keying, Y.: 2003, *Bioinformatics* **19(2)**, 122

Mitchell, T.: 1997, *Machine Learning*, McGraw-Hill, New York

Mitra, S. and Banka, H.: 2006, *Pattern Recognition* **39(12)**, 2464

Morita, M., Sabourin, R., Bortolozzi, F., and Suen, C. Y.: 2003, *icdar* **02**, 666

Nadon, R. and Shoemaker, J.: 2002, *Trends Genet* **18(5)**, 265

Navajas-Pérez, R.: 2005, *J. Mol. Evol* **60**, 391

Navajas-Pérez, R., Rubio-Escudero, C., Aznarte, J., Rejón, M., and Garrido-Ramos, M.: 2007, *Bioinformatics* **23(6)**, 767

Nevers, P. and Saedler, H.: 1977, *Nature* **268**, 109

Nikitin, A., Egorov, S., Daraselia, N., and Mazo, I.: 2003, *Bioinformatics* **19(16)**, 2155

Oliveros, J., Blaschke, C., Herrero, J., Dopazo, J., and Valencia, A.: 2000, *Genome Inform.* 10(106)

Orgel, L. and Crick, F.: 1980, *Nature* **284**, 604

Pan, S. and Cheng, K.: 2007, *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* **37(5)**, 827

Pan, W., Lin, J., and Le, C.: 2001, *Funct. Integr. Genomics* **3(3)**, 117

Pargas, R., M.J., H., and Peck, R.: 1999, *Test-data generation using genetic algorithms*

Park, T., Yi, S., Lee, S., Lee, S., Yoo, D., Ahn, J., and Lee, Y.: 2003, *Bioinformatics* **19(6)**, 694

Prelic, A., Bleuler, S., Zimmermann, P., Wille, A., Buhlmann, P., Gruissem, W., Hennig, L., Thiele, L., and Zitzler, E.: 2006, *Bioinformatics* **22(9)**, 1122

R., O.-R. and K., S.: 2006, *REVSTAT* **4**, 53

Rice, J. and Stolovitzky, G.: 2004, *Biosilico* **2(2)**, 70

Richardson, R., Rhyne, C., Fong, Y., D.G.Hesse, Tracey, K., Marano, M., Lowry, S., Antonacci, A., and S.E.Calvano: 1989, *Ann.Surg* **210(2)**, 239

Rissanen, J.: 1989, *Stochastic Complexity in Statistical Inquiry Theory*, World Scientific Publishing Co., Inc.

Romero-Zaliz, R., Zwir, I., and Ruspini, E.: 2004, *Applications of Multi-Objective Evolutionary Algorithms*, Chapt. Generalized Analysis of Promoters (GAP): A method for DNA sequence description, pp 427–450, World Scientific

Romero-Zaliz, R. C., Rubio-Escudero, C., Cobb, J. P., Herrera, F., O., C., and Zwir, I.: 2007, *IEEE Transactions on Evolutionary Computation* In press

Rosania, G., Chang, Y., Perez, O., Sutherlin, D., Dong, H., Lockhart, D., and Schultz, P.: 2000, *Nat Biotechnol* **18(3)**, 261

Rubio-Escudero, C., Romero-Záliz, R., Cordón, O., Harari, O., del Val, C., and Zwir, I.: 2005, *Optimal Selection of Microarray Analysis Methods using a Conceptual Clustering Algorithm*

Ruspini, E. and Zwir, I.: 1999, in *Proceedings of 16th IEEE Instrumentation and Measurement Technology Conference*, Vol. 2, pp 1086 – 1091, Venice, Italy

Ruspini, E. and Zwir, I.: 2001, in S. Pal and A. Pal (eds.), *Pattern Recognition: From Classical to Modern Approaches*, pp 453–474, World Scientific Company, Singapore

Ruspini, E. and Zwir, I.: 2002, in S. K. Pal and A. Pal (eds.), *Pattern recognition : from classical to modern approaches*, pp 454–474, World Scientific, New Jersey., editors: Sankar K. Pal, Amita Pal. ill. (some col.) ; 22 cm.

