Systems Biology: Knowledge Discovery and Reverse Engineering

- PhD Thesis -

Alessia Visconti

June 11, 2012

Supervisors: Marco Botta, Roberto Esposito

Doctoral School on Science and High Technology
Department of Computer Science
University of Torino

Contents

Abstract

Systems biology is the discipline aiming at understanding complex biological systems by integrating experimental and computational approaches. In the recent years, several developments in this field shed light on relevant aspects of the functioning of living being. Nonetheless, a number of important problems are open and worth investigation. Some of these problems have been studied in the work presented in this thesis. Among them we mention: the analysis of huge biological data sets, their interpretation in order to extract knowledge about the functioning of biological systems, and the problem of modeling biological systems themselves.

This dissertation is divided into four parts which cover two areas of investigation. The first two parts introduce the context we are concerned with, and propose several methods for discovering regulative information from different knowledge sources. Then, we focus on two approaches for the reverse engineering problem. The two big areas of investigation mentioned, i.e., knowledge discovery and reverse engineering, are not the only tasks systems biology is concerned with. In recent years, this field of research became a major actor in several other areas of research. Before concluding this dissertation, we apply a systems biology approach to one important problem: that of finding pharmacogenes.

Papers included in the thesis

The following papers are presented in this thesis.

- **Visconti A.**, Esposito R., and Cordero F., *Restructuring the Gene Ontology to Emphasize Regulative Pathways and to Improve Gene Similarity Queries*, Int. J. Computational Biology and Drug Design, Vol. 4, No. 3, 2011, Inderscience Publishers, pp. 220-238.

- **Visconti A.**, Esposito R., and Cordero F., *Tackling the DREAM Challenge for Gene Regulatory Networks Reverse Engineering*, In Proceedings of AI*IA 2011: Artificial Intelligence Around Man and Beyond, XIIth International Conference of the Italian Association for Artificial Intelligence - Palermo, September 15-17, 2011, volume 6934 of Lecture Notes in Artificial Intelligence (LNAI), Springer, pp. 373-383.

- **Visconti A.**, Calogero R. A., and Cordero F., *An integrated approach for pharmacogenes discovering*, Submitted to the 11th European Conference on Computational Biology (ECCB12).

Papers not included in the thesis

The following papers constitute a basis for the works described in this thesis.

- **Visconti A.**, Cordero F., Botta M., and Calogero R.A., *Gene Ontology rewritten for computing gene functional similarity*, In Proceedings of the Fourth International Conferences on Complex, Intelligent and Software Intensive Systems, February 15-18 2010, IEEE Computer Society Press, pp. 694-699.

- **Visconti A.**, Cordero F., Ienco D., and Pensa R.G., *Coclustering under Gene Ontology Derived Constraints for Pathway Identification*, Biological Knowledge Discovery Handbook: Preprocessing, Mining and Postprocessing of Biological Data, Mourad Elloumi and Albert Y. Zomaya (Eds.), Wiley, USA [To appear].

- Cordero F., Pensa R.G., **Visconti A.**, Ienco .D, Botta M., *Ontology-driven Co-clustering of Gene Expression Data*, In Proceedings of AI*IA 2009: Emergent Perspectives in Artificial Intelligence, XI International Conferences of the Italian Association for Artificial Intelligence - Reggio Emilia, December 9-12, 2009, volume 5883 of Lecture Notes in Artificial Intelligence (LNAI), Springer, pp. 426-435.

Work in progress

Chapter 4 and Chapter 6 are the base for two manuscripts that are in preparation.

Acknowledgements

A big thank to Roberto Esposito and Francesca Cordero. As my day-to-day supervisors, they always had an open ear and provided support whenever needed. I am further grateful for their commenting and proof-reading of the thesis. A special thank to Christopher Workman for the collaboration at the *Center for Biological Sequences Analysis* as well as for scientific input and enthusiasms. A thank also to Marco Botta, who involved me in this PhD program, for his proof-reading of the thesis.

Thanks to all current and past members of the *Computational Biology Group* for providing interesting discussions, both at lunch or in the office, and for some great nights out.

To be a PhD student has been an invaluable experience. I am very thankful to everyone who provided such a stimulating environment. I enjoyed the time with my colleagues both in Torino and in Lyngby.

An extra thank you to friends I found and to friends I had.

Part I

Prologue

Chapter 1

Introduction

In the past decade, a number of new high-throughput technologies designed to study life at the genome-wide scale arose. Nowadays, it is possible to routinely sequence the entire genome of an organism, measure the abundance of genes and of their products, map epigenetic modifications and transcriptional regulation, and comprehensively measure metabolites in virtually any biological specimen.

On the one hand, this sheer volume of data offer the unique opportunity to study how individual parts of a biological system work together to produce emerging phenotypes. Indeed, even though the *reductionist* method has been effective in explaining the basis of numerous living processes, it is actually inadequate to completely describe biological systems. In fact, at the molecular level, cells are a combinations of tightly interconnections among DNA, RNA, proteins, and metabolites. The knowledge of the wiring and interplay of these individual parts is a mandatory step to understand complex properties and functionalities of a biological system. In fact, cells are complex systems whose behaviour cannot be reduced to a sum of individual pieces [8, 106, 205].

On the other hand, the gigantic size of the available data makes it necessary to develop new computational methods to assist in the analysis processes and new languages to describe the extracted knowledge.

A new research area called *systems biology* arose in recent years to cope with these problems. It adopts an integrative approach to study the function of biological systems and to emphasize the importance of a holistic view [62, 87, 104, 105, 106]. The ambitious goal of systems biology is the understanding of an entire biological system by modeling, predicting, and controlling the behaviour of all its components.

Modeling and investigating complex biological processes are the cornerstone of systems biology. Models provide significant insights into the underlying biology so unveiling important application opportunities. They allow researchers to give precise definitions and to dissect the role of components of a given system. Also, they can be easily used for system simulations. The model investigation can have high impact in several fields, such as drug discovery and personalized medicine, and it allows one the study of the molecular basis of diseases. In this context, systems approaches are grounded in the idea that disease-perturbed cellular processes differ from their normal counterparts [9, 79]. In fact, cellular processes are defined by complex genetic programs that allow genes to be expressed in a tightly regulated manner, and errors in the regulatory machinery are a common trait of many diseases. For instance, a misregulation of genes controlling cell proliferation leads to an uncontrolled cell division (a characterizing property of tumours [216]) as well as to a metastatic tumour progression [108]. Furthermore, changes in expressions of a number of genes characterize metabolic diseases, such as diabetes [223]. However, little is known about mechanisms that control gene expressions and about their variations. The modeling of gene regulatory machinery remains a cardinal task on the biological research agenda.

Usually, gene interactions are represented by networks in which genes are vertices, and edges represent regulative connections. These networks are known as *gene regulatory networks* or *transcriptional regulatory networks.*

Many attempts to model gene regulatory networks have been presented to date, but we are still far from a complete understanding of cellular mechanisms. Models are readily available only for simple unicellular organism (e.g. *Saccharomyces cerevisiae* [69] and *Escherichia coli* [132]), while for higher eukaryiotic organisms only fragments of these networks have been modelled so far [19, 25]. In fact, the *reverse engineering* problem, i.e., the inference of gene regulatory networks from data, is not by any means a trivial process. Indeed, in each cell thousands of genes act at different time points, interacting with multiple partners either directly or indirectly, leading to dynamic and non-linear relationships [224].

An important issue in gene regulatory networks is that the number of genes (and of relationships) to model is several orders of magnitude larger than the number of independent measurements that can be made using today technologies [37]. To solve this issue, the number of genes to be modeled is often reduced by removing uninteresting profiles or by grouping together genes that are co-expressed under some experimental condition. Indeed, several authors showed that to decompose a gene regulatory network into a small set of recurring regulatory modules is a promising strategy to address this challenge [18, 168, 224]. Another approach is to try to augment the available information through data integration [36, 77, 119]. Indeed, several works showed that it is beneficial to integrate system-wide genomic, transcriptomic, and proteomic measurements as well as prior biological knowledge into a single modeling process. For instance, the exploitation of biological knowledge about the network structure [63, 91, 115] as well as the integration of protein-DNA interactions data and gene expression data [59, 74, 179, 200] increases the accuracy of techniques dealing with networks.

Unfortunately, the integration of data from different sources is itself a challenging problem so that presently only a few approaches try to cope with more than two different data sources [119, 139]. In fact, when one deals with the problem of integrating different data sets, there are many issues to be aware of. For instance, the same information could be stored in different databases using different identifiers, making it difficult to retrieve/merge the desired information. Also, some important pieces of information could be buried inside a huge amount of irrelevant data, and a data mining process could then be necessary to unveil it. Moreover, current high-throughput techniques generate data that involves substantial amount of noise, or some information could be missing because of the intrinsic difficulties in its measurement. As a consequence, appropriate analysis algorithms are required to make the plethora of available data more understandable and useful. Despite these difficulties the promising results obtained by integrative approaches motivated increasing efforts in this fledgling area of investigation.

In this thesis we focus on a number of issues in systems biology that are still wide open and worth investigation.

First, analysis tools need to be ameliorated in order to properly deal with the biological data. A primary step in this field consists in extracting and organizing biological knowledge provided by both raw data and literature. To this purpose, we propose a reorganization of the Gene Ontology aiming at enhancing the recovery of information about gene shared functionalities. Then, we develop a knowledge-driven biclustering approach aiming at discovering transcriptional modules.

Second, new approaches for the reverse engineering problem itself must be developed. In this thesis, we propose two techniques: a framework merging different experimental evidences about gene interactions, and a method aiming at deciphering temporal influences among genes and proteins.

Finally, we exploit a systems biology approach in a different and challenging area of research, that is pharmacogenomics.

Structure and original contributions

The thesis is divided into four parts.

- **Part I** is an introduction to the fields covered in the thesis. Specifically, Chapter 2 gives some insights on the fields of molecular biology and on gene regulatory networks. It is important to note that the main purpose is to make the thesis as self-contained as possible. Thus, readers familiar with these topics can skip this chapter without compromizing the understanding of later material.

- **Part II** describes two works dealing with the problem of knowledge discovery. Chapter 3 presents a reorganization of the Gene Ontology that makes explicit information about gene cooperation, functions and localizations that were implicitly coded in the original structure. Chapter 4 describes a knowledge-driven biclustering approach aiming at discovering transcriptional modules.

- **Part III** presents two methodologies for the reverse engineering of gene regulatory networks. Specifically, Chapter 5 describes a framework based on a Naive Bayes approach that merges multiple pieces of information derived from microarray experiments, and Chapter 6 presents a new integrative reverse engineering approach based on a well-known econometrics measure, namely the *Granger Causality.*

- **Part IV** finalizes this work. Chapter 7 describes a systems biology approach for pharmacogenes discovering. It shows the importance of systems biology research in valuable fields, such as human-health. Chapter 8 summarizes the work presented on this thesis.

Most of the ideas presented in this thesis come from my academic career at the University of Torino. The work described in Chapter 6 has been carried out during my visiting period at the Technical University of Denmark, Center for Biological Sequence Analysis, under the supervision of prof. Christopher Workman.

Chapter 2

Background

This thesis is concerned with molecular and systems biology. Molecular biology is the study of biological activities at the molecular level. It involves the understanding of the genetic and biochemical organisation of living matter, and of the interactions between various systems of a cell. Systems biology is the discipline aiming at understanding complex biological systems by integrating experimental and computational approaches.

This chapter presents a short introduction to molecular biology and to the biological data used in this dissertation. It provides a glossary of terms and concepts that are used in the next chapters. Moreover, this chapter briefly introduces gene regulatory networks.

2.1 Molecular biology

The biochemistry of cells is based on bio-polymers. A bio-polymer is a *macromolecule* produced by living organisms and composed by repeating structural units. Bio-polymers are the building blocks of a cell, and they are the distinctive traits of living matters. In this dissertation, three bio-polymers play a central role: *DeoxyriboNucleic Acid* (DNA), *RiboNucleic Acid* (RNA) and *proteins*.

DNA is a nucleic acid composed by two long chains (*strands*) of units called *nucleotides*. DNA codifies the genetic information into genes: "locatable regions of genomic sequence, corresponding to units of inheritance, which are associated with regulatory regions, transcribed regions, and/or other functional sequence regions" [152]. Notably, not all the DNA encodes information responsible for protein synthesis: remaining DNA sequences have a regulative or a structural role [146].

RNA is a nucleic acid composed by a single strand of nucleotides. Several types of RNA are present in the cell. The *messenger RNA* (mRNA) carries the genetic information responsible for the synthesis of proteins. *Non-coding RNA* (ncRNA) is responsible for collateral activities. Examples of ncRNA are *transfer RNA* (tRNA), which pursues the passing of information from genes to proteins, *ribosomal RNA* (rRNA), which decodes the mRNA, and *micro RNA* (miRNA) which is responsible for gene regulation [30].

Proteins are composed by chains of units called amino acids, and are responsible for all cellular functions. For instance, as *enzymes*, proteins catalyze biochemical reactions; as *transcription factors*, proteins regulate protein synthesis itself; as *antibodies*, proteins are used by the immune system to identify and neutralize bacteria and viruses. In the cytoskeleton, proteins perform a structural function by maintaining the cell shape. Most of the cellular activities involve a large number of proteins interacting with one another by taking part in *protein complexes.* Proteins interactions are both direct (*physical*) and indirect (*functional*) and they can be stable, when a protein complex concerns multi-subunit complexes (e.g. hemoglobin), or transient, when interactions are promoted only if a set of conditions is present. Transient interactions are responsible for the majority of cellular processes (e.g. transport, protein modification, signaling). Proteins are translated upon request, and their concentration changes promptly according to cell requirements. Some proteins are *degraded* rapidly, while many others are stable under given conditions and become unstable upon a state change.

The *central dogma of molecular biology*, illustrated in Figure 2.1, explains how the genetic information goes from DNA to proteins through mRNA [43]. The process by which the genetic information stored in DNA is transferred to mRNA is called *transcription.* The step leading from mRNA to protein is called *translation.*

In transcription, a DNA sequence encoding genetic information is copied into mRNA sequences. From each DNA sequence thousands of copies of mRNA can be produced, and the total amount of mRNAs transcripted is called *gene expression level.* In translation, the mRNA is decoded by rRNA and tRNA and synthetized into a specific protein.

It is worth mentioning that it is a common assumption to take the gene expression level as proportional to the amount of proteins translated. Indeed, several studies show that there exists a strong correlation between expression levels and protein abundance [60, 68, 127]. However, it is well known that there exist situations where the opposite is true [71, 174]. In these cases, it is said that a *post-transcriptional* activity occurred.

Gene expression depends on transcription factor activities. A transcription factor is a protein that binds specific DNA sequences, called *binding sites*, in order to regulate transcription. A single transcription factor can target several binding sites in a genome. As a consequence, changes in its activity may affect hundreds of genes. If a transcription factor hinders mRNA production it is called a *repressor*, otherwise it is called an *enhancer.* In

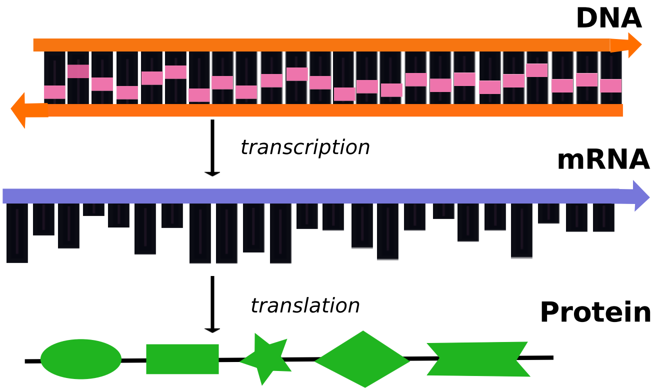

Figure 2.1: General transfers of biological sequential information. The genetic information flows from DNA to mRNA (transcription) and from mRNA to protein (translation). Figure based on [4].

studying transcription factors-DNA interactions there are at least three aspects to be aware of. First, the presence of a binding site is simply a clue of an expression control, and actual actions might not occur. Second, more than one transcription factor may bind the same intergenic region, and it is impossible to ascertain if they function in a cooperative or in a competitive manner. Third, a transcription factor can bind the DNA indirectly, i.e., through interactions with other transcription factors.

2.2 Gene regulatory networks

The understanding of bio-polymer interactions is a key part in the understanding of cell activities. Unfortunately, it is well known that cells are complex systems whose behaviour *emerges* from a seemingly chaotic interplay of bio-polymers and cannot be simply defined as the sum of its constituents. In fact, it is not possible to reliably predict cell behaviour despite a good knowledge of the fundamental laws governing individual components [8].

Interactions can be described as networks, in which bio-polymers are the vertices, and the relationships are the edges. Several biological networks have been introduced [103]. This thesis focuses on a particular kind of biological networks known as gene regulatory networks. A gene regulatory network describes how gene expression is controlled in cells. Since the expression of genes is directly controlled by transcription factors, a directed graph is used to model this kind of networks where vertices of the graph are genes and edges represent regulative interactions.

Gene regulatory networks (and biological networks at large) are characterized by several key properties that are important for both their modeling

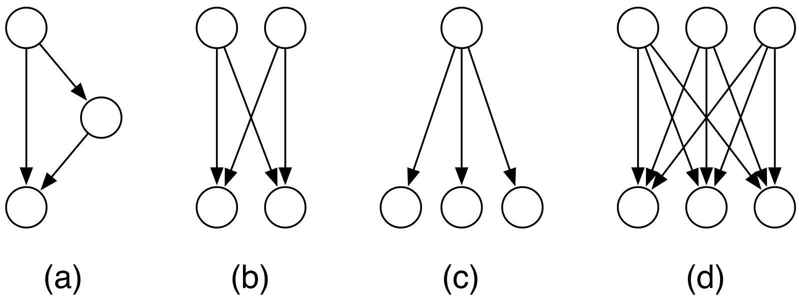

Figure 2.2: Some modules relevant in biological networks. (a) feed-forward loop, (b) bifan motif, (c) simple-input motif, and (d) multi-input motif. Figure based on [103].

and their understanding. First, biological networks show a *scale-free topology*, i.e., they are characterized by a power-law degree distribution. The probability that a node has k links is given by $P(k) = k^{-\gamma}$ [2, 14]. The values of γ determines many properties of the system: for $\gamma < 2$ the role of the highly connected nodes (*hubs*) becomes more important, for $\gamma > 3$, hubs are not relevant, while for $2 \leq \gamma \leq 3$ (as it is in biological networks), a hierarchy of hubs exists. Scale-free networks have a high degree of robustness against random failures, although they are sensitive to hub failures.

Biological networks are *modular*. Modules constitute discrete entities with dense internal functional connections and relatively looser external connections [76, 196]. Figure 2.2 lists a set of modules that has been shown to be functionally relevant in biological networks [38]. Functional modules group fractions of cell components that are involved in a relatively autonomous activity. However, functional modules are not rigid fixed structures: a given component may belong to different modules at different times (i.e., it can perform different functions). Modules play a key role in response to failures and account for some of the biological networks robustness.

Robustness is the property of a system to retain stability despite several perturbations, both internal and external [106, 107, 196]. This allows a slow degradation of system functions after a damage, avoiding catastrophic failures. Robustness is achieved by using several complex mechanisms such as: *i)* negative-feedback and feed-forward controls, *ii)* redundancy, whereby multiple components with equivalent functions are introduced as a backup, and *iii)* modularity, where sub-systems are physically of functionally isolated in order to not spread the failure.

2.3 Data sources

Nowadays, researchers have information about most of the cellular components, and additional pieces of information are available from literature, biological databases, and ontologies. In the following, we introduce several sources of biological information that we exploit in this thesis.

Gene expression data

The majority of the data available to researchers consists of gene expression measurements from DNA microarray experiments [180]. A single microarray experiment provides a snapshot of the expression level of thousands of genes measured in different experimental conditions.

The term *reference* is used to designate the results of a microarray experiment where the cell is in an untreated state. It could be used as a *control* datum for the discovering of how transcription changes when a cell is subject to a *treatment*. *Differentially expressed genes* are the genes whose expression profiles vary in a statistically significant way in treatments with respect to controls. Usually, they are identified by means of statistical techniques [24, 39, 190].

Microarray experiments produce two kinds of data: *time series data* and *steady state data*. The former refers to repeated measurements taken at specific time points. The latter may refer to measurements taken *i)* in an altered gene activity, *ii)* in presence of a perturbation, or *iii)* in different cell states. Examples of altered gene activities are knock-out and over-expression experiments. They refer to experiments where a gene is silenced or enhanced, respectively. Usually, these kinds of experiments affect a gene at a time. They are mainly used to understand how the altered gene interacts with others. Perturbation experiments, instead, may affect the expression levels of an unknown number of genes at the same time. Perturbation experiments are useful to understand how cells behave in presence of an environmental stress (e.g., lack of oxygen or heat shock) as well as under a chemical influence (e.g., drug administration). It is also possible that some experiments use a combination of different experimental settings. For instance, a perturbation could be done in conjunction with a gene alteration, or observed at different time points.

Microarray experiments are routinely used in almost every area of biomedical research. This leads to a remarkable growth of available gene expression data, and to a need for their free accessibility, evaluation and comparison. As a consequence, the Minimum Information About a Microarrays Experiment (MIAME) guidelines were proposed [22], and public repositories were built (e.g., Gene Expression Omnibus (GEO) [55] and ArrayExpress databases [149]). Moreover, computational techniques for microarray data analysis were developed [83, 94, 133]. For instance, the Bioconductor project and the statistical open source software R [64, 165] provides tools for the analysis and comprehension of high-throughput genomic data, such as statistical tools for the identification of differentially expressed genes.

Protein abundance data

Proteins are direct mediators of cellular processes. Thus, the measurement of protein levels is perhaps the most important indicator of cellular activities. The overwhelming majority of the evidence shows that, in most cases, it is fair to consider the gene expression levels (that are easier to measure) as estimators of the protein abundance. However, it is worth mentioning that the possibility of post-transcriptional activities may sometimes hinder their accuracy as proxies for protein levels and thus to measure protein abundance should be preferable.

The standard technology to measure protein levels is the mass spectronomy [191]. This technique is able to measure the protein expression levels both in time series and at steady state. Besides, the hybrid linear ion trap-Orbitrap mass spectrometer can characterize intact proteins, providing the opportunities to precisely identify and quantify them [207]. Unluckily, proteins measurements are still difficult to obtain due to the quick degradation rate of some of them.

Protein-DNA interaction data

Since the gene expression regulation happens by a physical interaction between transcription factors and DNA sequences, the identification of binding sites is crucial for understanding the interactions among genes.

Nowadays, the state-of-the-art approaches to map transcription factor binding sites are *i)* chromatin immunoprecipitation combined with genome-wide tiling array analysis [169], *ii)* chromatin immunoprecipitation followed by high-throughput sequencing techniques [96], and *iii)* Protein Binding Microarrays technology [26]. These technologies aim at determining the genome locations bound by a transcription factor, and at providing candidate genes that the transcription factor is likely to regulate. However, the binding site resolutions obtained from such technologies is not sufficient, and *motif discovery* algorithms are required to identify the sites precisely [29, 218]. Moreover, computational approaches are useful in modeling the protein-DNA binding specificity. Databases, such as JASPAR or TRANSFAC, are available for the storage of binding site models [178, 217].

Protein-protein interaction data

Most cellular activities are performed by means of protein interactions. Various methods have been described for protein interaction discovery; co-immunoprecipitation and two-hybrid system are two of the most popular techniques [160]. Public databases, such as BioGRID and STRING, exist to archive Protein-Protein-DNA interaction data [193, 199].

Annotation databases

Efforts have been made to create databases containing detailed information (*annotations*) about genes and gene products. Some annotations are created extracting the sought information from the literature by means of text mining techniques [7, 46]. In addition, manually curated annotations have been created leveraging existing literature or new experiments. Several databases of annotations have been built. Some of them, like the Gene Ontology project and the KEGG database, have been designed to support storing species-independent information. Others, like the SGD database, are instead tailored to store species-specific data.

Gene Ontology (GO). The Gene Ontology Consortium has developed three structured and controlled vocabularies (ontologies) for describing genes and gene products in terms of their associated biological processes, cellular components, and molecular functions [10, 40]. The practice of describing the activities or the localisation of a gene/gene product by associating it to GO terms is known as *annotation*. Besides, the Gene Ontology Consortium also provide tools that facilitate the creation, maintenance, browsing, and use of both the ontologies and the annotations.

Kyoto Encyclopedia of Genes and Genomes (KEGG). The Kyoto Encyclopedia of Genes and Genomes is a manually curated knowledge base of biological systems composed by several databases [112, 145]. It integrates genomic, chemical, and systemic functional information. The core component is represented by a set of interactions and reaction networks (*pathways*), joined with the definition of smaller pathway modules [99]. Besides, KEGG stores also information about: all the drugs approved in United States and Japan; the relationships existing among them; diseases; pathways; and diagnostic markers. Moreover, an ontology database representing functional hierarchies of various biological objects, and various computational tools are provided [100, 101].

Saccharomyces Genome Database (SGD). The Saccharomyces Genome Database stores manually curated information of *Saccharomyces cerevisiae* [35, 186]. It provides an encyclopedia of genome, genes and encoded proteins, as well as chromosomal features, their functions and interactions. SGD also oversees the *S. cerevisiae* genetic nomenclature, and provides several bioinformatic tools for mining and querying the stored data.

Part II

Knowledge Discovery

Chapter 3

Restructuring the Gene Ontology

The Gene Ontology represents a collaborative effort to provide a structured vocabulary for consistent gene descriptions. Although the Gene Ontology facilitates information retrieval, its structure may hide some useful knowledge, such as gene cooperation. As a consequence, automated tools may find it difficult to fully exploit the stored information. Conversely, such tools would benefit from a structure tailored to emphasise genes involved in the same activities or that are co-localized.

This chapter introduces a reorganization of the Gene Ontology. The *Restructured Gene Ontology* (RGO) is useful for unveiling gene involved in the same biological process, and for inferring new pieces of information about gene functions and localizations.

3.1 Introduction

The Gene Ontology (GO) is a structured and controlled vocabulary defined as a set of terms related by parenthood relationships forming a direct acyclic graph [10, 40, 41]. The GO is composed by three separate sub-ontologies representing three types of orthogonal aspects of gene/protein functions: the *Biological Process* sub-ontology describes the biological events in which a gene is involved; the *Molecular Function* sub-ontology indicates the biochemical activities that occur at the molecular level; the *Cellular Component* sub-ontology details the cellular places where the gene product carries out its functions.

Terms are the concepts in the sub-ontologies and vertices in the graph. Terms are described by a name and by a set of synonyms. For instance, the biological process *in which a cell irreversibly increases in size over time by accretion and biosynthetic production of matter similar to that already present* is described by the name *"cell growth"* and by the synonyms: *"cell expansion"*, *"cellular growth"*, *"growth of cell"*, and *"metabolism resulting in cell growth"*.

Edges define relationships between terms. Each edge has a type that belongs to the following set: *is_a, part_of, regulates, positively_regulates*, and *negatively_regulates*[1]. The relation *A is_a B* means that *A* is a subtype of *B*. For instance, *multidimensional cell growth is_a cell growth.* The relation *part_of* is used to represent the part-whole relationship. For instance, *detection of nuclear:cytoplasmic ratio part_of cell growth.* The *regulates* relationship specifies if one process directly affects the manifestation of another process. For instance, *regulation of cell growth regulates cell growth.*

Knowledge about genes/proteins is codified by means of *annotations*, i.e., associations among GO terms and genes/proteins, joint with specific references that describe the annotations themselves.

The areas of application of the GO are diverse. For instance, a common practice to gain a higher-level understanding of gene/protein functionalities is by querying the annotations stored in the GO [81]. A further research line is related to the automated measuring of gene functional similarity by using gene annotations. Several evaluation methods have been developed [110, 118, 170, 213].

The main goal in the design of the GO is the definition of a structured vocabulary for gene description. While this is a worthy goal, it is still true that some applications that relies on the GO (e.g., functional similarity measures) would benefit from a structure purposely built to emphasize genes involved in the same biological context, or containing interconnections among the sub-ontologies. As a consequence, such applications often find it difficult to fully exploit the information in the GO. These difficulties are likely to be amplified by the fact that the information content in the GO is not evenly spread through the ontology [6]. Motivated by these issues, several attempts to restructure the GO have been pursued [16, 95, 208]. A remarkable effort is represented by the so called "Cross-product extension" of the Gene Ontology that aims at normalizing the GO by explicitly stating the term descriptions in a form that can be used by reasoners, and at integrating the GO with other OBO ontologies [143].

In this chapter we introduce the *Restructured GO* (RGO), a reorganization of the GO that emphasizes the regulative connections between GO terms, and that links together the three sub-ontologies in order to gain a global view of biological mechanisms. We also show that RGO makes explicit information that was previously only implicitly represented in the GO structure by using a *Saccharomyces cerevisiae* gene as a case study.

[1] For the purposes of this dissertation, we do not distinguish among *regulates*, *positively_regulates*, and *negatively_regulates.* Thus, in the following, we simply write "*regulates*" in place of anyone of them.

3.2 Methods

Let us formalize the Gene Ontology as a pair $\langle T, E\rangle$, where T is a set of terms, and E is a set of edges, and let us define the RGO as a pair $\langle \mathcal{N}, \mathcal{E}\rangle$, where $\mathcal{N}$ is a set of *nodes* (each node is itself a set of GO terms), and $\mathcal{E}$ is a set of edges.

The construction of the RGO is a two step process. First nodes are created by grouping co-regulating terms, then the edges are built. In the following we detail how nodes and edges are created.

RGO nodes

To emphasize the functional correlation between genes we propose to merge into a single RGO node all those terms that participate to the same regulatory process and are neighbouring GO terms, i.e., nodes connected by *regulates* edges. Formally, let us consider the equivalence relation over T, defined as:

$$t \equiv_R t' \Leftrightarrow (t \ \mathit{regulates}\ t' \vee t' \ \mathit{regulates}\ t)$$

and set $\mathcal{N}^1$ to be the partition over T induced by $\equiv_R$. We define $\mathcal{N}$ as:

$$\mathcal{N} = \left\{ \mathit{extend}(n) | n \in \mathcal{N}^1 \right\},$$

where $\mathit{extend}(n)$ is a function that merges n with all singletons in $\mathcal{N}^1$ whose term t can be inferred *without ambiguity* to regulate or to be regulated by one of the terms in n. We say that an inference is ambiguous if t could be inferred to regulate or be regulated by terms belonging to different nodes. Inference about regulations is made using the rules given by the GO consortium:

$$is\_a \circ regulates \rightarrow regulates \tag{3.1}$$

$$regulates \circ is\_a \rightarrow regulates \tag{3.2}$$

$$regulates \circ part\_of \rightarrow regulates \tag{3.3}$$

These rules allows one to infer new relationships by combining two GO relations. Rule 3.1 states that if a term A *is_a* B, and B *regulates* C, then it can be inferred that A *regulates* C, i.e., if a general term B regulates a term C all its specifications regulate C as well. Rule 3.2 states that if a term A *regulates* B, and B *is_a* C, then it can be inferred that A *regulates* C, i.e., if a term A regulates a specific term B, it also regulates all its generalizations. Rule 3.3 states that if a term A *regulates* B, and B *part_of* C, then it can be inferred that A *regulates* C, i.e., if a term A regulates a component of the process, then it also regulates the entire process.

As mentioned, in using these rules we require that their application is not ambiguous, that is we require that a singleton is not merged into any node if the rules would imply merging it into different ones. Figure 3.1 reports an example of how RGO nodes are built. In the proposed example t_6 is merged with node $\{t_2, t_4\}$ since t_6 *is_a* t_4 and t_4 *regulates* t_2 (rule 3.1). The

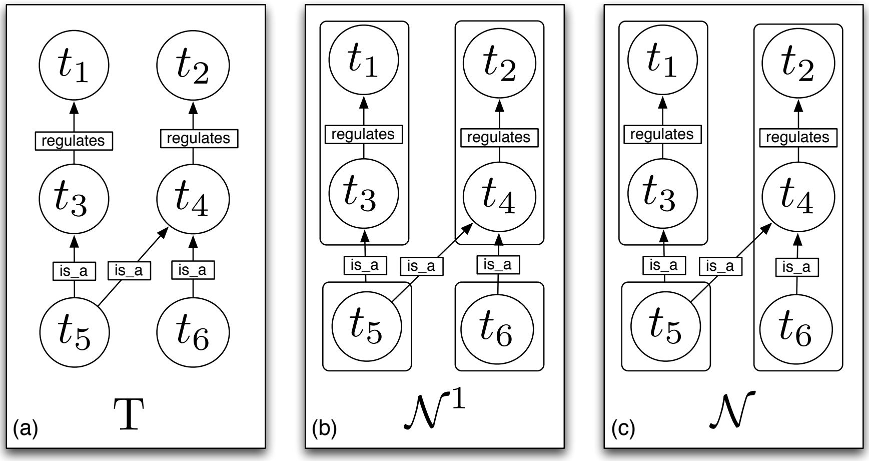

Figure 3.1: RGO node building procedure. Circles represent GO terms. Rounded rectangles represent RGO nodes. All edges are from the GO. In (a), an example ontology is shown. (b) shows the nodes in $\mathcal{N}^1$: all terms that are $\equiv_R$ equivalent are grouped into a single RGO node. (c) shows the resulting $\mathcal{N}$.

same cannot be done for t_5: rule 3.1 can be *ambiguously* applied in two different ways. In fact, t_5 *is_a* t_4 and t_4 *regulates* t_2, but also t_5 *is_a* t_3 and t_3 *regulates* t_1.

As a last remark, let us note that the definition of $\mathcal{N}^1$ and of *extend*$(\cdot)$ implies that all terms that are not involved in any regulation are not to be merged with other terms. Also, let us mention that the GO *molecular function* and the GO *cellular components* sub-ontologies do not contains any *regulates* edges. Hence, all terms in these two sub-ontologies are to be mapped to singletons.

RGO Edges

In the RGO, we distinguish between two kinds of edges: *native* and *cross-ontology* edges. A native edge is inferred from one or more of the existing GO edges (implying that native edges never connect distinct sub-ontologies); cross-ontology edges are those that link nodes belonging to different RGO sub-ontologies.

Native Edges. The creation of native edges is again a two step process: *i)* find GO edges that cross node boundaries and use them to create the RGO edges; *ii)* find and break resulting cycles.

In step *i)*, we start by considering all pairs of nodes (n, n') in $\mathcal{N} \times \mathcal{N}$ (with $n \neq n'$). An edge (n, n') is included in $\mathcal{E}$ if and only if there exist $t \in n, t' \in n'$ such that $(t, t') \in E$.

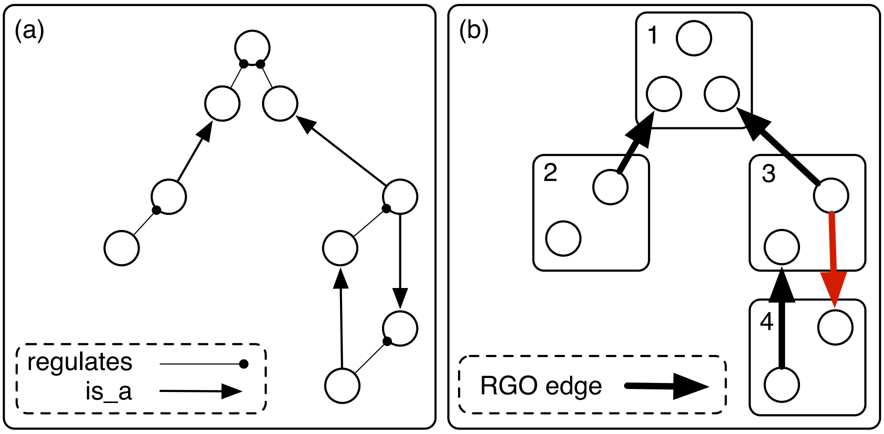

Figure 3.2: Example of how cycles form and could be broken during RGO construction.The figure reports: (a) a toy example of the GO that does not contain any cycle, and (b) the corresponding RGO. In the process of creating the RGO, the edges between nodes 3 and 4 form a cycle. By performing a breadth first search the edge drawn in red is chosen for removal.

Step *ii)* is necessary because even though cycles does not exist in the GO, they can be created as a consequence of merging terms during the creation of the RGO nodes. In Figure 3.2, an example of how this can happen is given. In breaking a cycle one needs to choose the best edge to remove among the ones that form the cycle. To this regard, we propose to start a breadth first search from the sub-ontologies roots and to remove all edges that are found to point back to already visited nodes. This strategy has the advantage of guaranteeing that a path is always retained between each node and the root of its sub-ontology.

Cross-Ontology Edges. Cross-ontology edges connect nodes in different RGO sub-ontologies based on a lexical similarity between biological descriptions associated to nodes. Specifically, we propose to interpret the sub-ontologies as a set of documents, and to create cross-ontology edges by means of a similarity measure between those documents. To this purpose, as detailed below, we represent nodes as vectors of *tf-idf* values [97]. Then, we use the cosine similarity between these vectors in order to decide whether two nodes are similar enough to be linked by a cross-ontology edge.

Let us recall that a name and a set of synonyms is defined for each term in the GO. Therefore, for each node n in RGO, we can construct a *description* ($d(n)$) of the node as the bag of words[2] that appear in the names and in the synonyms of all the terms in n. We interpret the node description as a

[2] We actually consider the *stem*, i.e., the base part of a word not including inflectional morphemes, rather than the word.

document, and the whole RGO as a corpus of documents. It is now possible to measure the salience of a word in a document $d(n)$ by means of the *tf-idf* measure. The *tf-idf* measure is calculated by weighting on one side how important the given word is in $d(n)$ (the *tf* component), and on the other side how important the word is in the corpus (the *idf* component). A word is considered salient for a document if it is very frequent in the document, but infrequent in the corpus. Formally, the *term-frequency* of word w in document $d(n)$ is defined as:

$$\mathit{tf}(\mathrm{w}, n) = \frac{\mathbf{1}_{d(n)}(\mathrm{w})}{|d(n)|},$$

where $\mathbf{1}_{d(n)}(\mathrm{w})$ denotes the number of occurrences of w in $d(n)$. The *inverse-document-frequency* of word w is defined as:

$$\mathit{idf}(\mathrm{w}) = \log \frac{|\mathcal{N}|}{|\{n \in \mathcal{N} | \mathrm{w} \in d(n)\}|}.$$

Finally, the *term-frequency/inverse-document-frequency* of word w in document $d(n)$ is defined as:

$$\mathit{tf\text{-}idf}(\mathrm{w}, n) = \mathit{tf}(\mathrm{w}, n) \cdot \mathit{idf}(\mathrm{w}).$$

In order to determine the lexical similarity between two RGO nodes, we propose to use the vector-space model [177] and the cosine similarity measure. In the vector-space model, each document is represented as a vector of *tf-idf* values. Cosine similarity is defined as the cosine of the angle that two vectors form. In formulae:

$$\mathit{cos\text{-}similarity}(n, n') = \frac{\mathrm{v}(n) \cdot \mathrm{v}(n')}{\|\mathrm{v}(n)\| \|\mathrm{v}(n')\|}, \tag{3.4}$$

where v(.) is the representation of $d(.)$ in the vector space.

We have now all the elements that are needed to specify how cross-ontology edges are added to the RGO. It is again a two step process: *i)* select a set $\mathcal{C}_n$ of candidate cross-ontology edges for each node n belonging to the *biological process* sub-ontology; *ii)* identify the *elected* cross-ontology edge $e \in \mathcal{C}_n$ to add to $\mathcal{E}$.

In step *i)*, we add an edge (n, n') to $\mathcal{C}_n$ if and only if the following conditions hold:

$$n \text{ is in RGO } \textit{biological process} \tag{3.5}$$

$$n' \text{ is not in RGO } \textit{biological process} \tag{3.6}$$

$$\exists \mathrm{w} \in d(n) \cap d(n') : \mathit{tf\text{-}idf}(\mathrm{w}, n) \geq \theta_1 \tag{3.7}$$

$$\mathit{cos\text{-}similarity}(n, n') \geq \theta_2 \tag{3.8}$$

that is: we add only edges exiting from nodes in the *biological process* and entering into nodes in the *molecular function* or in the *cellular component* sub-ontologies (Expressions (3.5) and (3.6)); we only consider pairs of nodes that share a word that is salient in document $d(n)$ (Expression (3.7)); we add the edge only if the cosine similarity is significant (Expression (3.8)). The role of θ_1 is to allow one to avoid considering nodes which are only marginally related to n. The threshold θ_2 allows one to retain only edges which guarantee a minimum similarity between the involved nodes.

In step *ii)*, we select the elected edge $\mathfrak{e} = (n, n')$ such that:

$$\mathfrak{e} = argmax_{(n,n') \in \mathcal{C}_n} \ cos\text{-}similarity(n, n'). \tag{3.9}$$

Finally, $\mathfrak{e}$ is added to $\mathcal{E}$.

RGO annotations

In the RGO, we distinguish between two kinds of annotations: *original* and *inferred*. An original annotation is one that is already present in the GO, i.e., the original RGO annotations of a node n are the union of all GO annotations of the GO terms that belong to n. An inferred annotation is derived by following a cross-ontology edge, i.e., the inferred RGO annotations of a node n' are the union of all original RGO annotations of the RGO nodes n that are connected to n' by means of a cross-ontology edge (n, n').

As a last remark, let us recall that since by construction RGO nodes in the *molecular function* and in the *cellular component* sub-ontologies are singletons their original annotations are identical to the GO annotations.

3.3 Results

All experiments described in this section are built on an RGO constructed using the GO database published on November 2011. We set $\theta_1 = 0.2$ and $\theta_2 = 0.1$. As the number of cross-ontology edges added is a function of these thresholds, lower threshold values produce richer representation at the expense of higher computational costs. We choose these values to retain most of the significant information while making computationally feasible the process of creating and using the RGO,

In the RGO *biological process* sub-ontology we count 15,588 nodes (versus the 21,551 GO terms). We add 12,400 cross-ontology edges directed to the RGO *cellular component* sub-ontology, and 13,786 cross-ontology edges directed to the RGO *molecular function* sub-ontology. Table 3.1 shows the number of annotations (both original and inferred) in the three sub-ontologies for several organisms, namely *Arabidopsis thaliana*, *Escherichia coli*, *Homo sapiens*, *Saccharomyces cerevisiae*, and *Schizosaccharomyces pombe*.

In the following we show that RGO nodes make explicit useful information that was previously only implicitly represented in the GO. Specifically, we point out that *i)* the groups induced by RGO nodes allow the discovery of gene shared annotations, enhancing the understanding about gene cooperation, and *ii)* the inferred annotations add important pieces of information

Table 3.1: Number of RGO annotations for several organisms.

	biological process		***cellular component***		***molecular function***	
Organism	**Original**	**Inferred**	**Original**	**Inferred**	**Original**	**Inferred**
A. thaliana	47,987	0	44,765	23,509	55,209	26,089
E. coli	49,252	0	24,649	25,214	66,585	29,730
H. sapiens	125,094	0	102,956	60,370	99,625	67,683
S. cerevisiae	29,365	0	32,826	16,529	25,160	17,630
S. pombe	13,015	0	14,503	10,307	8,885	10,738

about gene functionalities and localizations that are hidden in the GO. These findings may enhance automated analysis, such as gene profiling and clustering (see Chapter 4), statistical enrichment, as well as the evaluation of gene functional similarities [210]. We use as case study a specific biological process, namely the cell cycle. In particular, we use as an example a *S. cerevisiae*'s gene, namely CDC20.

For the sake of self-containedness, it is useful to describe the cell cycle, i.e., the series of events leading to the cell division and duplication. Cell cycle goes through two main phases: the S (Synthesis) phase, and the M (Mitosis) phase. In the S phase, chromosomes are duplicated. During M phase, the replicated chromosomes are segregated, and the cell splits in two parts. The M phase itself is composed by two steps: the *metaphase*, i.e., when the chromosomes get aligned in the middle of the cell, and the *anaphase*, i.e., when chromosomes move to opposite poles of the cell. Usually, the S phase and the M phase are separated by gap phases called G_1 and G_2, during which cell cycle progression is regulated. Specifically, the cell cycle progression is controlled by a complex formed by two proteins: the cyclin-dependent kinases (CDK) and the cyclins. After a CDK-cyclin complex has performed its function, the associated cyclin is degraded by the proteasome [5, 141].

CDC20 is a cell cycle activator. According to the GO annotations, CDC20 is concerned with the protein catabolic process and with the activities connected to the activation of the anaphase stage for the mitotic metaphase/anaphase transition. Annotations refer also to the segregation of sister-chromatid to opposite poles of the cell using the mitotic spindle elongation. Several other annotations concern with details about these processes such as the protein binding activity, and the anaphase-promoting complex activity. All these processes occur in the nucleus and we find annotations also about it and about multi-protein complexes that regulate the stages of anaphase. Table 3.2 shows the GO terms annotated by CDC20.

Before delving into the experimentation, it is important to show that the groups induced by RGO nodes are adequate from a biological point of view. To this purpose, we singled out the *original* annotations of CDC20 over the RGO. Since only the RGO *biological process* sub-ontology contains non-singleton nodes, here and in the following section, we analyze only this

Table 3.2: Terms annotated by CDC20 over the GO. These annotations correspond to the original annotations over the RGO.

Sub-ontology	GO terms description
biological process	activation of mitotic anaphase-promoting complex activity cell cycle cell division mitosis mitotic metaphase/anaphase transition mitotic sister chromatid segregation mitotic spindle elongation protein catabolic process
cellular component	anaphase-promoting complex mitotic checkpoint complex nucleus
molecular function	protein binding, bridging ubiquitin-protein ligase activity

sub-ontology. Table 3.3 shows such nodes, and terms they include. A coarse analysis reveals that original annotations over the RGO get a more general view of gene functioning with respect to the GO. Indeed, the RGO can be best thought of as an abstraction over the GO, and once the analysis over the abstract structure is complete, nothing hinders to return to the GO to make finer grained inferences.

Most of the identified nodes are self-explanatory. The exception is the RGO node describing the *cell cycle and its regulation*, which correctly contains all the GO terms related to cell cycle and to its regulation (both positive and negative). We note that it also contains the GO terms describing the regulation of cyclin-dependent protein kinase activity. As previously explained, CDK-cyclin complex regulates the progression through the cell cycle.

Gene cooperation information

The identification of genes annotated over the same GO term is a widespread approach to discover genes that participate to the same biological activity. One of the techniques in widespread usage to this aim is the *enrichment* of genes for GO terms [51, 225]. A set of genes is said to be enriched for a GO term t if the proportion of genes within the set that are annotated over t exceeds the number that would be expected by chance. Enriched GO terms describe the activities of the set of genes. Our aim is to show that by using the original RGO annotations and by leveraging the RGO structure, it is possible to gain information about gene cooperation, i.e., to discover shared annotations that are not directly codified in the GO structure.