Salvador, S. and Chan, P.: 2004, *Determining the Number of Clusters/Segments in Hierarchical Clustering/Segmentation Algorithms*

Schäfer, J. and Strimmer, K.: 2005, *Bioinformatics* **21**, 754

Schuler, G., Boguski, M., and Stewart, E.: 1996a, *Science* **274(5287)**, 540

Schuler, G., Epstein, J., Ohkawa, H., and Kans, J.: 1996b, *Methods Enzymol* **266**, 141

Simon, I., Siegfried, Z., Ernst, J., and Bar, Z.: 2005, *Nat. Biotechnol.* **23(12)**, 1503

Siripurapu, V., Meth, J., Kobayashi, N., and Hamaguchi, M.: 2005, *Journal of Molecular Biology* **346(1)**, 83

Smyth, G.: 2004, *Stat Appl Genet Mol Biol* **3**, Article3

Spellman, P. T., Sherlock, G., Zhang, M., Iyer, V., Anders, K., Eisen, M., Brown, P., Botstein, D., and Futcher, B.: 1998, *Mol. Biol. Cell* **9(12)**, 3273

Strachan, T.: 1985, *EMBO J* **4**, 1701

Stryer, L., Berg, J., and Tymoczko, J.: 2003, *Biochemistry, Fifth Edition*, W. H. Freeman and Company

Subramanian, A., Tamayo, P., Mootha, V., Mukherjee, S., Ebert, B., Gillette, M., Paulovich, A., Pomeroy, S., Golub, T., Lander, E., and Mesirov, J.: 2005, *Proc Natl Acad Sci USA* **102(43)**, 15545

Systems, I., *Ingenuity Pathways Analysis*

Tanay, A., Sharan, R., Kupiec, M., and Shamir, R.: 2004, *Genetics* **101(9)**, 2981

Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J., and Church, G. M.: 1999, *Nat Genet* **22(3)**, 281, 99318101 1061-4036 Journal Article

Tusher, V.G., T.-R. and Chu, G.: 2001, *Proc. Natl. Acad. Sci. USA.* **98**, 5116

van Someren, E., Wessels, L., Backer, E., and Reinders, M.: 2002, *Pharmacogenomics* **3(4)**, 507

Vaquerizas, J., Conde, L., Yankilevich, P., Cabezón, A., Minguez, P., Díaz-Uriarte, R., Al-Shahrour, F., Herrero, J., and Dopazo, J.: 2005, *Nucleic Acids Res* **33**, W616

Velarde, C.: 2006, *Ph.D. thesis*, University of Buenos Aires

Wahde, M., Hertz, J., and Andersson, M.: 2001, *Reverse Engineering of Sparsely Connected Genetic Regulatory Networks*

Wallace, M., Andersen, L., Saulino, A., Gregory, P., Glover, T., and Collins, F.: 1991, *Nature* **353**, 864

Witten, I. and Frank, E.: 2000, *Data mining: practical machine learning tools and techniques with Java implementations*, Morgan Kaufmann Publishers, San Francisco, CA, USA

Yeung, K. and Ruzzo, W.: 2001, *Bioinformatics* **17**, 763

Zwir, I., Huang, H., and Groisman, E.: 2005a, *Bioinformatics* **21(22)**, 4073

Zwir, I., Huang, H., and Groisman, E. A.: 2005b, *Bioinformatics* **21(22)**, 4073

Zwir, I., Romero-Zaliz, R., and Ruspini, E.: 2002, in F. Valafar (ed.), *Techniques in bioinformatics and medical informatics*, Vol. 980, pp 65–82

Zwir, I., Shin, D., Kato, A., Nishino, K., Latifi, T., Solomon, F., Hare, J. M., Huang, H., and Groisman, E. A.: 2005c, *Proc Natl Acad Sci U S A* **102(8)**, 2862, 0027-8424 Journal article

Zwir, I., Shin, D., Kato, D. A., Nishino, K., Kunihiko, T., Solomon, F., Hare, J., Huang, H., and Groisman, E.: 2005d, *PNAS* **102(8)**, 2862