A thorough analysis of the RGO nodes over which CDC20 is annotated reveals that the RGO nodes:

Table 3.3: GO terms belonging to RGO nodes annotated by CDC20 over the *biological process* sub-ontology. Only original annotations are reported.

RGO node description	**GO terms description**
cell-cycle and its regulation	cell cycle regulation of cell cycle negative regulation of cell cycle positive regulation of cell cycle regulation of cyclin-dependent protein kinase activity negative regulation of cyclin-dependent protein kinase activity positive regulation of cyclin-dependent protein kinase activity
cell division and its regulation	cell division regulation of cell division negative regulation of cell division positive regulation of cell division
mitosis and its regulation	mitosis regulation of mitosis negative regulation of mitosis positive regulation of mitosis
mitotic metaphase/anaphase transition and its regulation	mitotic metaphase/anaphase transition regulation of mitotic metaphase/anaphase transition negative regulation of mitotic metaphase/anaphase transition positive regulation of mitotic metaphase/anaphase transition
mitotic sister chromatid segregation and its regulation	mitotic sister chromatid segregation regulation of mitotic sister chromatid segregation negative regulation of mitotic sister chromatid segregation
mitotic anaphase-promoting complex activity	activation of mitotic anaphase-promoting complex activity negative regulation of mitotic anaphase-promoting complex activity inhibition of mitotic anaphase-promoting complex activity anaphase-promoting complex-dependent proteasomal ubiquitin-dependent protein catabolic process
protein catabolic process and its regulation	protein catabolic process regulation of protein catabolic process negative regulation of protein catabolic process positive regulation of protein catabolic process
mitotic spindle elongation and its regulation	mitotic spindle elongation regulation of mitotic spindle elongation

- *mitosis and its regulation* annotates (among others) BFR1, ECO1, NIS1, PSE1, and RTT101;
- *cell cycle and its regulation* annotates (among others) BCK2, CCL1, CDC33, CDC36, CDC39, CLG1, CTK2, ESA1, FKH1, FKH2, HSL1, NME1, PCL6, PCL8, PHO80, PTC2, PTC3, RSC3, SEM1, SIC1, SKM1, and SSN8;
- *protein catabolic process and its regulation* annotates (among others) MAC1, MRK1, RPN1, RPN2, RPN3, and RPN4.

It is worth mentioning that in the GO structure none of the listed genes shares any annotations with CDC20. To assess the reliability of the association among these sets of genes and the specific biological activities, we use the FunSpec web application [172]. It takes as input a list of genes and it outputs a summary of functional classes that are functionally enriched in that list. FuncSpec provides a p-value representing the probability that the intersection of a given list with any given functional category occurs by chance. For each set of annotations listed above, we analyzed all the MIPS Functional Classification with a p-value lower than 0.01. The functional classes have been downloaded from the MIPS database [137]. If genes in these sets perform an activity related to the one described by the node, we would find a significative enrichment for the related functional classes. The functional enrichment confirm the aforementioned finding reporting that the most significant functional classes are, respectively, *mitotic cell cycle* (p-value of $5.95e^{-8}$), *mitotic cell cycle and cell cycle control* (p-value of $5e^{-10}$), and *proteasomal degradation* (p-value of $5.16e^{-8}$). Let us recall that proteasomal degradation is a catabolic process essential for many cellular processes, including the cell cycle [5]. Hence, notwithstanding that these genes do not share any GO annotation with CDC20, their function is strictly connected with it.

Extraction of new pieces of information

In the following we focus on the assessment of the quality of cross-ontology edges and show that inferred annotations add useful pieces of information about gene functions and localizations. Specifically, we show how the cross-ontology edges allow one to retrieve interesting nodes in the *cellular component* and in the *molecular function* sub-ontologies. It is worth stressing that the retrieved nodes may not be reached using the original annotations: this information is specifically unveiled by the proposed tool and we would argue that it could be hardly found using the GO alone.

We analyze the nodes annotated by CDC20 -just to maintain the same context as before. Let us consider the set N_{CDC20} of nodes in *biological process* annotated by CDC20. Table 3.4 reports the nodes in the *cellular component* sub-ontology on the receiving end of cross-ontology edges that start from nodes in N_{CDC20}. Only one original annotation can be found among these

Table 3.4: Description of RGO nodes belonging to the *cellular component* sub-ontology that are reached by cross-ontology edges starting from nodes in the *biological processes* sub-ontology annotated by CDC20. They correspond to the inferred annotations. Similarity values are evaluated using Equation (3.4).

Source node description	**Destination node description**	**Similarity value**
cell-cycle and its regulation	cyclin-dependent protein kinase 5 holoenzyme complex	0.38
cell division and its regulation	cell division site	0.75
mitotic metaphase/anaphase transition and its regulation	metaphase plate	0.45
mitotic sister chromatid segregation and its regulation	mitotic spindle	0.33
mitotic anaphase-promoting complex activity	anaphase-promoting complex	0.67
protein catabolic process and its regulation	protein complex	0.57
mitotic spindle elongation and its regulation	mitotic spindle	0.71

Table 3.5: Description of RGO nodes belonging to the *molecular function* sub-ontology that are reached by cross-ontology edges starting from nodes in the *biological processes* sub-ontology annotated by CDC20. They correspond to the inferred annotations. Similarity values are evaluated using Equation (3.4).

Source node description	**Destination node description**	**Similarity value**
cell-cycle and its regulation	cyclin-dependent protein kinase 5 activator regulator activity	0.52
cell division and its regulation	MHC class I receptor activity	0.15
mitotic metaphase/anaphase transition and its regulation	mitotic anaphase-promoting complex activity	0.84
mitotic sister chromatid segregation and its regulation	mitotic anaphase-promoting complex activity	0.23
mitotic anaphase-promoting complex activity	mitotic anaphase-promoting complex activity	0.73
protein catabolic process and its regulation	14-3-3 protein binding	0.48
mitotic spindle elongation and its regulation	protein-glycine ligase activity, elongating	0.43

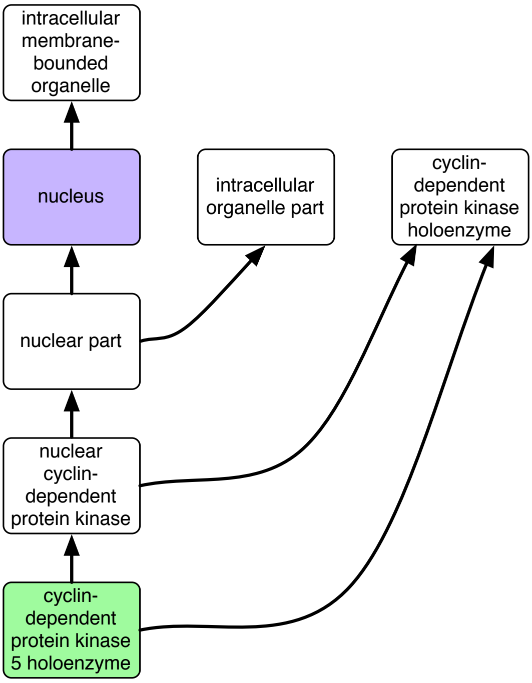
intracellular membrane-bounded organelle
nucleus
intracellular organelle part
cyclin-dependent protein kinase holoenzyme
nuclear part
nuclear cyclin-dependent protein kinase
cyclin-dependent protein kinase 5 holoenzyme

Figure 3.3: RGO fragment. Portion of RGO *cellular component* annotated by CDC20. The violet node is annotated by an original annotation. The green node is annotated by an inferred annotation.

nodes (*anaphase-promoting complex*). All other original annotations refer to more general nodes (i.e., nodes on higher levels of the ontology hierarchy). In this particular case, see Figure 3.3, the nearest original annotation in the *cellular component* refers to the *nucleus* which is three levels above the inferred annotation *cyclin-dependent protein kinase 5 holoenzyme complex*.

As a last remark, let us note that a node can be pointed to by several RGO *biological process* nodes. For instance, the *mitotic spindle* node is connected with two nodes, namely *mitotic sister chromatid segregation and its regulation* and *mitotic spindle elongation and its regulation*.

Table 3.5 shows the nodes pointed by the cross-ontology edges in the *molecular function* sub-ontology. In contrast with the previous example, here one of the found relationships is not accurate. Indeed, the node *MHC class I receptor activity* is not related to cell division nor to its regulation. The wrong association is due to two stems shared by the nodes, namely *cell* and *activ*. We note that the problem is mitigated by the fact that the inferred similarity value is the smallest of the bunch. A higher similarity threshold as well as a more sophisticated similarity measure among nodes would avoid these misassociations. For instance, one may replace the cosine similarity with a measure that takes into account the semantics of node descriptions (e.g., by leveraging well-known tools from the natural language research field, such as WordNet [138] or Obol [142]).

3.4 Conclusions

In this chapter we presented the Restructured Gene Ontology, a reorganization of the Gene Ontology. We showed that the RGO nodes allow the discovery of gene shared annotations, improving the understanding of gene cooperation, and that the inferred annotations add pieces of information that are not easily found in the the GO. As a future work we plan to consider other hints for placing cross-ontology edges. A particular interesting approach is to take into consideration whether two terms share the same annotations across all the genomes.

It must be emphasized here that we are *not* proposing the RGO as a replacement for the GO. The RGO is to be regarded as an accompanying tool specifically tailored to draw attention to gene cooperation, activity and localization.

Chapter 4

Metadata-driven Biclustering of Gene Expression Data

In the last decade, microarray experiments became the most popular approach to screen thousands of genes in different experimental conditions so producing enormous amount of data. Thus, the development of new computational techniques for microarray data analysis as well as for formulating new biological hypotheses is a need sorely felt by the biological community.

This chapter describes a new biclustering approach that leverages background information. As a case study we develop a tool that creates homogeneous biclusters which recover transcriptional modules.

4.1 Introduction

Clustering allows to partition data into groups (*clusters*) such that each data object is similar to all objects within the same group and is dissimilar from any other object belonging to any other group [94]. Clustering techniques have been commonly used in microarray data analysis. They enable the discovery of homogeneous gene (or experiment) groups based on a distance measure quantifying the degree of correlation of expression profiles [56]. In this context, the goal of clustering algorithms is to group together genes or experimental conditions (*samples*) sharing similar expression profiles.

By applying a clustering approach to both gene and sample dimensions it is possible to produce a "grid of clusters". A limitation of this "traditional" clustering technique is that it is applied on gene or sample sets independently. Indeed, clustering genes and samples simultaneously is different than clustering them separately, since in the former case the metric to be optimized needs to take into account the correlation between the partition of genes and the partition of samples. To exceed this limitation of clustering algorithms, biclustering has been proposed [133]. Biclustering algorithms examine gene and sample dimensions simultaneously, enabling the discovery of more coherent and meaningful groups (*biclusters*). In microarray data analysis, biclusters are potentially overlapping groups of genes that show similar activity patterns under a specific subset of experimental conditions. The biclustering process allows the identification of potential transcriptional modules, i.e., self-consistent subsets of co-expressed (and thus co-regulated) genes and of experimental conditions inducing this co-regulation [89]. Grouping genes into modules reduces the effective complexity of a data set, thus enabling a number of expensive analysis tools that cannot be performed over thousands of genes at a time (e.g., the reverse engineering of a regulatory network).

Many biclustering methods tailored for gene expression data analysis have been developed so far. For instance, Cheng and Church define a bicluster as a subset of genes and a subset of samples having a small *Mean Squared Residue Score* (MSRS) [34]. When MSRS is equal to 0, the bicluster contains genes having the same profiles on bicluster conditions. When MSRS is greater than 0, genes or samples can be removed so to decrease this value. The Cheng and Church's algorithm finds maximal size biclusters such that the MSRS is smaller than a given threshold δ. Tanay *et al.* describe a heuristic method, called SAMBA, that combines a graph-theoretic approach with a statistical data model [201]. SAMBA models the gene expression matrix as a bipartite graph, and biclusters as sub-graphs. It uses a likelihood score to assess the significance of each distinct sub-graph. Ihmels *et al.* propose the *Iterative Signature Algorithm* (ISA) [89, 90]. ISA starts with a random bicluster and iteratively updates it in order to improve a scoring function that captures the notion of transcriptional modules.

A further limitation of both clustering and biclustering approaches is that they mainly use distance metrics based only on expression levels. Indeed, these metrics are not optimized to capture biologically meaningful groups. Thus, several works proposed to define distance metrics based on additional sources of information (*metadata*), such as the Gene Ontology, biological networks, operon annotations, intergenic distances, and transcriptional co-responses [21, 72, 195]. These works derive some metrics from the metadata, and combine them with classical metrics on gene expression values. Unfortunately, these approaches have been proposed for *simple* clustering only, also because of the difficulty in designing new measures able to combine different sources of information. Nonetheless, in data mining context, several solutions are available for the exploitation of knowledge coming from metadata. As proposed by Schifanella *et al.* [182], these methodologies can be classified into three main groups: *metadata-injection*, *metadata-constrained*, and *metadata-driven* approaches. Injection-based methods combine metadata information and original data in a pre-processing step before the actual bicluster process starts. Constrained methods use the metadata information for limiting the admissible grouping in the biclustering process. Metadata-driven methods modifies existing biclustering algorithms so to choose, at each biclustering step, grouping both improving distance measure and preserving contextual relationships implied by metadata. To the best of our knowledge, only one work exists that exploits metadata-injection or metadata-driven approaches for gene expression data biclustering [168]. A few works addressed the problem of finding partitions of gene expression data under constraints [42, 156, 157, 209].

In this chapter we propose a new metadata-driven method for the biclustering of gene expression data.

4.2 Methods

Let us first introduce some notation. Let $A \in \mathbb{R}^{m \times n}$ denote a gene-condition expression matrix. Let a_{ij} be the expression level corresponding to the ith gene under the jth condition. Let $I \subseteq \{1, \ldots, m\}$, $|I| = k$ and $J \subseteq \{1, \ldots, n\}$, $|J| = l$ be clusters of genes and conditions respectively. A bicluster $B \in \mathbb{R}^{k \times l}$ is a submatrix of the matrix A specified by the pair (I, J), in formulae: $B = \{a_{ij} | i \in I, j \in J\}$. The problem addressed by a biclustering algorithms is the identification of a set of biclusters such that each bicluster $B_h = (I_h, J_h)$ satisfies some homogeneity conditions. The notation $A_{I,J}$ refers to the submatrix of A formed by the rows specified by I and by the columns specified by J. When one of the two sets is intended to contain all possible indices, a dot is used instead. For instance, the notation $A_{\cdot,J}$ refers to the submatrix of A containing all rows, but only the columns specified by J.

Let us identify, for both genes and samples, a set of characteristics (*features*) describing genes/samples themselves, i.e., metadata information. Here, without loss of generality, we assume that features are binary valued vectors. Sets of features are represented by a Boolean matrix M. Given an object (gene/condition) p and a feature f, we set M_{pf} to `true` if p has the characteristic described by f, `false` otherwise. In formulae:

$$M_{pf} = \begin{cases} \texttt{true} \text{ if } p \text{ has feature } f, \\ \texttt{false} \text{ otherwise.} \end{cases} \tag{4.1}$$

By leveraging these sets of features, we define two distance matrices, namely $D^G \in \mathbb{R}^{m \times m}$ (the gene distance matrix), and $D^C \in \mathbb{R}^{n \times n}$ (the sample distance matrix). Each matrix entry D_{pq} is set to the distance between the pth and the qth object. Specifically, the distance of two objects (p, q) is evaluated using the *Tanimoto distance* [173]:

$$T_d(p, q) = -log_2 T_s(p, q),$$

where T_s is the *Tanimoto similarity*, defined as:

$$T_s(p, q) = \frac{\sum_f \mathbf{1}(M_{pf} \wedge M_{qf})}{\sum_f \mathbf{1}(M_{pf} \vee M_{qf})},$$

where $\mathbf{1}(b)$ assumes the value 1 if b is equal to `true`, 0 otherwise. Tanimoto similarity computes the ratio of the number of features set in both the vectors to the number of features set in one or the other vector.

Note that the Tanimoto distance is not a distance metric, because it violates the triangle inequality, which requires that for any three objects p, q, and r: $T_d(p,r) \leq T_d(p,q) + T_d(q,r)$. For instance, consider following matrix:

$$M = \begin{bmatrix} \texttt{true} & \texttt{false} \\ \texttt{true} & \texttt{true} \\ \texttt{false} & \texttt{true} \end{bmatrix},$$

and assume that p, q, and r are described by the first, second and third row of the matrix respectively. The Tanimoto distances among these objects are:

$$\begin{aligned} T_d(p,q) &= T_d(1,2) = -log_2 \frac{\sum_f \mathbf{1}(M_{1f} \wedge M_{2f})}{\sum_f \mathbf{1}(M_{1f} \vee M_{2f})} \\ &= -log_2 \frac{\mathbf{1}(\texttt{true} \wedge \texttt{true}) + \mathbf{1}(\texttt{false} \wedge \texttt{true})}{\mathbf{1}(\texttt{true} \vee \texttt{true}) + \mathbf{1}(\texttt{false} \vee \texttt{true})} = -log_2 \frac{1}{2} = 1, \end{aligned}$$

$$\begin{aligned} T_d(p,r) &= T_d(1,3) = -log_2 \frac{\sum_f \mathbf{1}(M_{1f} \wedge M_{3f})}{\sum_f \mathbf{1}(M_{1f} \vee M_{3f})} \\ &= -log_2 \frac{\mathbf{1}(\texttt{true} \wedge \texttt{false}) + \mathbf{1}(\texttt{false} \wedge \texttt{true})}{\mathbf{1}(\texttt{true} \vee \texttt{false}) + \mathbf{1}(\texttt{false} \vee \texttt{true})} = -log_2 \frac{0}{2} = \infty, \end{aligned}$$

$$\begin{aligned} T_d(q,r) &= T_d(2,3) = -log_2 \frac{\sum_f \mathbf{1}(M_{2f} \wedge M_{3f})}{\sum_f \mathbf{1}(M_{2f} \vee M_{3f})} \\ &= -log_2 \frac{\mathbf{1}(\texttt{true} \wedge \texttt{false}) + \mathbf{1}(\texttt{true} \wedge \texttt{true})}{\mathbf{1}(\texttt{true} \vee \texttt{false}) + \mathbf{1}(\texttt{true} \vee \texttt{true})} = -log_2 \frac{1}{2} = 1, \end{aligned}$$

and thus $T_d(p,r) > T_d(p,q) + T_d(q,r)$.

It is to be noted that in the biological context semi-metrics (distances that do not satisfy the triangle inequality) usually perform better than full-fledged metrics as it is often the case that two genes are involved in no common activities, but both share a common function with a third gene.

Metadata-Driven ISA

In order to introduce metadata information inside a biclustering algorithm, we modified the Iterative Signature Algorithm (ISA). The algorithm we propose is called Md-ISA and is summarized in Algorithm 1, where the procedure for discovering a single bicluster is described. Multiple biclusters can be discovered by changing the score thresholds (t^G and t^C) as well as the random seed. Lines 6-7 and 9-10 refer to the standard ISA implementation. Lines 5 and 8 refer to the refinement procedure, called MetaRef. This procedure uses the metadata information contained in the distance matrices to adjust biclusters by refining the set of genes/samples they includes. The details of MetaRef implementation will be explained in the next section.

Algorithm 1 MD-ISA: Metadata-driven ISA

Input: A^G, A^C, m, n, D^G, D^C, t^G, t^C, δ_r, δ_e, N
Output: $B_h = (I, J)$

```
1:  Initialize: assign to $I$ a random sub set of $\{1, \ldots, m\}$, $s^G = \mathbf{1}^{1 \times m}$,
    $J = \varnothing$, $I' = \varnothing$, $J' = \varnothing$, $n = 0$
2:  repeat
3:      $n \leftarrow n + 1$
4:      $I' \leftarrow I$; $J' \leftarrow J$
5:      MetaRef($I'$, $m$, $D^G$, $\delta_r$, $\delta_e$)
6:      $s^C \leftarrow s^G_I \times A^G_{I,\cdot}$
7:      $J' \leftarrow J' \cup \{j' \in \{1, \ldots, n\} | s^C_{j'} \geq t^C \sigma^C\}$
8:      MetaRef($J'$, $n$, $D^C$, $\delta_r$, $\delta_e$)
9:      $s^G \leftarrow s^C_J \times (A^C_{\cdot,J})^T$
10:     $I' \leftarrow I' \cup \{i' \in \{1, \ldots, m\} | s^G_{i'} \geq t^G \sigma^G\}$
11: until $(I = I' \wedge J = J') \vee (n > N)$
```

MD-ISA is a two step iterative procedure. It starts from a random set of genes I, to which a default *gene score* (s^G) of 1 is assigned, and from two gene expression matrices, A^G and A^C, that are built from A by normalizing it so to have zero mean and unit variance with respect to genes and conditions, respectively. In the first step, the set of genes is refined by the METAREF procedure (line 5). Let us emphasize that this refinement process is performed once immediately after the random initialization, thus allowing the actual biclustering step to take advantage of starting from a set of more homogeneous genes. Afterwards, the change in the weighted average expression for each condition is evaluated using the gene scores as weights (line 6). The obtained average values are called *condition scores* (s^C). Only conditions with a score greater then a threshold t^C are retained (line 7). In the second step, the METAREF procedure is evaluated again (line 8). Then, the change in the weighted average expression for the retained conditions is evaluated for each gene using the condition scores as weights, and the gene score updated (line 9). Only genes with a score greater then a threshold t^G are retained (line 10). These two steps are repeated until the set of genes and the set of conditions do not change anymore, i.e., a bicluster is identified. It is worth mentioning that the METAREF procedure may sometimes make the whole algorithm to diverge. In fact, it is possible that a set of objects is added (respectively removed) by the main biclustering algorithm, and then removed (added) by the METAREF procedure, and then added (removed) again, leading to a *ping-pong* behaviour. To avoid this possibility the algorithm is stopped also if the number of iterations is larger than N (line 11).

Algorithm 2 MetaRef: Metadata-driven algorithm for cluster refinement

```
Input: $I$, $m$, $D^G$, $\delta_r$, $\delta_e$
Output: $I$

1: $\hat{i} \leftarrow argmin_{i \in I} \sum_{\forall i' \in I, i \neq i'} D^G_{i,i'}$
2: $\bar{d}_I \leftarrow \frac{\sum_{\forall i,i' \in I, i \neq i'} D^G_{i,i'}}{2(|I|-1)}$

3: for all $i \in I$ do
4:     if $D^G_{i,\hat{i}} > \delta_r \bar{d}_I$ then
5:         $I \leftarrow I \setminus \{i\}$

6: $\bar{d}_I \leftarrow \frac{\sum_{\forall i,i' \in I, i \neq i'} D^G_{i,i'}}{2(|I|-1)}$

7: for all $i' \in \{1, \ldots, m\}$, $i' \notin I$ do
8:     if $D^G_{i',\hat{i}} \leq \delta_e \bar{d}_I$ then
9:         $I \leftarrow I \cup \{i'\}$
```

Metadata-driven refinement procedure

In this section, we describe a new metadata-driven procedure, called MetaRef, that is used inside the Md-ISA algorithm. MetaRef is summarized in Algorithm 2. We describe the application of the procedure to the gene dimension with the understanding that it can be applied symmetrically on the sample dimension.

The first step of our algorithm is the selection of a cluster representative $\hat{i}$ (line 1). The representative is defined as:

$$\hat{i} = argmin_{i \in I} \sum_{\forall i' \in I, i \neq i'} D^G_{i,i'},$$

where $D^G_{i,i'}$ is the distance value between the ith and the i'th genes. Thus, $\hat{i}$ is set to the object closest to objects in I. Afterwards, the algorithm evaluates the cluster average distance $\bar{d}_I$ (line 2) as:

$$\bar{d}_I = \frac{\sum_{\forall i,i' \in I, i \neq i'} D^G_{i,i'}}{2(|I|-1)}.$$

All the genes belonging to I that are distant from the representative more than δ_r times the average cluster distance $\bar{d}_I$ are removed from the cluster (line 3-5). Then, the cluster average distance is updated to reflect the change in the bicluster composition (line 6). Finally, all the genes not belonging to I having a distance from $\hat{i}$ smaller or equal than δ_e times the new average cluster distance are added to I (line 7-9). This step add to the bicluster all the objects that are likely to be related, but that have been excluded by the main biclustering process so far.

As a last remark, we note that objects for which no information is available are not included in metadata matrices. As a consequence, they are not affected by the metadata-driven procedure and the biclustering algorithm is still able of grouping objects whose properties are unknown.

4.3 Results

To show the effectiveness of our approach we compared the results obtained by MD-ISA to those obtained by ISA. Moreover, we show that MD-ISA is effective in finding transcriptional modules.

For the experiments we used the gene expression data set created by Hughes *et al.* (Rosetta yeast compendium) [84]. This data set is composed by 300 full-genomes microarray experiments of *Saccharomyces cerevisiae.* Experiments correspond to mutations in both characterized genes and uncharacterized open reading frames as well as to treatments with compounds having a known molecular target. All the experiments were conducted under a single growth condition, allowing the direct comparison of all genes over all profiles. From this data set we selected 6514 genes having less than 30 missing values, and 276 samples corresponding to deletion mutants. Having chosen experimental conditions that explicitly refer to (mutated) genes only has one important side effect: the same features and the same evaluation metric can be used on both the gene and the sample dimensions.

To assess the quality of the obtained biclusters we used the Biological Homogeneity Index (BHI) [47]. The BHI measures whether, on average, genes belonging to the same cluster also belong to the same functional class. It is evaluated as:

$$BHI(C) = \frac{1}{h}\sum_{i=1}^{h}\frac{1}{n_i(n_i-1)}\sum_{p,q\in C_i, p\neq q}\mathbf{1}\left(\Phi(p)\cap\Phi(q)\neq\varnothing\right),$$

where Φ is a function mapping each gene $g \in G$ to a subset of the functional classes $F = \{\mathtt{f}_1,\ldots,\mathtt{f}_k\}$ describing its activity (specifically, $\Phi(g) = \{\mathtt{f}_i \in F \mid g \text{ is annotated over } \mathtt{f}_i\}$), and n_i is the number of functionally annotated genes in C_i, i.e., $n_i = |\{g \in C_i | \Phi(g) \neq \varnothing\}|$. We chose as functional classes gene mutant phenotypes. A mutant phenotype is the observable effect that a single mutation has on an organism. To mutate a gene is a common experimental design in order to understand processes the gene is involved in. We used the phenotype data collected by the SGD project [185].

BHI ranges between 0 and 1. A good biclustering algorithm should have high BHI. Since BHI has been designed to evaluate clusters, it cannot be directly applied to biclustering algorithms. However, since in this case conditions explicitly refer to genes, we still applied this metric to the biclustering results by evaluating each dimension separately.

Table 4.1: BHI values obtained by the ISA algorithm

	Average BHI	**Standard Deviation**
gene	0.930	0.005
sample	0.089	0.044

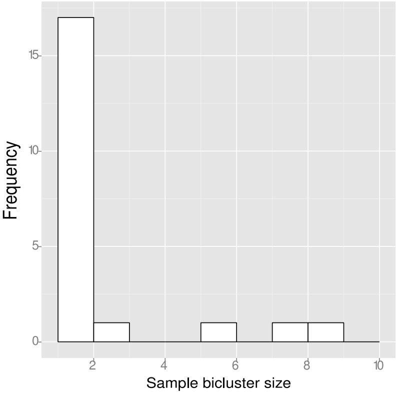

Figure 4.1: Size of biclusters found by ISA algorithm (sample dimension). The graph shows how many biclusters contain a given number of samples.

Thresholds have been set as suggested by Ihmels *et al.* [89] and by Csardi *et al.* [44]. Specifically, threshold t^G has been set to range from 1.8 to 4 (in steps of 0.1), and t^C has been fixed to 2. These values were used in both ISA and MD-ISA. In METAREF we set δ_r and δ_e to 2 and 0.5, respectively. These values allow the deletion of very far objects and the addition of very close objects. Thus, on the one hand, only completely unrelated objects are discarded, and on the other hand, the noise is kept under control and biclusters cannot grow arbitrarily. The value of N in MD-ISA has been set to 100. For each experiment 20 runs have been performed and results averaged.

Before delving into the evaluation of both METAREF and MD-ISA performances, let us describe the results obtained by ISA. As shown in Table 4.1, ISA obtains good results when the BHI is evaluated on gene clusters. This result is not surprising: ISA is considered one of the best approaches for identifying functional enriched biclusters [163]. Nevertheless, the BHI value dramatically decreases when sample clusters are examined. This is due to the size of discovered biclusters (see Figure 4.1). Indeed, ISA creates many biclusters grouping few samples (we will refer to this issue as "the sample bicluster size problem"). For instance, 17 out of the 21 identified biclusters group only two samples. Thus, ISA is able to identify biclusters of functionally enriched genes, but only in a small subset of samples.

Using the RGO for metadata-driven biclustering

MetaRef uses *a priori* knowledge to support the biclustering process in order to obtain biologically-relevant results. Thus, the choice of the metadata must be coherent with the aim of the analysis. Since our main goal is to show that biclusters represent transcriptional modules, we selected metadata information (and then features) that highlights cooperative genes and that emphasize common responses to experimental settings. To this purpose, we used the information codified into the *Restructured Gene Ontology* (RGO) (see Chapter 3). Specifically, we used three sets of features for describing each gene g (and each sample). Each set of features refers to one of the three RGO sub-ontologies: RGO *biological process* (BP), RGO *cellular component* (CC), and RGO *molecular function* (MF). These sets of features identify genes that participate to the same biological activities (sets for BP and MF) or take into consideration physical closeness of genes (the set for CC). We identify a feature for each RGO node n. Then, we redefine formulae (4.1) as:

$$M_{gn} = \begin{cases} \texttt{true} \text{ if } g \text{ is annotated over } n \\ \texttt{false} \text{ otherwise} \end{cases}$$

For instance, let us consider a toy ontology composed of five nodes $N_1, \ldots, N_5$, and two genes: g_1, annotated on node N_3 and N_5, and g_2, annotated on N_1 and N_5. By using this set of features genes are then represented by the following Boolean matrix M:

$$\begin{array}{c c} & \begin{array}{ccccc} N_1 & N_2 & N_3 & N_4 & N_5 \end{array} \\ \begin{array}{c} g_1 \\ g_2 \end{array} & \begin{bmatrix} \texttt{false} & \texttt{false} & \texttt{true} & \texttt{false} & \texttt{true} \\ \texttt{true} & \texttt{false} & \texttt{false} & \texttt{false} & \texttt{true} \end{bmatrix} \end{array}.$$

In this example $M_{11} = \texttt{false}$ states that g_1 is *not* annotated on N_1, and $M_{21} = \texttt{true}$ states that g_2 *is* annotated on N_1.

Table 4.2 (Columns 3 and 4) describes, for each set of features, its size and the percentage of genes having at least one annotation on genes/samples dimension. Let us recall that only genes/samples having at least one associated information are included in the distance matrices. Table 4.2 also reports (Columns 5 and 6) the obtained BHI values, showing that the metadata-driven procedure performed by Md-ISA highly increases BHI values with respect to those obtained by ISA, especially, but not limited to, on the sample dimension. Moreover, the size of biclusters on the sample dimension is well spread with respect to that obtained by the ISA algorithm (see Figure 4.2). It can be argued that this property makes them more meaningful from a biological point of view.

Table 4.2: BHI values obtained by the MD-ISA algorithm when metadata are extracted from the RGO.

		Metadata		**BHI**	
RGO sub-ontology		**Number of features**	**Annotated genes**	**Average**	**Standard Deviation**
BP	gene	15,589	83.8%	0.940	0.005
	sample		70.0%	0.838	0.090
CC	gene	2,918	83.8%	0.935	0.002
	sample		70.0%	0.673	0.060
MF	gene	9,149	83.8%	0.939	0.006
	sample		70.0%	0.849	0.072

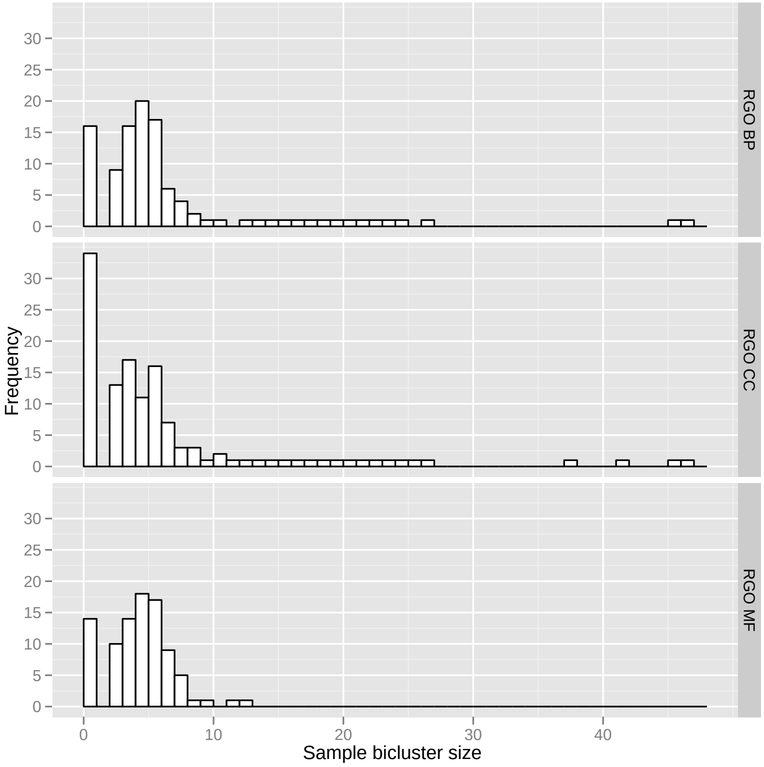

Figure 4.2: Size of biclusters found by MD-ISA algorithm (sample dimension). The figure shows, for each set of features, how many biclusters contain a given number of samples.

Table 4.3: BHI values obtained by applying MetaRef in different ways

		post-processing		**metadata-driven**	
RGO sub-ontology		**Average BHI**	**Standard Deviation**	**Average BHI**	**Standard Deviation**
BP	gene	0.939	0.004	0.940	0.005
	sample	0.020	0.009	0.838	0.090
CC	gene	0.938	0.003	0.935	0.002
	sample	0.017	0.009	0.673	0.060
MF	gene	0.938	0.002	0.939	0.006
	sample	0.018	0.009	0.849	0.072

Metadata-driven procedure vs metadata post-processing

It could be argued that the metadata-driven procedure would yield to comparable results when used as a mere data post-processing (instead of being incorporated into the algorithm as in our proposal). To rule out this hypothesis, we applied MetaRef to the biclusters outputted by ISA as a final refinement step. Table 4.3 shows the obtained BHI values, and the figures for Md-ISA that we already discussed. Results are grouped according to the metadata used. BHIs evaluated on gene clusters are larger than those obtained by the standard ISA in both experimental settings. However, this simple post-processing cannot avoid the "sample bicluster size problem". On the contrary, it seems that a post-processing driven approach leads to even worse performances. This is due to the small size of the original biclusters. Indeed, when the MetaRef procedure is applied starting from few genes the result is unpredictable and usually meaningless.

Transcriptional modules discovery

In addition to the evaluation performed by means of the BHI values, we performed an in-depth analysis to assess the biclusters quality from a biological point-of-view. To this purpose, we now focus on biclusters obtained by a run of Md-ISA using the RGO BP metadata. It results in 83 biclusters showing a BHI value of 0.935 for the gene dimension, and a BHI value of 0.831 for the sample dimension. These figures are close to the average values already presented. Within this result set, we focused on its smallest member (so to allow manual analysis) which groups 39 genes and 3 different experimental conditions (the mutant genes HST3, TUP1, and SSN6). In the rest of this section we will refer to the set of genes in the selected bicluster with the symbol I and to the set of mutants with the symbol J.

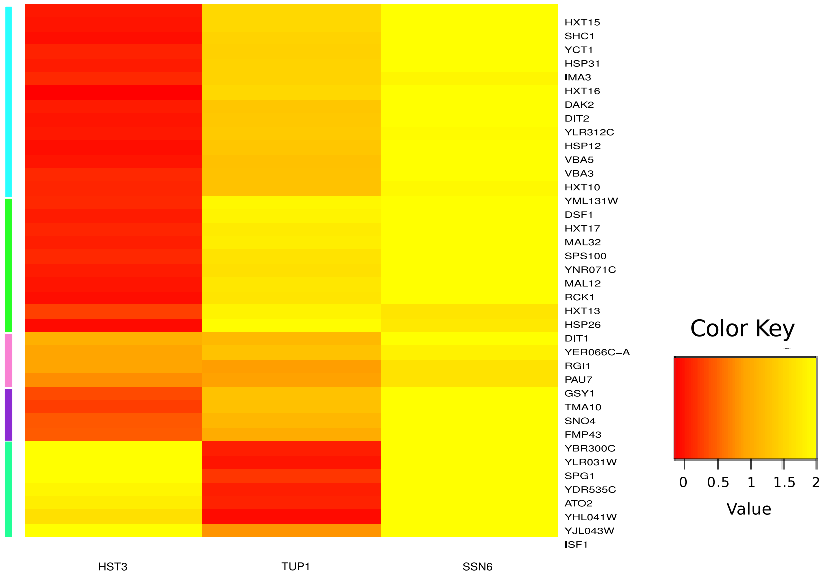

Figure 4.3: Bicluster heat map. The rows represent genes belonging to I, and the columns represent genes (mutant conditions) belonging to J. Each cell is filled based on the level of expression of that gene in that condition. Coloured segments on the left show a possible division on the gene dimension based on expression levels.

In order to asses the quality of the association between genes and samples, we analyzed biological functions that both groups perform. Mutant genes belonging to J are involved in metabolic activities. In particular, the SSN6-TUP1 protein complex is involved in the metabolism of galactose as well as of alternative carbon sources [1]. HST3 is involved in short-chain fatty acid metabolism [192] and several studies showed that calorie restrictions may interfere with HST3 activity [128]. A functional enrichment of genes in I performed by means of the FunSpec web application using the MIPS Functional Classification (see Section 3.3 for details) reveals that they are involved in *sugar transport* (p-value $8.37e^{-7}$), *metabolism* (p-value $3.40e^{-5}$), and in *metabolism of energy reserves* (p-value $3.06e^{-4}$). Summarizing, both I and J genes are involved in metabolic processes, thus confirming the faithfulness of the obtained bicluster.

In the following discussion we point out evidences suggesting that the genes in the selected bicluster forms a transcriptional module: i.e., they are co-expressed and bound by the same transcription factors. Let us then consider Figure 4.3 that shows the heat map of the selected bicluster. An important observation about the figure is that, in contrast to what one would expect for a transcriptional module, the genes do not appear to have a very

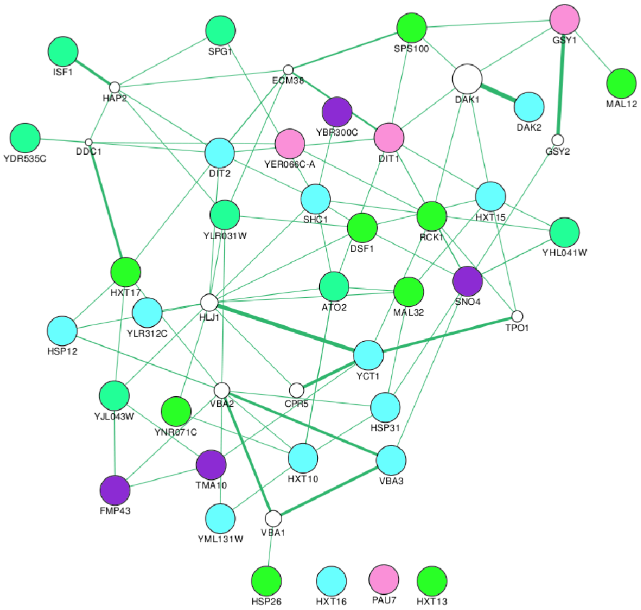

Figure 4.4: Interactions among genes biclustered together (gene dimension). Genes belonging to I are coloured according to clustering reported in Figure 4.3; specifically using the colours reported in colour bar on the left of the figure. Edges represent genetic interactions. The network is built and visualized using the GeneMANIA Cytoscape plugin.

close profile. In fact, as shown by the coloured segments on the left, one can easily recognize five different groups of co-expressed genes. However, the following argument shows that they participate to the same process nonetheless. This shows that the metadata driven approach allows the discovery of transcriptional modules that could not be recovered using expression profiles alone. The first evidence we present supporting our claim about genes in I is based on an analysis we conducted using the Yeast Promoter Atlas [31]. We used this tool to obtain the list of transcription factors that bind genes in I. Among the found transcription factors there is SPT15 (an essential general transcription factor involved in directing the transcription of genes [219]) that binds 11 of the genes in I, MSN2 (a transcription factor involved in response to several stresses including glucose starvation [66]) that binds 8 genes, and NRG1 (a transcriptional repressor that recruits the SSN6-TUP1 complex to promoters and mediates glucose repression [226]) that binds 5 genes. Interestingly, the genes bound by the these transcription factors are not localized in any one of the five co-expressed groups we mentioned before. Then, despite being in different co-expression groups, there exist transcriptional interactions among them. The second piece of evidence we provide confirms this finding. Figure 4.4 shows the genetic interactions among genes belonging to

I as created by the GeneMANIA Cytoscape plugin [140, 187]. GeneMANIA leverages data collected from several primary studies and from the BioGRID database and reports information about gene interactions. Specifically, the resulting network has an edge between two genes if they are functionally associated, i.e, "the effects of perturbing one gene were found to be modified by perturbations to a second gene" [140].

4.4 Conclusions

In this chapter we presented a new metadata-driven biclustering method. In detail, we described *i)* a new approach to extract metadata information from the RGO, *ii)* a general algorithm, namely MetaRef, for exploiting the extracted metadata, and *iii)* a modified version of ISA (Md-ISA), that implements our proposal. A test performed on a *S. cerevisiae* microarray compendium showed that Md-ISA obtains better results than the ISA algorithm.

Let us remark that the MetaRef definition is very general: it only requires the availability of some metadata, without focusing on a specific kind of knowledge. Hence, it can be used with any source of information. Moreover, the MetaRef algorithm can be exploited in other biclustering algorithms (e.g., Chang and Church biclustering approach). As a future work, we plan to test the performances of the presented metadata-driven technique when different prior information are available as well as when it is used to drive other biclustering algorithms.

Part III

Reverse Engineering

Chapter 5

Reverse Engineering and the DREAM Challenge

The reverse engineering of gene regulatory networks is a challenging task that requires the exploitation of mathematical and computational techniques. This task requires the inference of the interactions among the genes that compose a biological system. A widespread approach to extract such interactions is by analyzing microarray data.

This chapter presents a state-of-the-art of reverse engineering methods, and describes an approach integrating evidences derived from several types of microarray experiments. This approach has been applied to the DREAM5 microarray compendia, and despite its simplicity the approach fared fairly well when applied to real data sets.

5.1 Introduction

In the last decade, a lot of research has been carried out to address the problem of modeling of gene regulatory networks. Nonetheless, a gene regulatory network is still far from being trivial to define and to build.

According to a recent definition by Lefebvre *et al.*, a gene regulatory model is a "a computable representation based on empirical data that allows the inference of measurable macroscopic *dependent* variables as function of other *independent* variables" [120]. This definition includes several types of models, ranging from a highly-abstract view of the cellular system to very specific models. Inside each class of models, a large number of formalisms arose, and several attempts for their classification have been proposed [85, 114, 119, 131].

In this chapter we introduce a state-of-the-art of reverse engineering and a simple approach that makes use of statistical tools to extract information

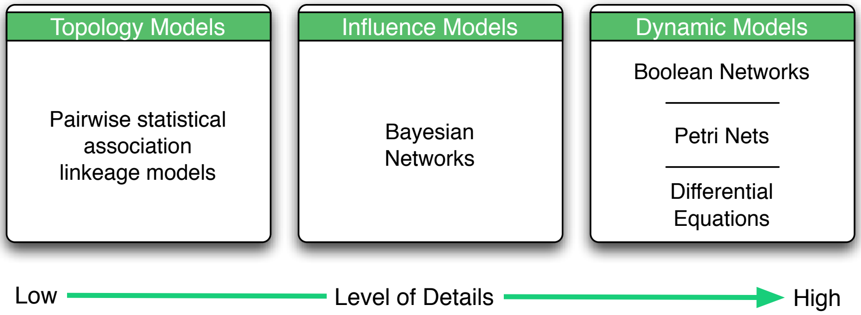

Figure 5.1: Overview of models and formalisms for the reverse engineering of gene regulatory networks. The most left formalism allows one to model the coarsest level of knowledge about cellular system. Moving to the right, the level of details increases.

about gene interactions from several types of microarray experiments. This approach has been used in the DREAM5 network inference challenge [53], obtaining appreciable results.

5.2 A state-of-the-art of reverse engineering

In the following, a subset of the state-of-the-art formalisms is presented. We classify these formalisms according to the class they belong to (and thus according to the level of detail they can manage). This classification, summarized in Figure 5.1, has been commonly used in literature [88, 120, 184].

Topology models are at the highest level of abstraction. They describe the network as a graph. Genes are represented by nodes, and relationships are represented by edges connecting nodes. From a semantic point of view, this model represents the knowledge about interactions among genes, with no additional (e.g., causal) information. Even though a topology model contains only a mere *qualitative* graphical representation of a cellular system, it is useful to point out relevant information. For instance, by executing algorithms for clique discovery, genes whose activities are involved in the same cellular behaviour or in the same phenotype (i.e., genes that belong to the same *functional module*) can be identified. Thanks to the high level of abstraction they enjoy, topological models are capable of representing genome-wide networks.

In this class we single out pairwise statistical association linkage models. They use a measure of similarity (distance) between gene expression levels to capture the statistical dependence between pairs of genes: two genes are predicted to interact if their distance value is above (below) a given threshold. Euclidean distance, Chebyshev distance, Minksowski distance and Angular separation are examples of such measures. The most widely used similarity measures are the Pearson Correlation Coefficient [56, 116, 198] and the mutual information measure [15, 28, 134].

Influence models add semantic information to topology models. Gene regulatory networks are still represented as graphs, but edges refer to a specific kind of influence: an edge may describe that a change in the gene expression level of one gene is due to the change of the gene expression level of another gene (*causal* influence) or it may represent the interaction between two different gene products (*physical* influence).

Bayesian Networks belong to this class of models. They are probabilistic graphical models that explicitly state the independencies in a multivariate joint probability distribution using a graph based formalism [151]. Gene networks that use this formalism model each gene by a random variable (i.e., by a node in the graph). A directed edge between two variables indicates an influence from the parent to the child. A Bayesian network associates a conditional probability $P(X|Pa_X)$ to each variable X, where Pa_X is the set of parents of X. The joint distribution over the set of all the system variables is represented as the product of all the associated conditional probabilities: $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|Pa_{X_i})$. Bayesian Networks provide a flexible and well-formalized language, allowing the introduction of prior biological knowledge [215]. Moreover, learning methods for inferring both structure and parameters, even in the case of incomplete data, are known [109]. These advantages lead to a wide use of Bayesian Networks in modeling gene regulatory networks [58, 57, 75, 166]. Unfortunately, limitations in the Bayesian formalism (e.g., they cannot cope with loopy graph structures) may hinder the correct modeling of real world dependencies. A particular class of Bayesian Networks, the Dynamic Bayesian Networks, can be used to allow feedback-loops or to deal with time series measurements [147, 158, 221].

Dynamic models represent both the *qualitative* and the *quantitative* information. A dynamic model allows the simulation of cellular system behaviours. In these models the identification of both the structure and the parameters is not trivial: it requires prior knowledge about reaction mechanisms as well as experimental data to estimate the kinetic parameters. Thus, its complexity allows to handle only small-size networks.

In this class are included Boolean networks, Petri nets, and differential equations based models. Boolean network models are discrete dynamic models, where each gene is codified by a Boolean variable representing active and inactive states [102]. If a gene is expressed the corresponding variable assumes the value 1, otherwise it assumes the value 0. At each time step, the value of a single variable depends on the values of its regulators at the previous time step. The Boolean network structure could be known *a priori* or could be inferred from microarray experiments [54, 122]. Despite using a very simple model, Boolean Networks yield accurate predictions of biological systems, as proved by studies about the development module in *Drosophila* and about the *Saccharomyces cerevisiae* cell-cycle dynamics [3, 121].

Petri nets are bipartite graphs represented as collections of *places* and *transitions* connected by directed arcs. The *marking* of the network is an assignment of tokens to places, and the marking of a place represents its state value. System dynamics are given by the evolution of the marking, that changes every time that a transition *fires* [159]. In the molecular biology context, tokens describe the amount of biochemical compounds (e.g., proteins, genes, and chemical complexes), whereas rates of transitions model the occurrence speed of their reactions. Petri nets are particularly suitable to represent reactions where compounds are consumed (such as in metabolic pathways [78, 167]), but they have also been used to represent gene regulatory networks [32, 33, 203].

Differential equations describe the change in the expression level of a gene as a function of other gene expression levels over time. Even though the regulatory processes are characterized by non-linear dynamics, few approaches take into account non-linear functions [48, 212], instead most often linear models are adopted as a viable approximation [12, 50, 125].

5.3 Methods

Here we introduce a Naive Bayes based approach for reverse engineering. It allows the integration of multiple sources of information as well as the easy elicitation of background knowledge from domain experts. We use the DREAM5 microarray compendia as a case study (to be described in Section 5.4). Each compendium is composed by microarray experiments that correspond to several experimental designs. Our goal is to obtain a list of edges linking transcription factors to their targets sorted by a plausibility measure.

We start by processing each experiment in order to identify the sets of differentially expressed genes. Different statistical techniques are used according to the experimental design at hand. By assuming that differentially expressed genes are likely to interact with one another, we set edges among them as candidates for being included in the final network. The outcome of this step is a set of candidate edges for each experimental setting and our goal is to compute the probability that a candidate edge belongs to the network given its experimental evidence. Our approach is based on the observation that relationships derived from different experimental designs may have different reliabilities (e.g., knock-out and over-expression experiments offer more reliable evidences than wild-type or perturbation experiments [86, 111]).

Let us consider a sampling experiment where possible edges are drawn at random and denote by X a stochastic variable that assumes value 1 when the drawn edge belongs to the network and 0 otherwise. Let us also define $Y_1 \dots Y_m$ to be the values of statistics assessing the given edge (e.g., the p-values provided by the maSigPro algorithm during the analysis of a time series experiment). We would like to compute the probability $P(X = 1|Y_1, \dots, Y_m)$, that is probability that an edges exists given the output of the statical analyses. By Bayes's theorem, the probability $P(X = 1|Y_1, \dots, Y_m)$ could be written as:

$$\begin{aligned}P(X=1|Y_1\dots Y_m)=&\frac{P(Y_1,\dots,Y_m|X=1)P(X=1)}{\sum_{x\in\{0,1\}}P(Y_1,\dots,Y_m|X=x)P(X=x)}\\=&\frac{\prod_{i=1,\dots,m}P(Y_i|X=1)P(X=1)}{\sum_{x\in\{0,1\}}\prod_{i=1,\dots,m}P(Y_i|X=x)P(X=x)}\end{aligned}\tag{5.1}$$

where equality holds by assuming Y_i and Y_j to be independent given X for each i, j $(i \neq j)$, and term $P(X = 1)$ is the *a priori* probability of observing an edge belonging to the network.

To compute the sought probabilities in Formulae (5.1), we need to specify the distributions of X and of each Y_i given X. To compute the former we exploit a common accepted assumption: that biological networks have a scale-free topology [2]. In a scale-free network, the probability that a randomly chosen node has exactly k edges is $P(k) = k^{-\gamma}$, with $\gamma \in [2, 3]$. Here, as suggested by Barabàsi and Albert [14], we set $\gamma = 3$. It follows that the number of edges in a scale-free network with N genes can be computed as:

$$e(N)=\sum_{k=1}^{N}N\times P(k)\times k=N\sum_{k=1}^{N}\frac{1}{k^2}.$$

By approximating the quantity $\sum_{k=1}^{N}\frac{1}{k^2}$ with $\frac{\pi^2}{6}$ (its limit for $N \to \infty$), it follows that in a network of size N the probability that 'picking at random two genes they are connected' is $e(N)/N^2 = \frac{\pi^2}{6N}$. Then we set $P(X = 1) = \frac{\pi^2}{6N}$ and $P(X = 0) = 1 - \frac{\pi^2}{6N}$.

To compute formula (5.1) the distributions of the Y_i given X still need to be specified. Since no data is available to estimate them, it is necessary to elicit them from a domain expert on the basis of the confidence she/he has in the tools that generated the Y_i values. Statistics she/he is less confident about (e.g., the Pearson Correlation Coefficient evaluated in a wild-type experiments) are associated with uniform-like distributions, other ones (e.g., limma evaluated in a knock-out experiment) are associated with distributions more peaked at one of the extremes. Figure 5.2 shows the distributions one may choose for an analysis based on the Pearson Correlation Coefficient and one based on the limma package. The complete system architecture is shown in Figure 5.3.

The presented framework allows for very different kind of knowledge to be merged in a straightforward way. This flexibility is of paramount importance nowadays. The huge amount of *omics* data available (e.g., Next Generation Sequencing data [136]) and the trustable set of literature-derived edges make for perfect examples of very different and informative data that are valuable to merge. Augmenting an existing predictor with data from an additional source of information, such as Next Generation Sequencing data, consists in selecting a proper analysis tool and in specifying the distribution that model the analyst's confidence in the selected tool.

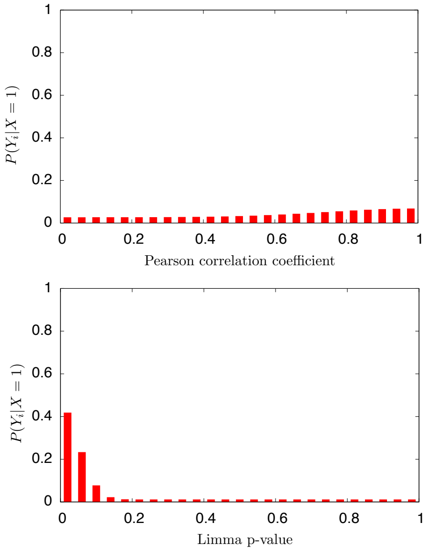

Figure 5.2: $P(Y_i|X = 1)$ distribution for PCC and limma p-value statistics used in the Naive Bayes approach.

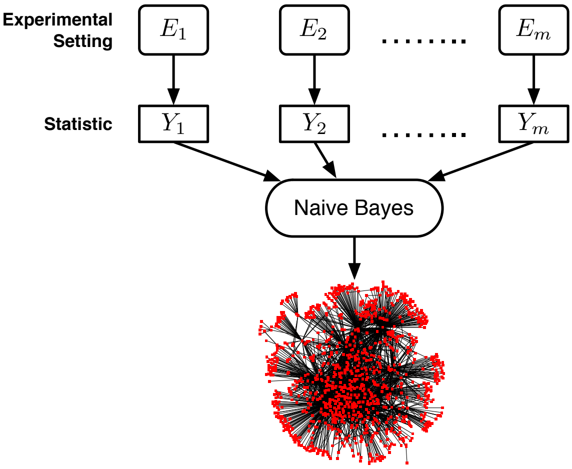

Figure 5.3: System architecture. From each experimental settings, evidences are evaluated by means of a statistical tool. Then, the network is build by using a Naive Bayes approach.

5.4 Results

As mentioned, the presented approach competed in 2010 to the *Dialogue for Reverse Engineering Assessments and Methods* contest (DREAM5) [53]. Each year, the DREAM organizers propose several challenges about the inference of biological networks and/or the prediction of how they are influenced by perturbations. Each method submitted to a challenge is evaluated using rigorous metrics and compared with others. Thus, the challenge allows a fair assessment of the strengths and weaknesses of the proposed methods and an objective judgement about the reliability of submitted models. Besides, experimental and synthetic benchmark data sets are provided to the community.

Four different challenges have been proposed in 2010. The challenge number four (the *network inference challenge*) deals with the inference of genome-scale gene regulatory networks over several organisms. Among the four provided data sets, one derives from *in silico* experiments, the others correspond to the three real organisms. Each data set is composed by three files containing: a list of putative transcription factors, the gene expression data, and meta-information describing the experimental design of each microarray experiment. Gene expression data were pre-processed by the DREAM organizers in order to allow direct comparison of all genes over all conditions. Four broad classes of experimental settings are used in the experiments: gene deletion, gene over-expression, time series, and perturbations; some experiments are made by using a combination of different experimental settings. In order to allow a fair comparison of the methods, each data set is made available in an anonymized format: genes and experimental settings are provided with meaningless identifiers so that the only meaningful information in the data sets is the value of the measured gene expression levels. A file containing the gold standard network was revealed after the contest deadline.

The challenge requests to submit a list of (at most) 100,000 edges linking transcription factors to their targets (either genes or transcription factors). The list needs to be sorted according to a plausibility measure.

We processed the microarray experiments according to to their experimental settings using appropriately chosen statistical tools. Wild-type experiments were analyzed using the Pearson Correlation Coefficient (PCC) [153]. We added to the candidate list an edge linking a transcription factor to a gene if the absolute value of their correlation is greater than 0.5. Knockout/over-expression experiments went through a limma analysis. We added to the candidate list an edge among the deleted/over-expressed transcription factor and all the differentially expressed genes having a p-value < 0.05. In addition, a z-score of the difference between wild-type and treatment experiments was used when the limma analysis could not find any differentially expressed gene. An edge among the deleted/over-expressed transcription factor and all the genes is added to the candidate list. Perturbations were analyzed with the limma algorithm, and the PCC was used to assess the dependence among the identified transcription factors and all the other differentially expressed genes. An edge was added to the candidate list if it insists on a pair $\langle$transcription factor, differentially expressed gene$\rangle$ having p-value < 0.05 and $|\text{PCC}| > 0.5$. Time series experiments were processed with the maSigPro algorithm in order to detect the differentially expressed genes that are clustered together (again only differentially expressed genes having a p-value < 0.05 were considered). Within each cluster returned by maSigPro, we evaluated the PCC among all transcription factors and genes profiles, and we added an edge to the candidate lists if pairs of transcription factors/differentially expressed genes have $|\text{PCC}| > 0.5$. We generated our results by enumerating all possible pairings of transcription factors and genes and using the formula (5.1) to assess the edges plausibility. The sorted edges list was then truncated to size $100,000$.

Submitted results have been compared, by the DREAM organizers, with a gold standard. Of the four networks under investigation: network 1 corresponds to a synthetic data set; network 2 corresponds to the *Staphylococcus aureus*, a bacterium for which a gold standard is not available (consequently, result sets targeting this network has not been evaluated); networks 3 and 4 correspond to well-known organisms, namely *Escherichia coli* and *Saccharomyces cerevisiae.*

The *Area Under the Receiver Operating Characteristic* curve (AUROC) and the *Area Under Precision/Recall* curve (AUPR) evaluation measures have been used to compare the submitted networks. The two curves are built by varying the size of edges list and measuring how the re-dimensioned lists performed. The AUROC is computed using the ratio between the true positive rate, and the false positive rate (see Section 6.3 for details). The AUPR is calculated using the ratio between *precision* (i.e., $\frac{\texttt{TP}}{\texttt{TP}+\texttt{FP}}$) and *recall* (i.e., $\frac{\texttt{TP}}{\texttt{TP}+\texttt{FN}}$).

The method overall score is as follows. A p-value is calculated for each AUROC and AUPR scores. Then, two statistics, the AUROC p-value (p_{AUROC}) and the AUPR p-value (p_{AUPR}), are calculated by taking the geometric mean of the p-value scores on each network. Finally, the method overall score is calculated as $-\frac{1}{2}\log_{10}(p_{\text{AUROC}} \times p_{\text{AUPR}})$ (additional details can be found in [164, 197]).

Table 5.1 reports the method rankings as published by the DREAM organizers. Results for the presented method are typeset in a boldface font. Our methodology obtains middle-ranking performances. Upon closer inspection, however, it is apparent that the proposed approach performs badly on the task of reconstructing the synthetic network. Table 5.2 shows the resulting ranking along with the re-evaluation of the measures on networks 3 and 4 only. In this is new evaluation our method ranks in the top positions.

5.5 Conclusions

In this chapter we presented a state-of-the-art for reverse engineering and a Naive Bayes approach. The presented approach has two advantages: *i)* it can be applied even when no information is available on the network structure, and *ii)* it is easy to extend with new evidence. Needless to say, this method

Table 5.1: Results for the DREAM Network Inference Challenge. Columns 6-11 report p-values for the AUPR and AUROC measures on networks 1, 3, and 4. Columns 4 and 5 report their average scores p_{AUPR} and p_{AUROC}. Column 3 reports the overall methods score. All values correspond to those published by the DREAM organizers [53].

	Team	Overall	p_{AUPR}	p_{AUROC}	$p_{\text{AUPR}}1$	$p_{\text{AUPR}}3$	$p_{\text{AUPR}}4$	$p_{\text{AUROC}}1$	$p_{\text{AUROC}}3$	$p_{\text{AUROC}}4$
1	415	40.279	$1.97e^{-41}$	$1.83e^{-39}$	$1.60e^{-104}$	$5.15e^{-20}$	$1.58e^{-01}$	$3.06e^{-106}$	$5.00e^{-11}$	$1.06e^{-02}$
2	543	34.023	$8.09e^{-30}$	$1.37e^{-37}$	$8.17e^{-053}$	$1.07e^{-39}$	$2.15e^{-02}$	$1.34e^{-056}$	$1.64e^{-53}$	$1.77e^{-03}$
3	776	31.099	$2.33e^{-41}$	$6.78e^{-20}$	$7.20e^{-118}$	$4.34e^{-07}$	$2.53e^{-01}$	$3.48e^{-059}$	$3.42e^{-03}$	$2.71e^{-02}$
4	862	28.747	$4.89e^{-44}$	$6.40e^{-12}$	$8.58e^{-135}$	$9.99e^{-01}$	$1.00e^{+00}$	$3.82e^{-039}$	$1.00e^{+00}$	$1.00e^{+00}$
5	548	22.711	$6.24e^{-36}$	$4.24e^{-08}$	$8.41e^{-084}$	$4.76e^{-12}$	$1.03e^{-16}$	$4.09e^{-017}$	$1.17e^{-09}$	$2.76e^{-01}$
6	870	21.398	$6.89e^{-32}$	$9.06e^{-09}$	$2.83e^{-091}$	$1.73e^{-08}$	$6.25e^{-01}$	$2.46e^{-024}$	$4.98e^{-06}$	$1.09e^{-01}$
7	868	20.694	$9.23e^{-31}$	$2.64e^{-09}$	$1.42e^{-092}$	$5.11e^{-04}$	$1.75e^{-01}$	$6.97e^{-022}$	$1.86e^{-04}$	$4.19e^{-04}$
8	842	16.686	$3.45e^{-14}$	$6.81e^{-18}$	$5.98e^{-035}$	$4.47e^{-10}$	$9.10e^{-01}$	$5.13e^{-057}$	$7.21e^{-01}$	$8.56e^{-01}$
9	861	13.397	$6.04e^{-13}$	$1.03e^{-13}$	$1.35e^{-039}$	$2.18e^{-02}$	$1.55e^{-01}$	$6.95e^{-033}$	$3.26e^{-03}$	$4.04e^{-05}$
10	395	10.586	$3.10e^{-10}$	$4.80e^{-10}$	$4.20e^{-008}$	$1.53e^{-13}$	$5.25e^{-12}$	$3.42e^{-002}$	$5.56e^{-05}$	$4.78e^{-27}$
11	799	8.887	$1.19e^{-03}$	$5.00e^{-14}$	$1.58e^{-008}$	$3.85e^{-02}$	$9.78e^{-01}$	$8.18e^{-045}$	$9.94e^{-01}$	$9.83e^{-01}$
12	875	8.800	$9.64e^{-03}$	$4.13e^{-13}$	$3.08e^{-010}$	$3.64e^{-03}$	$1.00e^{+00}$	$1.50e^{-041}$	$9.44e^{-01}$	$1.00e^{+00}$
13	823	8.126	$4.29e^{-05}$	$4.16e^{-10}$	$3.63e^{-013}$	$3.53e^{-05}$	$9.89e^{-01}$	$1.59e^{-032}$	$9.60e^{-01}$	$9.07e^{-01}$
14	48	6.420	$1.52e^{-08}$	$4.55e^{-04}$	$3.80e^{-007}$	$3.11e^{-15}$	$2.39e^{-04}$	$4.63e^{-002}$	$7.08e^{-09}$	$3.24e^{-05}$
15	**702**	**6.332**	**$2.17e^{-08}$**	**$2.12e^{-04}$**	**$7.98e^{-001}$**	**$9.76e^{-06}$**	**$1.25e^{-20}$**	**$8.82e^{-001}$**	**$4.32e^{-13}$**	**$2.74e^{-01}$**
16	772	6.026	$8.39e^{-08}$	$1.34e^{-03}$	$1.42e^{-011}$	$2.26e^{-16}$	$5.24e^{-01}$	$8.78e^{-001}$	$2.43e^{-09}$	$1.93e^{-01}$
17	864	4.973	$8.85e^{-09}$	$1.00e^{+00}$	$1.65e^{-030}$	$8.77e^{-01}$	$1.00e^{+00}$	$9.99e^{-001}$	$1.00e^{+00}$	$1.00e^{+00}$
18	705	2.484	$1.36e^{-02}$	$6.84e^{-02}$	$1.00e^{+000}$	$7.17e^{-01}$	$5.58e^{-07}$	$2.34e^{-005}$	$8.97e^{-04}$	$1.49e^{-01}$
19	504	2.266	$3.40e^{-04}$	$1.00e^{+00}$	$3.39e^{-014}$	$7.53e^{-01}$	$1.00e^{+00}$	$9.99e^{-001}$	$1.00e^{+00}$	$1.00e^{+00}$
20	281	1.897	$1.16e^{+00}$	$5.33e^{-03}$	$1.00e^{+000}$	$9.99e^{-01}$	$6.32e^{-01}$	$9.99e^{-001}$	$8.54e^{-12}$	$7.74e^{-01}$
21	829	1.475	$2.36e^{-02}$	$3.79e^{+00}$	$1.00e^{+000}$	$1.36e^{-07}$	$5.63e^{-01}$	$9.99e^{-001}$	$4.98e^{-01}$	$3.69e^{-02}$
22	802	0.997	$1.00e^{+00}$	$9.89e^{-01}$	$1.00e^{+000}$	$9.99e^{-01}$	$9.99e^{-01}$	$1.12e^{-006}$	$9.98e^{-01}$	$9.28e^{-01}$
23	756	0.331	$1.49e^{+00}$	$3.08e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$3.04e^{-01}$	$8.21e^{-001}$	$1.00e^{+00}$	$4.17e^{-02}$
24	736	0.257	$1.10e^{+00}$	$2.97e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$7.58e^{-01}$	$5.69e^{-001}$	$1.00e^{+00}$	$6.71e^{-02}$
25	854	0.256	$1.10e^{+00}$	$2.96e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$7.58e^{-01}$	$5.71e^{-001}$	$1.00e^{+00}$	$6.71e^{-02}$
26	638	0.144	$1.78e^{+00}$	$1.09e^{+00}$	$1.79e^{-001}$	$9.99e^{-01}$	$9.93e^{-01}$	$9.52e^{-001}$	$1.00e^{+00}$	$8.08e^{-01}$
27	784	0.035	$1.01e^{+00}$	$1.17e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$9.83e^{-01}$	$9.99e^{-001}$	$1.00e^{+00}$	$6.26e^{-01}$
28	787	0.000	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$1.00e^{+00}$	$9.99e^{-001}$	$1.00e^{+00}$	$1.00e^{+00}$
29	821	0.000	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+000}$	$9.99e^{-01}$	$1.00e^{+00}$	$9.99e^{-001}$	$1.00e^{+00}$	$1.00e^{+00}$

Table 5.2: Results for the DREAM Network Inference Challenge. Columns 6-9 report p-values for the AUPR and AUROC measures on networks 3 and 4. Columns 4 and 5 report their average scores p_{AUPR} and p_{AUROC}. Column 3 reports the overall methods score.

	Team	Overall	p_{AUPR}	p_{AUROC}	$p_{\text{AUPR}}3$	$p_{\text{AUPR}}4$	$p_{\text{AUROC}}3$	$p_{\text{AUROC}}4$
1	543	24.0435	$4.80e^{-21}$	$1.70e^{-28}$	$1.07e^{-39}$	$2.15e^{-02}$	$1.64e^{-53}$	$1.77e^{-03}$
2	395	13.6677	$8.96e^{-13}$	$5.15e^{-16}$	$1.53e^{-13}$	$5.25e^{-12}$	$5.56e^{-05}$	$4.78e^{-27}$
3	**702**	**9.4601**	$\mathbf{3.49e^{-13}}$	$\mathbf{3.44e^{-07}}$	$\mathbf{9.76e^{-06}}$	$\mathbf{1.25e^{-20}}$	$\mathbf{4.32e^{-13}}$	$\mathbf{2.74e^{-01}}$
4	548	9.2009	$2.21e^{-14}$	$1.79e^{-05}$	$4.76e^{-12}$	$1.03e^{-16}$	$1.17e^{-09}$	$2.76e^{-01}$
5	415	8.0907	$9.03e^{-11}$	$7.30e^{-07}$	$5.15e^{-20}$	$1.58e^{-01}$	$5.00e^{-11}$	$1.06e^{-02}$
6	48	7.6919	$8.62e^{-10}$	$4.79e^{-07}$	$3.11e^{-15}$	$2.39e^{-04}$	$7.08e^{-09}$	$3.24e^{-05}$
7	772	6.3140	$1.09e^{-08}$	$2.16e^{-05}$	$2.26e^{-16}$	$5.24e^{-01}$	$2.43e^{-09}$	$1.93e^{-01}$
8	870	3.5576	$1.04e^{-04}$	$7.38e^{-04}$	$1.73e^{-08}$	$6.25e^{-01}$	$4.98e^{-06}$	$1.09e^{-01}$
9	281	2.8448	$7.95e^{-01}$	$2.57e^{-06}$	$9.99e^{-01}$	$6.32e^{-01}$	$8.54e^{-12}$	$7.74e^{-01}$
10	868	2.7896	$9.45e^{-03}$	$2.79e^{-04}$	$5.11e^{-04}$	$1.75e^{-01}$	$1.86e^{-04}$	$4.19e^{-04}$
11	776	2.7481	$3.31e^{-04}$	$9.63e^{-03}$	$4.34e^{-07}$	$2.53e^{-01}$	$3.42e^{-03}$	$2.71e^{-02}$
12	705	2.5684	$6.32e^{-04}$	$1.15e^{-02}$	$7.17e^{-01}$	$5.58e^{-07}$	$8.97e^{-04}$	$1.49e^{-01}$
13	842	2.4002	$2.02e^{-05}$	$7.85e^{-01}$	$4.47e^{-10}$	$9.10e^{-01}$	$7.21e^{-01}$	$8.56e^{-01}$
14	861	2.3381	$5.81e^{-02}$	$3.63e^{-04}$	$2.18e^{-02}$	$1.55e^{-01}$	$3.26e^{-03}$	$4.04e^{-05}$
15	829	2.2131	$2.77e^{-04}$	$1.36e^{-01}$	$1.36e^{-07}$	$5.63e^{-01}$	$4.98e^{-01}$	$3.69e^{-02}$
16	823	1.1292	$5.91e^{-03}$	$9.33e^{-01}$	$3.53e^{-05}$	$9.89e^{-01}$	$9.60e^{-01}$	$9.07e^{-01}$
17	875	0.6161	$6.03e^{-02}$	$9.71e^{-01}$	$3.64e^{-03}$	$1.00e^{+00}$	$9.44e^{-01}$	$1.00e^{+00}$
18	756	0.4746	$5.51e^{-01}$	$2.04e^{-01}$	$9.99e^{-01}$	$3.04e^{-01}$	$1.00e^{+00}$	$4.17e^{-02}$
19	799	0.3587	$1.94e^{-01}$	$9.88e^{-01}$	$3.85e^{-02}$	$9.78e^{-01}$	$9.94e^{-01}$	$9.83e^{-01}$
20	736	0.3236	$8.70e^{-01}$	$2.59e^{-01}$	$9.99e^{-01}$	$7.58e^{-01}$	$1.00e^{+00}$	$6.71e^{-02}$
21	854	0.3236	$8.70e^{-01}$	$2.59e^{-01}$	$9.99e^{-01}$	$7.58e^{-01}$	$1.00e^{+00}$	$6.71e^{-02}$
22	784	0.0528	$9.91e^{-01}$	$7.91e^{-01}$	$9.99e^{-01}$	$9.83e^{-01}$	$1.00e^{+00}$	$6.26e^{-01}$
23	504	0.0309	$8.68e^{-01}$	$1.00e^{+00}$	$7.53e^{-01}$	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+00}$
24	638	0.0240	$9.96e^{-01}$	$8.99e^{-01}$	$9.99e^{-01}$	$9.93e^{-01}$	$1.00e^{+00}$	$8.08e^{-01}$
25	864	0.0143	$9.36e^{-01}$	$1.00e^{+00}$	$8.77e^{-01}$	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+00}$
26	802	0.0085	$9.99e^{-01}$	$9.63e^{-01}$	$9.99e^{-01}$	$9.99e^{-01}$	$9.98e^{-01}$	$9.28e^{-01}$
27	862	0.0002	$9.99e^{-01}$	$1.00e^{+00}$	$9.99e^{-01}$	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+00}$
28	787	0.0002	$9.99e^{-01}$	$1.00e^{+00}$	$9.99e^{-01}$	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+00}$
29	821	0.0002	$9.99e^{-01}$	$1.00e^{+00}$	$9.99e^{-01}$	$1.00e^{+00}$	$1.00e^{+00}$	$1.00e^{+00}$

is one of the simplest approaches that can be developed for a problem of this complexity. Indeed, simplicity has been one of the goals we strove to attain in its design. This choice has been also motivated by the fact that when real networks are to be analyzed, data scarcity and its quality demand for classifiers built using a small number of well understood parameters. Also, past DREAM conferences emphasized that simpler methods tend to perform as well as others [164, 197].

Our method performs remarkably better in the case of real networks than with synthetic ones: it is among the top performers (it ranks third out of 29) when the synthetic data set is hold out from the evaluation (it ranks 15th otherwise). A number of interesting questions could be raised by this observation: what is in synthetic data set that set them apart from natural ones? Should one strive to optimize new algorithms more aggressively on natural data sets? Could the culprit be found in the quality of real data, so that most of these methods will perform much better when this quality increases? We believe that the answers to these questions may be important to better understand current tools and to develop new ones.

As a future work, we plan to leverage the available gold standards using a learning approach. In this way, the importance of each experimental design as well as of each additional data source for setting correct edges can be captured. The aim of the resulting framework is to assess which experiments to execute in order to obtain the better performances in the reverse engineering task.

Chapter 6

Reverse Engineering through Granger Causality

Time series data contains a lot of information about causal relationships whose exploitation may allow a deep comprehension of biological mechanisms. Also, the progress of high-throughput technologies provides insights on different aspects of the regulatory process. Thus, the use of time series information derived from multiple high-throughput data sources is likely to foster the development of techniques able to improve the understanding of the regulatory machinery.

This chapter presents a method for the reverse engineering of gene regulatory networks that uses a popular econometrics statistical hypothesis test, namely the Granger Causality. This approach integrates two kinds of temporal data: transcription and protein profiles. When applied to *Saccharomyces cerevisiae* under oxidative stress the proposed approach gives promising results.

6.1 Introduction

Gene expression is a temporal process. Usually, the response machinery starts by activating some transcription factors that control the gene expression resulting in a regulative cascade. The discovery of these temporal events is made feasible by time series microarray experiments, that measure transcript expression profiles over several time points at the genome scale.

Several formalisms have been used to model time relationships; dynamic Bayesian networks and ordinary differential equations are the most popular (see Section 5.1 for a brief review). Recently, new techniques from other fields of research have been applied to the reverse engineering task. For instance, the Granger Causality test [67] has been recently used for modeling gene regulatory networks [126, 227, 230] and a comparative study shows that it outperforms dynamic Bayesian networks on large data sets [229]. Roughly speaking, a random variable Y is said to *Granger cause* a variable X if knowledge about the past behaviour of Y improves the predictions of X. In fact, although Granger Causality actually measures correlations, it is a widespread belief that it reflects causality among variables. Granger Causality is naturally able to solve two important problems in the reverse engineering task: topology reconstruction and directionality identification: the Granger test infers topology, and time indicates directionality [206].

A limitation of the majority of the reverse engineering methods is that they infer gene regulatory networks from transcript expression profiles alone. Thus, they are likely to miss important pieces of information as well as to identify misassociations. For instance, these methods assume that the proteins abundance is proportional to mRNA levels. In Section 2.1 we mentioned that this is not always the case: post-transcription as well as post-translation modifications may occur. Moreover, when transcript profile alone are considered, the Granger Causality test does not allow the presence of self-loops, that are instead an important facet of biological networks. Recently, however, the progress of high-throughput technologies made available a lot of quality information about different aspects of cellular systems and about roles of cellular components. For instance, several techniques are available to measure protein abundance in both steady state and time series [191, 207], and it seems likely that such data will be increasingly available in the future [154]. Thus, the idea of exploiting such information by integrating multiple data sources is becoming increasingly appealing and several works have been proposed that cope with data of protein-protein interactions and of transcription regulation as well as database information [135, 214, 220]. However, a limitation of these approaches is that they do require that a network for the problem at hand is already available: they cannot deal with raw data alone. This is a big limitation since in most cases building the network is one of the research goals.

In this chapter we propose a Granger Causality test that makes use of mRNA and protein time series data. Specifically, we propose to infer a gene regulatory network by evaluating a Granger Causality test among protein and mRNA profiles. The idea behind this approach is the exploitation of the process that drives real cellular systems: changes in transcript expression levels are caused by changes in protein concentrations. By considering influences between mRNA and protein we also overcome some limitations presented by other reverse engineering approaches, such as post-transcription and post-translation modifications. The presented approach is still in a preliminary state. Nonetheless, the results we obtained are promising, and further investigation may lead to more appreciable outcomes.

6.2 Methods

Before explaining our reverse engineering approach, we describe the shortcomings of the data sets that are currently available for this kind of task, and how these data sets are used in the presented method. Usually, both mRNA and protein profiles are measured at few and specific time points, that are thickly sampled shortly afterwards a cell treatment; few values are instead measured long afterwards. In this way, fast changes that do not happen at the beginning of the experiments can be easily missed. However, it is a widely accepted assumption that unexpected dynamics are unlikely to occur a long time after the treatment. Also, the number of measured time points is greatly lower than the number of interactions to model, thus leading to an underdetermination problem. A widespread approach to overcome low-frequency sampling is to use interpolation, random effect regression or smoothing [13]. Following this idea, we fit a cubic smoothing spline to the experimental data. Hereafter, any figure mentioned will refer to the results of spline interpolation instead of raw experimental values, as exemplified by Figure 6.1.

The idea behind the presented approach derives from the assumption that a transcript expression level at a certain time can be the result of a protein expression level at previous time points. That is, we assume that causal relationships among proteins and mRNAs exist if the protein activation occurs before the mRNA activation, while no interaction exists otherwise. We evaluate causal relationships by means of a bivariate Granger Causality test.

Let us consider a set of random variables $\mathbf{X} = \{X_1, \ldots, X_n\}$ describing mRNA profiles, with X_i being the expression profile of the ith mRNA; and a set of random variables $\mathbf{Y} = \{Y_1, \ldots, Y_m\}$ describing protein profiles, with Y_j being the expression profile of the jth protein. Let us denote by $X_i^{(l)}$ and by $Y_j^{(l)}$ the l past values of X_i and Y_j, respectively. Let us then consider the regulative process (X_i, Y_j): we aim at evaluating whether $Y_j^{(l)}$ is useful for predicting X_i, i.e., we seek to determine whether the protein Y_i influences the transcript X_i. The null hypothesis to test is that Y_j do not help in predicting the value of X_i. Let us define as $e(X_i|X_i^{(l)})$ the error made by the autoregression model of X_i given its l past values, and as $e(X_i|X_i^{(l)}, Y_j^{(l)})$ the error made by the regression model of X_i given its l past values and the l past values of Y_j. If we can determine with confidence that $e(X_i|X_i^{(l)}, Y_j^{(l)}) < e(X_i|X_i^{(l)})$, we conclude that Y_j *Granger cause* X_i. F-statistics and p-values are used in order to reject the null hypothesis and to evaluate its confidence.

The bivariate Granger Causality test is evaluated for each regulative process $(X_i, Y_j) \in \mathbf{X} \times \mathbf{Y}$. An edge between X_i and Y_j is set if the null hypothesis is rejected. Finally, edges are sorted according to increasing p-values and decreasing F-measures.

The choice of the lag l is not straightforward. Previous Granger causality approaches assume $l = 1$, thus ignoring the possibility of slow influences. Moreover, it is widespread accepted that regulatory processes do not show

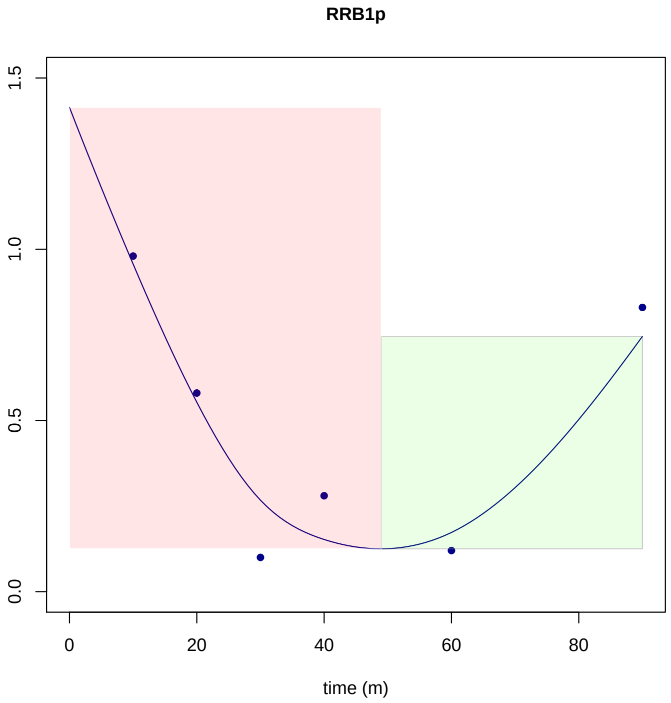

Figure 6.1: RRB1p protein profile under oxidative stress. Points represent experimental measurements. Line represents the interpolating cubic spline. The red box highlights the phase of response to stress. The green box highlights the recovery phase.

homogeneous lags, and to set l to a fixed value can be a poor approximation. For instance, Lozano *et al.* showed that considering time lags larger than 1 improves the model accuracy [126]. Here, we propose to evaluate the lag between Y_j and X_i as the difference between the *start time* of Y_j and the *start time* of X_i. In order to identify the start time, we divide each profile into intervals such that the expression profile is monotonic in each of them. The interval where mRNA/protein fold-change is the largest, or (when multiple dynamic intervals were observed) the first interval exceeding a threshold change is said to be the *primary interval*. We set the start time T_s to the time point where the primary interval begins. For instance, Figure 6.1 shows the profile of a protein regulating the ribosome biogenesis, namely RRB1p, when exposed to oxidative stress. Its profile goes through two main phases: as a response to the stress (red box) the protein is down-regulated; when recovering (green box) the protein abundance returns to a normal level. The response phase shows the largest fold-change and thus corresponds to the primary interval. Since it starts at time 0, we set the start time of RRB1p to

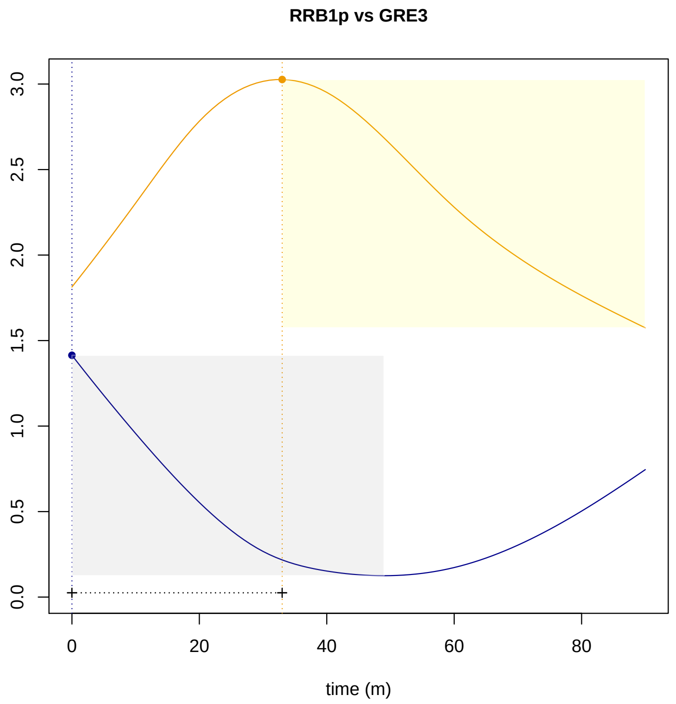

Figure 6.2: RRB1p and GRE3 profiles under oxidative stress. The blue line represents RRB1p profile. The orange line represents GRE3 profile. Boxes show the primary intervals. Dashed lines and points highlight the response start times. The length of the dashed black segment represents the value we infer for the lag variable.

this value, that is $T_s = 0$. Figure 6.2 shows how the lag between RRB1p and GRE3 (a stress induced gene) is evaluated. It reports the RRB1p profile and the GRE3 profile. GRE3 primary interval starts at time 33. Thus, the lag between RRB1p and GRE3 is evaluated as $l = 33 - 0 = 33$ (black segment).

Due to the lower-frequency experimental sampling and to the spline fitting, it often happens that the starting time of Y_j and the starting time of X_i are both inferred to start at time 0. In this case the given procedure would set the lag to 0. In this case we propose to set it equal to the minimum allowed lag, that is $l = 1$.

In the general setting the Granger Causality uses lags to determine the directionality. In our specific case, we imply a specific direction by evaluating the Granger Causality using protein profiles *versus* mRNA profiles. A shortcoming of the current analysis is that due to the unique directionality we impose on the analysis, feed-forward interactions cannot be detected.

6.3 Results

To test our approach we used mRNA and protein concentrations of *S. cerevisiae* in response to a mild oxidative stress induced by diamide [61, 211]. The two experiments have been performed by Vogel *et al.* according to the same experimental design, then measurements have been sampled at fixed time points. Specifically, samples were recovered at 10, 20, 30, 40, 60, and 90 minutes. DNA microarrays were used to measure changes in mRNA levels, while changes in protein levels have been measured by liquid chromatography coupled with tandem mass spectrometry (LC-MS/MS) [155]. 6195 mRNAs/proteins where profiled. Among them we selected 27 mRNAs/proteins having strong pattern variations among both profiles and no missing values. Smoothing splines were fit using the *smooth.spline* function of the *stats* R package [165]. The smoothing parameter was set to 0.4. In order to evaluate our predictions, we downloaded a reference network (*gold standard*) from the BioGRID database [193]. The gold standard is composed by 77 edges.

The network inference problem can be casted as a classification problem, where the class to learn is the presence of edges. Thus, we use as performance measure the Area Under the Receiver Operating Characteristic (AUROC) curve, which is a common statistics for evaluating the goodness of a predictor in binary classification tasks [20].

A classifier prediction is *positive* if it says that an edge exists. Conversely, a classifier prediction is *negative* if it says that an edges does not exist. Within the positive and negative predictions it is necessary to further distinguish between:

- *true positives*: edges correctly predicted to exist; we denote by TP the number of true positive predictions;

- *true negatives*: edges correctly predicted to not exist; we denote by TN the number of true negative predictions;

- *false positives*: edges incorrectly predicted to exist; we denote by FP the number of false positive predictions;

- *false negatives*: edges incorrectly predicted to not exist; we denote by FN the number of false negative predictions.

By leveraging these definitions, it is possible to evaluate the *sensitivity* or *true positive rate* (TPR), i.e., the ability to identify edges in the network as:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}},$$

and the *false positive rate* (FPR), that defines how many incorrect edges we predict:

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}.$$

The Receiver Operating Characteristic (ROC) is the plot (for many pairs) of TPR values versus FPR values, and measures the trade-off between benefits and costs. Usually, reverse engineering algorithms predict a network as a list of edges ranked in a decreasing order of reliability, i.e., edges predicted with high confidence are at the top of the list. Thus, it is possible to plot the ROC by varying the size of the edge list, and to measure how the re-dimensioned lists performs in terms of TPR and FPR. The AUROC is evaluated as the integral of linear interpolation of the ROC. A perfect classification yields to no false positive and no false negative, that corresponds to the point with coordinate $(0, 1)$ in the ROC plot. If this is the case for all the re-dimensioned lists, the curve is actually a rectangle whose area (i.e., the AUROC) is 1. Conversely, a random classifier generates predictions which are most likely on the line $TPR = FPR$, thus having AUROC $\simeq 0.5$. Hence, any classifier with an AUROC > 0.5 is better than the random guess, and optimality is attained when the AUROC approaches 1. The AUROC values have been evaluated as proposed by the DREAM consortium for the evaluation of the network inference challenges (see Section 5.4 and [164, 197] for details).

The Granger based predictor when tested over the mentioned reverse engineering problem obtains an AUROC of 0.68. To assess whether both the mRNA and the protein profiles are necessary to obtain this performance, we test also the performance obtained when the mRNA and the protein profiles alone were used for building the regulatory network. The regulative processes considered were $(X_i, X_j) \in \mathbf{X} \times \mathbf{X}$ and $(Y_i, Y_j) \in \mathbf{Y} \times \mathbf{Y}$, respectively. In both cases the AUROC decreases to 0.59. This supports the claim that the integration of the information contained in both the mRNA and the protein profiles increases the accuracy of the inferred gene regulatory network. Figure 6.3 shows the ROCs we obtained in the described experimental settings.

For the sake of comparison, we ran a pairwise statistical linkage tool tailored to deal with temporal data, namely the TimeDelay-ARACNE algorithm [228] (an extension of the ARACNE algorithm [134]). This latter algorithm retrieves statistical dependencies among expression profiles by exploiting a pairwise time-delayed mutual information measure. The network construction is followed by a pruning step. The author showed that TimeDelay-ARACNE outperforms ARACNE as well as other formalisms for the reverse engineering of gene regulatory networks (i.e., dynamic Bayesian networks and ordinary differential equations). We used the TDARACNE R package and we set all parameters to default values. Since the TimeDelay-ARACNE does not allow to take into consideration both the mRNA and the protein profiles at a time, we ran it over each profile separately.

When applied on the mRNA profiles TD-ARACNE obtains an AUROC value of 0.62. Instead, when applied on the protein profiles it obtained an AUROC value of 0.59. Figure 6.4 shows the ROCs we obtained in the described experimental settings.

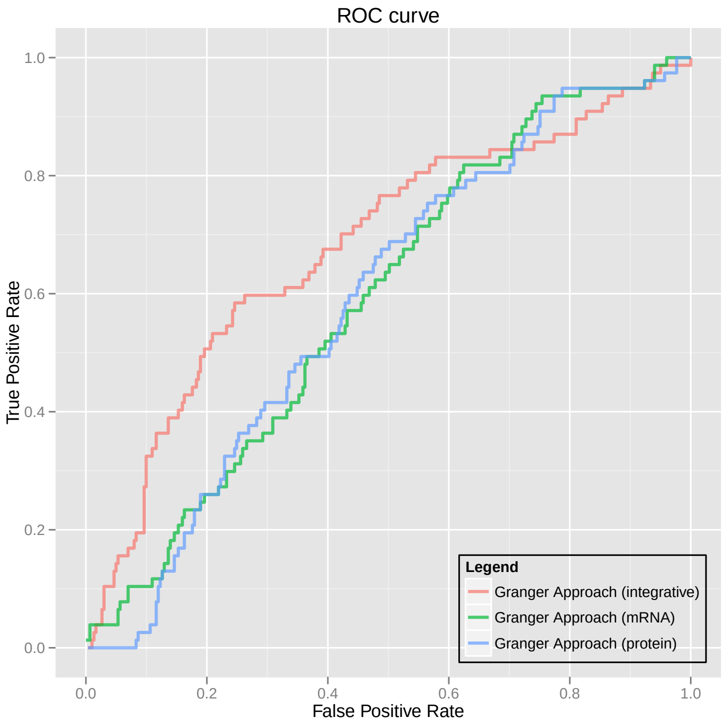

Figure 6.3: ROC obtained by the Granger Causality approach. The red line represents the performance we obtain when the Granger Causality test is evaluated among protein and mRNA profiles. The green (blue) line represents the performance resulting when mRNA (protein) profiles alone are used.

6.4 Conclusions

In this chapter we described a reverse engineering approach that makes use of a well-know econometrics measure, the Granger Causality test, and of two kind of temporal data: the mRNA and the protein profiles. The use of both these features allowed us to obtain promising results over a small *S. cerevisiae* data set.

Let us remark that this work is a preliminary step and further investigation is necessary to address several questions that are still open. First, we plan to test our approach on additional data sets in order to confirm the method performances. Second, the presented pairwise Granger Causality test is prone to two kind of errors: it may incorrectly report a direct relationship where only an indirect influence exists (and vice versa); it may fail in detecting combinatorial influences. In order to solve these issues we plan to try alternative Granger Causality approaches, such as the conditional Granger Causality [230] and the blockwise Granger Causality [65]. Also, we plan

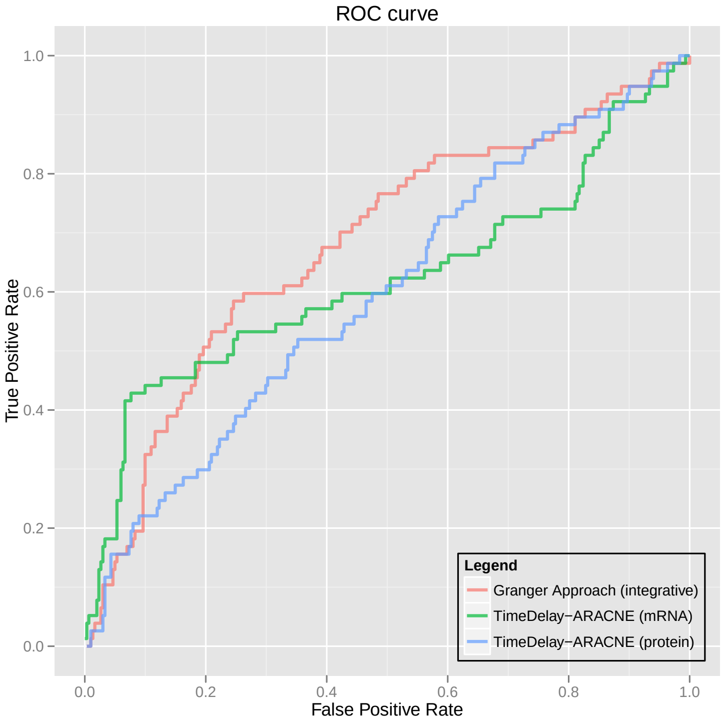

Figure 6.4: ROC obtained by the Granger Causality approach and by the TimeDelay-ARACNE tool. The red line represents the performance we obtain when the Granger Causality test is evaluated on protein and mRNA profiles. The green (blue) line represents the performance resulting when TimeDelay-ARACNE is performed over the mRNA (protein) profiles.

to investigate post-processing techniques that use both Granger Causality outcomes and prior information in order to detect indirect connections.

Part IV

Epilogue

Chapter 7

Pharmacogenes Discovery

Nowadays systems biology approaches, high-throughput data, and biological networks describing cells at a systems level are precious tools to gain an understanding of biological mechanisms. They have led to important fallouts in many areas of biomedical research. For instance, promising results have been obtained in the context of human-health and of pharmacogenomics, i.e., the large-scale study of how genes and their variations impact on drug responses. A crucial issue in this field is the discovery of the genes, called *pharmacogenes*, responsible for cell responses to drugs.

This chapter describes an integrative approach to pharmacogenes identification. It obtains promising and a diagnostic analysis shows that the selected pharmacogenes are strictly related to the administered drugs.

7.1 Introduction

Pharmacogenomics studies the molecular mechanisms of drug action in order to elucidate therapeutic roles of drugs and to improve the effectiveness of responses to therapies. The aim of this discipline is to optimize the drug therapies to ensure the maximum efficacy with the minimum adverse effects.

Most drug actions produce changes in gene expression, and genome-wide platforms provide new perspectives for studying interactions between drugs and organisms by giving a measure of biological effects. Several experimental studies have been performed for investigating drug effects. The main data source has been created by the National Cancer Institute in the Developmental Therapeutics Programs [189]. It consists in an *in vitro* screening of more than 700,000 chemical compounds over 60 human cancer cell-lines (NCI60). The NCI60 panel represents leukaemia, melanoma, and cancers of breast, central nervous system, colon, kidney, lung, ovary, and prostate. For each compound both the gene expression levels and the drug activities have been measured.

A crucial step in pharmacogenomics is to discover genes (*biomarkers* or *pharmacogenes*), that are responsible for drug response. By using each gene as a *feature* and drug response as the *class* to be predicted, the problem can be casted as the one of “finding the set of features that allows the best prediction of the class”. This is a well-known machine learning task: the *feature selection problem.* Feature selection algorithms are usually classified into three categories: filter, wrapper, and embedded approaches [70].

In literature, several filter methods have been described for the task of pharmacogene detection. Many of them exploit statistical tests to rank genes according to their expression profiles, and then identify the k-top genes as pharmacogenes. Among these, the COXEN algorithm is one of the most popular approaches [117]. An important facet of most statistical tests is that they do not consider the correlation among genes, implying that they may lead to a set of redundant pharmacogenes.

Wrapper and embedded methods have also been widely used. These methodologies assess the usefulness of a subset of genes using prediction performances of a given machine learning approach. Wrapper methods search through the space of possible features by selecting a subset of these features, training a predictor using this subset, and scoring its performances. Embedded methods exploit the feature selection mechanism implicitly encoded by some learning algorithm. These two techniques return very accurate results, but they are often computationally expensive making then onerous for a genome-scale analysis [176]. A popular embedded approach is based on the Random Forests algorithm [23]. Random Forests is an ensemble classifier composed by many decision trees. It is considered one of the standard tools for class prediction and gene selection with microarray data [52]. Hu *et al.* [80] employed the Random Forests algorithm to build two gene signatures for predicting the chemosensitivity of breast cancer. The authors evaluated the importance of each gene in both signatures in order to single out pharmacogenes. Ma *et al.* [130] proposed a method to classify cell lines response integrating transcriptional and proteomic profiles by using a Random Forests approach. They showed that the integration of multiple sources of data enhances biomarker selection performances. Riddick *et al.* [171] used Random Forests to create a model of drug response using a multi-step approach.

All the cited approaches exploit only the information carried by gene/protein profiles. However, as shown by Staunton *et al.* [194], a profile analysis can predict chemosensitivity only in a subset of compounds. In fact, it cannot capture gene interactions in biological pathways, although it is widespread believed that gene activities are strictly related to one another. Indeed, the observed changes in gene expression levels may be due to the activation of a *driver*, that is the real responsible for the cell line response [124]. As a consequence, to detect pharmacogenes network information as well as differentially expressed genes must be taken into account [11]. For instance, Taylor *et al.* [202] studied protein interaction networks showing that inter-modular hub proteins (i.e., proteins with a high in/out-degree value) that are co-expressed with their interacting partners are sufficient to predict breast cancer survival rates. Furthermore, independent studies identify non-overlapping sets of breast cancer pharmacogenes that are nonetheless related to the same biological module [188].

Some integrative approaches for pharmacogene identification have been proposed. Huang *et al.* [82] annotated gene expression patterns and drug activities with information derived from KEGG and BioCarta pathways, as well as from the Gene Ontology. Hansen *et al.* [73] proposed an approach merging known gene-gene and drug-gene interactions with available measures of drug-drug similarity. The proposed method is able to rank genes in human genome according to their likelihood of being pharmacogenes. As reviewed by Cun and Fröhlich [45], others approaches propose to bias the feature selection process for selecting connected genes. However, these approaches require the injection of an *a priori* information about gene cooperation that is often not well-understood.

In this chapter we propose an integrated approach to detect pharmacogenes responsible for responses to drug administration. Specifically, our approach integrates: *i)* a filter and a wrapper technique for pharmacogene discovery, and *ii)* three sources of knowledge, namely transcriptional profiles, drugs activity, and pathway interactions. We show that our proposal outperforms the Random Forests approach and that extracted pharmacogenes are strictly related to the administered drugs.

7.2 Methods

We identify pharmacogenes using a two-step approach that requires three kinds of data: *i)* gene expression levels of drug-treated cell lines, *ii)* drug activity data identifying the response (sensitive/resistant) of each cell line, and *iii)* pathway annotations.

In the first step, we apply a filter method. This is a pre-processing technique exploiting *a priori* knowledge about gene profiles and gene interactions. Specifically, we identify differentially expressed genes; then, we perform a pathway analysis in order to select genes that are likely to be responsible for the measured expression profiles. In the second step, we use a wrapper feature selection method to single out pharmacogenes. To this purpose, we use multiple runs of a genetic algorithm to assess the importance of each gene.

Selection of the candidate pharmacogenes

The first feature selection procedure consists of a two-phase filter method. In the first phase, we select the set of differentially expressed genes. The trustworthiness of this operation in pharmacogene discovery is confirmed by several state-of-art-papers [117, 129, 150]. Specifically, a Rank Products test with cutoff 0.05 is used to discover differentially expressed genes [24]. We consider sensitive cell lines as treated conditions and resistant cell lines as control conditions. We denote by D the obtained set of differentially expressed genes.

In the second phase, for each gene $d_i \in D$, we perform a pathway analysis. Specifically, we identify the pathways to which d_i belongs. Then, for each selected pathway, we rank genes according to two different network measures: the in/out degree centrality measure and the betweenness centrality measure. From each ranking we select the two top-ranked genes. This phase allows us to select *key network genes*, i.e., hubs and bottlenecks. Indeed, several works show that these genes are more likely to be essential in biological pathways: hubs correspond to genes having a high probability of playing a central role in the systems-level cellular organization, and bottlenecks establish communication or mass flow within a network [93, 98, 161]. Also, we select genes that are directly connected (both upstream and downstream) to d_i. We denote by P the set of genes identified by the pathways analysis.

The subset of genes obtained by merging D and P is the list of candidate pharmacogenes.

Selection of pharmacogenes

The second feature selection task is performed by means of a genetic algorithm [49]. Genetic algorithms are able to efficiently search in both poorly understood and large feature spaces by a process known as evolution by selection. More in detail, they perform the step-by-step procedure described in the following:

1. a number of random vectors is created; each feature correspond to an entry in these vectors;
2. each vector is evaluated according to a *fitness function* (in our case, the ability to predict the class label of each example in the data set);
3. if a vector shows a fitness score higher than a given threshold τ, it is selected and the procedure stops, otherwise the algorithm continues to step 4
4. the population of vectors is replicated in a such way that ones with high fitness score have higher probability to generate a large number of offsprings; then the vectors are combined through the so called crossover operators, and mutations are introduced randomly;
5. step 2 through step 4 are repeated until the stop condition in step 3 or a fixed number of generations is reached.

Since each evolution outputs a vector and we perform multiple runs of the algorithm, it is necessary to merge the results into a final model. To this aim forward selection strategy can be used: it selects the most frequent features in the population of vectors with the highest accuracies.

We use the genetic algorithm and the forward selection strategy implemented in the GALGO R package [204], using as features the set of candidate pharmacogenes identified by the filter approach, and considering the top-frequent features identified by the forward selection strategy as pharmacogenes. We set a population of 300 vectors, and we fixed $\tau = 0.95$. A Maximum Likelihood Discriminant Functions (MLHD) classifier has been chosen as fitness function for the genetic algorithm [148]. We performed 300 runs of the genetic algorithm.

7.3 Results

Let us recall that the proposed approach requires as input three different knowledge sources: transcriptional profiles, drugs activity data, and pathway interactions. In the following we described the data sets we used.

Transcriptional profiles. We used a gene expression data set generated from the NCI60 cell lines [175]. It contains $1,374$ genes having strong pattern variations among the cell lines, and no more than 4 missing values. We replaced the missing values with the median gene expression levels of the cell lines they belong to, and we used MatchMiner [27] to translate each IMAGE Clone ID to the correspondent gene symbol. This data set, that screens gene expression profiles of 60 different cancer cell lines, allows us to identify pharmacogenes that are independent of the cell lines of origin, and to obtain an higher-level view of drug mechanisms.

Drug activity profiles. The *drug activity* is defined as the compound concentrations required to produce 50% growth inhibition after 48 hours in cell line related to the control. Drug activity profiles we used in this work (GI_{50}) are related to 113 drugs showing a known mechanism of action (Table 7.1, Column 1 and 2) [181]. We processed the activity profiles in order to define drug resistance or sensitivity according to the procedure proposed by Ma *et al.* [130]. In detail, for each drug, we normalized the $log_{10}(GI_{50})$ values across the 60 cell lines. Cell lines having a $log_{10}(GI_{50})$ larger than 0.5 standard deviations above the mean were considered as drug resistant, while cell lines having a $log_{10}(GI_{50})$ lower than 0.5 standard deviations below the mean were considered as drug sensitive. The remaining cell lines, that correspond to an intermediate response, have been removed.

Pathway interactions. We used the pathway annotations stored in the hiPathDB pathway database [222]. hiPathDB integrates $1,661$ pathways acquired from the data of BioCarta, KEGG, NCI-Nature PID, and Reactome.

The proposed methodology returns a number of pharmacogenes ranging from 1 to 13 (Table 7.1, Column 3). The sets of pharmacogenes contains both differentially and not-differentially expressed genes (the percentage of differentially expressed genes shows a mean of 56.4%), confirming the hypothesis that the simple information contained in the transcription profiles is not enough for a correct pharmacogene identification. Genetic algorithm accuracies range from 0.663 to 0.977, with a mean of 0.89 (Table 7.1, Column 4).

To assess the reliability of our approach several validations have been performed. First, we compared the results we obtained to those returned by a Random Forests approach. Then, we demonstrate the biological coherence between drug mechanisms and associated pharmacogenes.

Before delving into the experimentation, it is worth showing that the obtained classifiers make better predictions than random guesses. For each drug we randomly permuted the class labels of the 60 cell lines. This results in the creation of a random prediction. This procedure is repeated 1,000 times. Then, we performed a z-score statistical test: the null hypothesis is that the obtained accuracies are equal to the random guess. Using the aforementioned randomized data set to model the distribution of random predictions, the null hypothesis is rejected at the 95% confidence level for all the 113 classifiers (data not shown).

To show that our approach is more effective than a simple embedded approach, we compared the accuracies we obtained to those returned by a Random Forests algorithm. We used the randomForest R package [123], giving as input all transcription profiles (without performing any filters) and setting the algorithm parameters to default values. We outperformed Random Forests approach for each drug (Table 7.1, column 6).

In analyzing these results it can be argued that the good performances we just reported could also be justified in a case where genes singled out by the filter step were very informative, thus allowing the wrapper approach to deal with a less noisy setting. To rule out this hypothesis we executed the Random Forests algorithm using the list of candidate pharmacogenes as input. Also in this experimental setting our methodology obtained a better accuracy value than the Random Forests approach in 109 out of 113 drugs (Table 7.1, Column 7). Notably, in the remaining 4 cases our accuracies are comparable to those of the competitor. Let us underline that when the list of pharmacogenes are used as input, also the Random Forests accuracies increased in 94 experiments and in 13 cases the obtained results remain steady. It confirms the hypothesis that performing a filter step before a wrapper/embedded approach enhances feature selection performances.

Finally, to show that there exist a biological coherence between sets of extracted pharmacogenes and administered drugs and to rule out the hypothesis of data overfitting, we performed a network analysis using the Ingenuity Pathway Analysis software [92]. Our aim is to show that the identified sets of pharmacogenes enrich the molecular pathways affected by drugs. To this purpose, we checked if there is an overlap between the direct and indirect drug targets and the pharmacogenes, and we verified if pharmacogenes overlay biological pathways that are coherent with the drug functions. The p-values in pathway analysis has been corrected with the Benjamini-Hochberg method [17]. In the following, we summarize our finding for 4 out of the 113 drugs under study: Antifolan, Busulfan, Cytarabine, and Hydroxyurea.

Antifolan (trade name of Methotrexate) is an anti-neoplastic anti-metabolite. It prevents the integrations of these substances with DNA by stopping normal development and division in the S phase of cell cycle. Antifolan selectively affects the most rapidly dividing cells and neoplastic cells [144]. Out of the 26 identified pharmacogenes 3 are known Antifolan targets: AKT, ANXA4 and CCND1. Moreover, 14 pharmacogenes are involved in networks which top functions (according to the evaluated p-values) are cellular development, cell cycle, and cellular growth and proliferation. It agrees with mechanisms involved in the drug response.

Busulfan is an alkylating agent which results in an interference of DNA replication and RNA transcription. It is used to treat various forms of cancer, and it results in a disruption of DNA functions and in cell death [113]. The selected pharmacogenes are enriched in DNA replication, recombination and repair (p-value = $4.13e^{-6}$). Moreover, 10 out of the 27 identified pharmacogenes belong to gene expression, cell death and free radical scavenging networks.

Cytarabine acts through direct DNA damage and incorporation into DNA. It is cytotoxic to a wide variety of proliferating mammalian cells in culture. Although the mechanism of action is not completely understood, it appears that Cytarabine acts through the inhibition of DNA polymerase [162]. The 13 identified pharmacogenes are involved in two biological functions concerned with the proliferation of epithelial cells and with the arrest in G_2/M phase transition of cancer cells (p-values $3.68e^{-8}$ and $1.29e^{-6}$, respectively). Out of the 13 identified pharmacogenes HBE1 and TP53 are known Cytarabine targets.

Hydroxyurea acts on the entire replicase complex, including ribonucleotide reductase. It inhibits the DNA synthesis, leading to cell death in S phase. Also, Hydroxyurea inhibits the repair of DNA damaged by chemicals or irradiation [183]. Among the 10 identified pharmacogenes CCND2 and TP53 are known Hydroxyurea targets, and the biological function associated with the more significative p-value ($2.41e^{-7}$) is the S phase of breast cancer cell lines.

7.4 Conclusions

In this chapter we presented an integrated approach for pharmacogene detection that exploits two feature selection approaches and merges three different sources of knowledge. By applying a filter method before a wrapper approach we decrease the computational demand, without loosing result accuracy. Furthermore, the modularity of the presented approach allows one to inject as much *a priori* knowledge as available. For instance other kinds of data, such

Table 7.1: Analyzed drugs and related results. For each drug the name and the mechanism of action are listed. Mechanisms of action refer to the classification given by Scherf *et al.* [181]. The number of pharmacogenes identified, and the percentage of differentially expressed pharmacogenes are also reported. The last three columns refer to the accuracies obtained by the classifiers trained in the feature selection procedures. The best accuracies are typeset in a boldface font. Experimental setting are described in Section 7.3.

Mechanism of action	Drug Name	Pharmacogenes number	differentially expressed genes (%)	our proposal accuracy	RF accuracy (without filter)	RF accuracy (with filter)
A2 - alkylating agent at N-2 position of guanine	Mitomycin	8	62.5	**0.773**	0.545	0.727
	Porfiromycin	8	50.0	**0.732**	0.500	0.705
A6 - alkylating agent at 0-6 position of guanine	Carmustine (BCNU)	2	50.0	**0.931**	0.759	0.897
	Chlorozotocin	6	100	**0.908**	0.766	0.851
	Clomesone	1	100	**0.882**	0.794	0.853
	Lomustine (CCNU)	7	57.1	**0.897**	0.756	0.756
	Mitozolamide	5	60.0	**0.865**	0.700	0.825
	PCNU	2	50.0	**0.970**	0.867	0.933
	Semustine (MeCCNU)	2	100	**0.844**	0.680	0.840
A7 - alkylating agent at N-7 position of guanine	Asaley	13	69.2	**0.921**	0.737	0.816
	Busulfan	13	69.2	**0.944**	0.865	0.865
	Carboplatin	4	75.0	**0.854**	0.609	0.652
	Chlorambucil	7	14.3	**0.811**	0.674	0.651
	Cisplatin	5	40.0	**0.751**	0.587	0.652
	Cyclodisone	6	66,7	**0.977**	0.655	0.793
	Diaminocyclohexyl-Pt-II	9	55.6	**0.781**	0.744	0.769
	Dianhydrogalactitol	4	100	**0.809**	0.522	0.630
	Diaziridinylbenzoquinone	8	62.5	**0.948**	0.759	0.759
	Fluorodopan	13	69.2	0.860	0.852	**0.889**
	Hepsulfam	8	50.0	**0.882**	0.725	0.775
	Iproplatin	4	50.0	**0.929**	0.742	0.774
	Mechlorethamine	5	20.0	**0.790**	0.725	0.750
	Melphalan	13	53.8	**0.833**	0.583	0.694
	Piperazine mustard	5	100	**0.917**	0.706	0.735
	Piperazinedione	13	46.2	**0.876**	0.578	0.578
	Pipobroman	3	66.7	**0.971**	0.704	0.741
	Spiromustine	5	40.0	**0.910**	0.674	0.744
	Teroxirone	9	55.6	**0.938**	0.611	0.750
	Tetraplatin	13	76.9	0.850	0.829	**0.857**
	Thiotepa	1	0	**0.838**	0.636	0.750
	Triethylenemelamine	13	61.5	**0.868**	0.727	0.750
	Uracil mustard	13	84.6	**0.917**	0.733	0.778
	Yoshi-864	5	60.0	**0.831**	0.650	0.675

Mechanism of action	Drug Name	Pharmacogenes number	differentially expressed genes (%)	our proposal accuracy	RF accuracy (without filter)	RF accuracy (with filter)
T1 - topoisomerase I inhibitor	Camptothecin	13	30.8	**0.875**	0.775	0.750
	Camptothecin,7-Cl	13	46.1	**0.833**	0.673	0.776
	Camptothecin,9-MeO	5	40.0	**0.789**	0.651	0.721
	Camptothecin,9-NH2 (RS)	7	28.6	**0.862**	0.720	0.740
	Camptothecin,9-NH2 (S)	13	84.6	**0.852**	0.686	0.765
	Camptothecin,10-OH	13	46.2	**0.852**	0.750	0.750
	Camptothecin,11-formyl (RS)	13	38.5	**0.865**	0.714	0.743
	Camptothecin,11-HOMe (RS)	13	76.9	**0.904**	0.828	0.862
	Camptothecin,20-ester (S)	10	50.0	**0.946**	0.762	0.786
	Camptothecin,20-ester (S)	5	80.0	**0.877**	0.650	0.750
	Camptothecin,20-ester (S)	8	75.0	**0.867**	0.714	0.738
	Camptothecin,20-ester (S)	13	46.2	0.795	0.667	**0.821**
T2 - topoisomerase II inhibitor	Amonafide	13	46.2	**0.848**	0.636	0.697
	Amsacrine	5	60.0	**0.902**	0.725	0.850
	Anthrapyrazole-derivative	13	69.2	**0.922**	0.756	0.829
	Daunorubicin	5	60.0	**0.854**	0.784	0.784
	Deoxydoxorubicin	10	60.0	0.833	0.676	**0.838**
	Doxorubicin	5	60.0	**0.802**	0.649	0.784
	Etoposide	8	62.5	**0.807**	0.683	0.707
	Menogaril	11	54.5	**0.900**	0.722	0.806
	Mitoxantrone	13	61.5	**0.897**	0.718	0.821
	Oxanthrazole (piroxantrone)	13	76.9	**0.915**	0.773	0.773
	Teniposide	6	83.3	**0.898**	0.667	0.697
	Zorubicin (Rubidazone)	10	60.0	**0.862**	0.406	0.500
Db - DNA binder	Cyanomorpholinodoxorubicin	13	46.2	**0.775**	0.526	0.500
	Hycanthone	5	80.0	**0.817**	0.600	0.633
	Morpholino-adriamycin	5	60.0	**0.805**	0.571	0.600
	N-N-Dibenzyl-daunomycin	11	45.4	**0.807**	0.647	0.618
	Pyrazoloacridine	9	22.2	**0.798**	0.625	0.688
Di - DNA incorporation	5-6-hydro-5-azacytidine	13	61.5	**0.840**	0.641	0.692
	α-2'-Deoxythioguanosine	2	0	**0.812**	0.517	0.690
	Azacytidine	6	50.0	**0.663**	0.333	0.367
	β-2'-Deoxythioguanosine	5	40.0	**0.779**	0.675	0.700
	Thioguanine	13	53.8	**0.688**	0.542	0.667
P90 - hsp90 binder	Geldanamycin	4	75.0	**0.901**	0.767	0.767
Dr - ribonucleotide reductase inhibitor	Guanazole	4	75.0	**0.840**	0.596	0.660
	Hydroxyurea	10	80.0	**0.842**	0.645	0.774
	Pyrazoloimidazole	5	80.0	**0.808**	0.457	0.657

Mechanism of action	Drug Name	Pharmacogenes number	differentially expressed genes (%)	our proposal accuracy	RF accuracy (without filter)	RF accuracy (with filter)
Ds - DNA synthesis inhibitor	Aphidicolin-glycinate	13	76.9	**0.875**	0.674	0.761
	Cyclocytidine	9	100	**0.837**	0.674	0.767
	Cytarabine (araC)	13	69.2	**0.884**	0.571	0.673
	Floxuridine (FUdR)	4	25.0	**0.843**	0.578	0.667
	Fluorouracil (5FU)	3	100	**0.831**	0.647	0.765
	Ftorafur	10	60.0	**0.782**	0.651	0.628
	Thiopurine (6MP)	13	69.2	**0.861**	0.714	0.786
Rs - RNA synthesis inhibitor	Acivicin	4	50.0	**0.783**	0.548	0.645
	Dichloroallyl-lawsone	13	15.4	**0.869**	0.619	0.619
	DUP785 (brequinar)	6	66.7	**0.868**	0.673	0.692
	L-Alanosine	5	40.0	**0.804**	0.424	0.758
	N-phosphonoacetyl-L-aspartic-acid	2	50.0	**0.773**	0.486	0.486
	Pyrazofurin	2	50.0	**0.760**	0.733	0.711
Df - antifols	Aminopterin	12	25.0	**0.804**	0.565	0.696
	Aminopterin-derivative	7	57.1	**0.873**	0.667	0.727
	Aminopterin-derivative	13	53.8	**0.797**	0.510	0.673
	antifolan	13	53.8	**0.700**	0.513	0.538
	antifolan	13	38.5	**0.733**	0.512	0.721
	Baker's-soluble-antifolate	9	44.4	**0.874**	0.688	0.708
	Methotrexate	8	62.5	**0.873**	0.600	0.733
	Methotrexate-derivative	9	66.7	**0.875**	0.775	0.775
	Trimetrexate	13	76.9	**0.702**	0.545	0.614
Tu - tubulin-active antimitotic agents	Colchicine	3	100	**0.898**	0.758	0.818
	Halichondrin B	11	54.5	**0.825**	0.658	0.658
	Maytansine	1	0	**0.864**	0.564	0.692
	Trityl-cysteine	4	50.0	**0.788**	0.697	0.727
	Vinblastine-sulfate	2	50.0	**0.892**	0.821	0.872
	Taxol (Paclitaxel)	10	80.0	**0.927**	0.731	0.846
	Taxol analog	13	53.8	**0.845**	0.724	0.759
	Taxol analog	8	62.5	**0.810**	0.738	0.738
	Taxol analog	13	61.5	**0.838**	0.719	0.750
	Taxol analog	4	50.0	**0.846**	0.758	0.758
	Taxol analog	13	76.9	**0.938**	0.825	0.825
	Taxol analog	7	71.4	**0.819**	0.707	0.805
	Taxol analog	10	60.0	**0.917**	0.700	0.733
	Taxol analog	4	75.0	**0.814**	0.629	0.714
	Taxol analog	10	50.0	**0.894**	0.811	0.838
	Taxol analog	5	60.0	**0.882**	0.692	0.744
	Taxol analog	6	33.3	**0.859**	0.780	0.756
Uk - unknown	3-Hydropicolinaldehyde-thiosemicarbazone	13	69.2	**0.844**	0.526	0.579
	5-Hydroxypicolinaldehyde-thiosemicarbazone	6	83.3	**0.836**	0.568	0.649
	Inosine-glycodialdehyde	5	80.0	**0.919**	0.846	0.897

as protein profiles, can be used to increase the lists of candidate pharmacogenes. Indeed, it is well known that the use of *omics* data allows one to gain a high-resolution genotyping and phenotyping profiling of drug response.

Chapter 8

Conclusions

Systems biology is a recent inter-disciplinary research field aiming at understanding complex biological systems by integrating experimental approaches and by using computational methods. Both the analysis of biological data to extract knowledge about system functioning and the system modeling are challenging tasks.

Within biological systems, the regulatory system (described by means of gene regulatory networks) plays a key role. It describes the regulative machinery and allows the understanding of how modifications in transcriptional regulation affect cell behaviours. Unfortunately, even though numerous efforts have been spent by the systems biology community in this task, a model that can precisely describe the regulative machinery is still not available.

In this thesis, we presented and discussed methodologies that extract and merge regulative information from different knowledge sources.

In the first part of the thesis, we presented works aiming at organizing and extracting knowledge from data. A reorganization of the Gene Ontology (RGO) has been introduced in Chapter 3. The RGO makes explicit useful information that was previously only implicitly represented in the original Gene Ontology structure. By highlighting gene cooperation and by inferring new knowledge about gene functionalities and localizations, the RGO allows the improvements of automated tools that need a structure specifically tailored to recognize the activities of genes/proteins. In Chapter 4 we used the pieces of information codified into the RGO for driving a biclustering process. The metadata-driven procedure we implemented is aimed at discovering transcriptional modules. Results improve on another well-known biclustering algorithm and allows the discovery of genetic interactions that could not be uncovered using expression profiles alone.

In the second part of the thesis, we presented a state-of-the-art of the reverse engineering problem and two attempts of handling it. Chapter 5 presented a Naive Bayes approach that merges information provided by different types of microarray experiments. The modularity of this approach allows one to add pieces of information provided by alternative data sources, such as those described by transcriptional modules. Chapter 6 described a new approach grounded in a well-established econometric measure, i.e., the Granger causality test. It leverages two complementary data sources obtaining good performances in the network reconstruction task. This is yet another evidence supporting the idea that the integration of multiple data sources is a promising approach to deal with complex problems such as those handled by systems biology.

As mentioned, systems biology is a relatively new and challenging research area, where a number of important issues are still open and worth investigation. In Chapter 7, we focused on a specific systems biology application: the detection of pharmacogenes. The problem was casted as the one of finding the best features to solve a classification problem and a combination of a filter and a wrapper method was devised to solve it. On a comparison with commonly used techniques the proposed algorithm improves the recognition rate. Again, multiple knowledge sources has been used, emphasizing the need of an integrative approach in studying complex systems.

Most of the techniques presented in this thesis are suitable for improvements. Future works describing the envisioned extensions has been described at the end of each chapter. For what concerns systems biology at large, instead, it is difficult to foretell what will happen in the years to come. Indeed, the way systems biology is performed today is gradually changing. For instance, Next Generation Sequencing [136] will probably replace microarray technology due to a higher throughput and accuracy and to a lower cost. This allows the sequencing not only of species, but also of different individuals within the species: knowing one's own genome is a possibility that will become open to everyone. Besides, large multivariate experimental designs, such as cohort studies, will probably be extensively used to study complex systems and disease. For sure, data integration as well as computational approaches already developed will be applied to future experiments. However, with the same degree of confidence it can be said that further improvements need to be made in order to deal with the endless challenges that this exciting area of research arises.

Bibliography

[1] L. Alberghina, G. Mavelli, G. Drovandi, P. Palumbo, S. Pessina, F. Tripodi, P. Coccetti, and M. Vanoni. Cell growth and cell cycle in saccharomyces cerevisiae: Basic regulatory design and protein-protein interaction network. *Biotechnology advances*, 2011. 42

[2] R. Albert. Scale-free networks in cell biology. *Journal of Cell Science*, 118:4947–4957, 2005. 10, 51

[3] R. Albert and H.G. Othmer. The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster. *Journal of Theoretical Biology*, 223(1):1–18, 2003. 49

[4] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *How Cells Read the Genome: From DNA to Protein*, chapter 6. Garland Science, 4th edition, 2002. 9

[5] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *The Cell Cycle and Programmed Cell Death*, chapter 17. Garland Science, 4th edition, 2002. 24, 27

[6] G. Alterovitz, M. Xiang, D. P. Hill, J. Lomax, J. Liu, M. Cherkassky, J. Dreyfuss, C. Mungall, M. A. Harris, M. E. Dolan, J. A. Blake, and M. F. Ramoni. Ontology engineering. *Nat Biotech*, 28(2):128–130, 02 2010. 18

[7] R.B. Altman, C.M. Bergman, J. Blake, C. Blaschke, A. Cohen, F. Gannon, L. Grivell, U. Hahn, W. Hersh, L. Hirschman, Jensen. L.J., M. Krallinger, B. Mons, S.I. O'Donoghue, M.C. Peitsch, D. Rebholz-Schuhmann, H. Shatkay, and A. Valencia. Text mining for biology-the way forward: opinions from leading scientists. *Genome Biol*, 9(Suppl 2):S7, 2008. 12

[8] P.W. Anderson. More is different. *Science*, 177(4047):393–396, 1972. 3, 9

[9] P. Antony, R. Balling, and N. Vlassis. From Systems Biology to Systems Biomedicine. *Current Opinion in Biotechnology*, 2011. 4

[10] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics*, 25(1):25–29, 2000. 13, 17

[11] F. Azuaje. What does systems biology mean for biomarker discovery? *Expert Opinion on Medical Diagnostics*, 4(1):1–10, 2010. 72

[12] M. Bansal, G. Della Gatta, and D. Di Bernardo. Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, 22(7):815–822, 2006. 50

[13] Z. Bar-Joseph. Analyzing time series gene expression data. *Bioinformatics*, 20(16):2493–2503, 2004. 61

[14] A. L. Barabàsi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999. 10, 51

[15] K. Basso, A.A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano. Reverse engineering of regulatory networks in human B cells. *Nature genetics*, 37(4):382–390, 2005. 48

[16] E. Beisswanger, V. Lee, J.J. Kim, D. Rebholz-Schuhmann, A. Splendiani, O. Dameron, S. Schulz, and U Hahn. Gene Regulation Ontology (GRO): Design Principles and Use Cases. *Studies in Health Technology and Informatics*, 136:9–14, 2008. 18

[17] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 289–300, 1995. 76

[18] R. Bonneau, D.J. Reiss, P. Shannon, M. Facciotti, L. Hood, N.S. Baliga, and V. Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome biology*, 7(5):R36, 2006. 4

[19] K. Bozek, A. Relógio, S.M. Kielbasa, M. Heine, C. Dame, A. Kramer, and H. Herzel. Regulation of clock-controlled genes in mammals. *PLoS One*, 4(3):e4882, 2009. 4

[20] A.P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997. 64

[21] M. Brameier and C. Wiuf. Co-clustering and visualization of gene expression data and gene ontology terms for Saccharomyces cerevisiae using self-organizing maps. *Journal of biomedical informatics*, 40(2):160–173, 2007. 32

[22] A. Brazma, P. Hingamp, J. Quackenbush, G. Sherlock, P. Spellman, C. Stoeckert, J. Aach, W. Ansorge, C. A. Ball, H. C. Causton, T. Gaasterland, P. Glenisson, F. C. Holstege, I. F. Kim, V. Markowitz, J. C. Matese, H. Parkinson, A. Robinson, U. Sarkans, S. Schulze-Kremer, J. Stewart, R. Taylor, J. Vilo, and M. Vingron. Minimum information about a microarray experiment (MIAME)-toward standards for microarray data. *Nature genetics*, 29(4):365–71, December 2001. 11

[23] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001. 72

[24] R. Breitling, P. Armengaud, A. Amtmann, and P. Herzyk. Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS letters*, 573(1):83–92, 2004. 11, 73

[25] A. Bruex, R.M. Kainkaryam, Y. Wieckowski, Y.H. Kang, C. Bernhardt, Y. Xia, X. Zheng, J.Y. Wang, M.M. Lee, P. Benfey, P.J. Woolf, and J. Schiefelbein. A Gene Regulatory Network for Root Epidermis Cell Differentiation in Arabidopsis. *PLoS Genetics*, 8(1):e1002446, 2012. 4

[26] M.L. Bulyk, X. Huang, Y. Choo, and G.M. Church. Exploring the DNA-binding specificities of zinc fingers with DNA microarrays. *Proceedings of the National Academy of Sciences*, 98(13):7158, 2001. 12

[27] K.J. Bussey, D. Kane, M. Sunshine, S. Narasimhan, S. Nishizuka, W.C. Reinhold, B. Zeeberg, W. Ajay, and JN Weinstein. MatchMiner: a tool for batch navigation among gene and gene product identifiers. *Genome Biol*, 4(4):R27, 2003. 75

[28] A.J. Butte and I.S. Kohane. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *Pac Symp Biocomput*, volume 5, pages 418–429, 2000. 48

[29] A. Califano. SPLASH: structural pattern localization analysis by sequential histograms. *Bioinformatics*, 16(4):341, 2000. 12

[30] G.A. Calin and C.M. Croce. MicroRNA-cancer connection: the beginning of a new tale. *Cancer research*, 66(15):7390, 2006. 8

[31] D.T.H. Chang, C.Y. Huang, C.Y. Wu, and W.S. Wu. YPA: an integrated repository of promoter features in Saccharomyces cerevisiae. *Nucleic acids research*, 39(suppl 1):D647–D652, 2011. 43

[32] C. Chaouiya, E. Remy, P. Ruet, and D. Thieffry. Qualitative modelling of genetic networks: From logical regulatory graphs to standard Petri nets. *Applications and Theory of Petri Nets 2004*, pages 137–156, 2004. 50

[33] C. Chaouiya, E. Remy, and D. Thieffry. Petri net modelling of biological regulatory networks. *Journal of Discrete Algorithms*, 6(2):165–177, 2008. 50

[34] Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings ISMB 2000*, pages 93–103, 2000. 32

[35] J.M. Cherry, E.L. Hong, C. Amundsen, R. Balakrishnan, G. Binkley, E.T. Chan, K.R. Christie, M.C. Costanzo, S.S. Dwight, S.R. Engel, et al. Saccharomyces Genome Database: the genomics resource of budding yeast. *Nucleic Acids Research*, 40(D1):D700–D705, 2012. 13

[36] H. Choi and N. Pavelka. When one and one gives more than two: challenges and opportunities of integrative omics. *Frontiers in Genetics*, 2, 2012. 4

[37] R. Clarke, H.W. Ressom, A. Wang, J. Xuan, M.C. Liu, E.A. Gehan, and Y. Wang. The properties of high-dimensional data spaces: implications for exploring gene and protein expression data. *Nature Reviews Cancer*, 8(1):37–49, 2008. 4

[38] G.C. Conant and A. Wagner. Convergent evolution of gene circuits. *Nature genetics*, 34(3):264–266, 2003. 10

[39] A. Conesa, Nueda M.J., A. Ferrer, and M. Talón. maSigPro: a method to identify significantly differential expression profiles in time-course microarray experiments. *Bioinformatics*, 22(9):1096–1102, 2006. 11

[40] The Gene Ontology Consortium. The Gene Ontology web-site. http://www.geneontology.org/. 13, 17

[41] The Gene Ontology Consortium. The Gene Ontology in 2010: extensions and refinements. *Nucleic acids research*, 38(Database issue):D331–5, January 2010. 17

[42] F. Cordero, R. G. Pensa, A. Visconti, D. Ienco, and M. Botta. Ontology-Driven Co-clustering of Gene Expression Data. *AI* IA 2009: Emergent Perspectives in Artificial Intelligence*, pages 426–435, 2009. 33

[43] F Crick. Ideas on Protein Synthesis. In *Symp. Soc. Exp. Biol. XII*, pages 139–163, 1958. 8

[44] G. Csárdi, Z. Kutalik, and S. Bergmann. Modular analysis of gene expression data with R. *Bioinformatics*, 26(10):1376, 2010. 38

[45] Y. Cun and H. Fröhlich. Biomarker gene signature discovery integrating network knowledge. *Biology*, 1(1):5–17, 2012. 73

[46] H.J. Dai, Y.C. Chang, R. Tzong-Han Tsai, and W.L. Hsu. New challenges for biological text-mining in the next decade. *Journal of Computer Science and Technology*, 25(1):169–179, 2010. 12

[47] S. Datta and S. Datta. Methods for evaluating clustering algorithms for gene expression data using a reference set of functional classes. *BMC bioinformatics*, 7(1):397, 2006. 37

[48] H. De Jong. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of computational biology*, 9(1):67–103, 2002. 50

[49] K. De Jong. Adaptive system design: a genetic approach. *Systems, Man and Cybernetics, IEEE Transactions on*, 10(9):566–574, 1980. 74

[50] G. Della Gatta, M. Bansal, A. Ambesi-Impiombato, D. Antonini, C. Missero, and D. Di Bernardo. Direct targets of the TRP63 transcription factor revealed by a combination of gene expression profiling and reverse engineering. *Genome research*, 18(6):939–948, 2008. 50

[51] G. Dennis Jr, B.T. Sherman, D.A. Hosack, J. Yang, W. Gao, H.C. Lane, and R.A. Lempicki. DAVID: database for annotation, visualization, and integrated discovery. *Genome Biol*, 4(5):P3, 2003. 25

[52] R. Díaz-Uriarte and S.A. De Andres. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1):3, 2006. 72

[53] DREAM. Dialogue for Reverse Engineering Assessments and Methods. http://wiki.c2b2.columbia.edu/dream/index.php/The_DREAM_Project, 2010. 48, 53, 55

[54] P. D'haeseleer, S. Liang, and R. Somogyi. Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, 16(8):707, 2000. 49

[55] R. Edgar and M Domrachev. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic acids research*, 30(1):207–210, January 2002. 11

[56] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863, 1998. 31, 48

[57] N. Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799, 2004. 49

[58] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian Networks to Analyze Expression Data. *Journal of Computational Biology*, 7(3-4):601–620, 2000. 49

[59] Y. Fu, L. Jarboe, and J. Dickerson. Reconstructing genome-wide regulatory network of E. coli using transcriptome data and predicted transcription factor activities. *BMC bioinformatics*, 12(1):233, 2011. 4

[60] B. Futcher, GI Latter, P. Monardo, CS McLaughlin, and JI Garrels. A sampling of the yeast proteome. *Molecular and Cellular Biology*, 19(11):7357, 1999. 8

[61] A.P. Gasch, P.T. Spellman, C.M. Kao, O. Carmel-Harel, M.B. Eisen, G. Storz, D. Botstein, and P.O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Science's STKE*, 11(12):4241, 2000. 64

[62] D. Gatherer. So what do we really mean when we say that systems biology is holistic? *BMC systems biology*, 4(1):22, 2010. 3

[63] F. Geier, J. Timmer, and C. Fleck. Reconstructing gene-regulatory networks from time series, knock-out data, and prior knowledge. *BMC Syst. Biol.*, 1(11), 2007. 4

[64] R.C. Gentleman, V.J. Carey, D.M. Bates, and others. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80, 2004. 11

[65] J. Geweke. Measurement of linear dependence and feedback between multiple time series. *Journal of the American Statistical Association*, pages 304–313, 1982. 66

[66] W. Görner, E. Durchschlag, J. Wolf, E.L. Brown, G. Ammerer, H. Ruis, and C. Schüller. Acute glucose starvation activates the nuclear localization signal of a stress-specific yeast transcription factor. *The EMBO journal*, 21(1):135–144, 2002. 43

[67] C.W.J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, pages 424–438, 1969. 60

[68] D. Greenbaum, R. Jansen, and M. Gerstein. Analysis of mRNA expression and protein abundance data: an approach for the comparison of the enrichment of features in the cellular population of proteins and transcripts. *Bioinformatics*, 18(4):585, 2002. 8

[69] N. Guelzim, S. Bottani, P. Bourgine, and F. Képès. Topological and causal structure of the yeast transcriptional regulatory network. *Nature genetics*, 31(1):60–63, 2002. 4

[70] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003. 72

[71] S.P. Gygi, Y. Rochon, B.R. Franza, and R. Aebersold. Correlation between protein and mRNA abundance in yeast. *Molecular and cellular biology*, 19(3):1720–1730, 1999. 8

[72] D. Hanisch, A. Zien, R. Zimmer, and T. Lengauer. Co-clustering of biological networks and gene expression data. *Bioinformatics*, 18(suppl 1):S145–S154, 2002. 32

[73] N.T. Hansen, S. Brunak, and RB Altman. Generating genome-scale candidate gene lists for pharmacogenomics. *Clinical Pharmacology & Therapeutics*, 86(2):183–189, 2009. 73

[74] A. J. Hartemink, D. K. Gifford, T. S. Jaakkola, and R. A. Young. Combining location and expression data for principled discovery of genetic regulatory network models. In *Proceeding of the Pacific Symposium on Biocomputing*, pages 437–449, 2002. 4

[75] A.J. Hartemink, D.K. Gifford, T.S. Jaakkola, and R.A. Young. Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks. In *Pacific Symposium on Biocomputing*, volume 6, pages 422–433, 2001. 49

[76] L.H. Hartwell, J.J. Hopfield, S. Leibler, and A.W. Murray. From molecular to modular cell biology. *Nature*, 402(6761):47, 1999. 10

[77] M. Hecker, S. Lamberk, S. Toepfer, E. van Someren, and R. Guthke. Gene regulatory network inference: Data integration in dynamic models - A review. *BioSystems*, 96:86–103, 2009. 4

[78] R. Hofestadt. A Petri net application of metabolic processes. *Journal of System Analysis, Modeling and Simulation*, 16:113–122, 1994. 50

[79] L. Hood, J.R. Heath, M.E. Phelps, and B. Lin. Systems biology and new technologies enable predictive and preventative medicine. *Science's STKE*, 306(5696):640, 2004. 4

[80] W. Hu. Identifying predictive markers of chemosensitivity of breast cancer with random forests. *Journal of Biomedical Science and Engineering*, 3:59–64, 2010. 72

[81] D. W. Huang, B.T. Sherman, and R.A. Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucl Acids Res*, 37:1–13, 2009. 18

[82] R. Huang, A. Wallqvist, N. Thanki, and DG Covell. Linking pathway gene expressions to the growth inhibition response from the national cancer institute's anticancer screen and drug mechanism of action. *The Pharmacogenomics Journal*, 5(6):381–399, 2005. 73

[83] W. Huber, A. Heydebreck, and M. Vingron. Analysis of microarray gene expression data. In *Handbook of Statistical Genetic, 2nd edn*, chapter 6, pages 331–363. Wiley, 2003. 11

[84] T.R. Hughes, M.J. Marton, A.R. Jones, C.J. Roberts, R. Stoughton, C.D. Armour, H.A. Bennett, E. Coffey, H. Dai, Y.D. He, Y.D. Matthew, J. Kidd, A.M. King, M.R. Meyer, D. Slade, P.Y. Lum, S.B. Stepaniants, D.D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S.H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, 2000. 37

[85] C.A. Hunt, G.E.P. Ropella, S. Park, and J. Engelberg. Dichotomies between computational and mathematical models. *nature biotechnology*, 26(7):737–738, 2008. 47

[86] V.A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts. Inferring regulatory networks from expression data using tree-based methods. *PLoS One*, 5(9):e12776, 2010. 50

[87] T. Ideker, T. Galitski, and L. Hood. A new approach to decoding life: systems biology. *Annual review of genomics and human genetics*, 2(1):343–372, 2001. 3

[88] T. Ideker and D. Lauffenburger. Building with a scaffold: emerging strategies for high-to low-level cellular modeling. *TRENDS in Biotechnology*, 21(6):255–262, 2003. 48

[89] J. Ihmels, S. Bergmann, and N. Barkai. Defining transcription modules using large-scale gene expression data. *Bioinformatics*, 20(13):1993–2003, 2004. 32, 38

[90] J. Ihmels, G. Friedlander, S. Bergmann, O. Sarig, Y. Ziv, and N. Barkai. Revealing modular organization in the yeast transcriptional network. *Nature genetics*, 31(4):370–378, 2002. 32

[91] S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, and S. Miyano. Combining microarrays and biological knowledge for estimating gene networks via Bayesian networks. In *Proceeding of the 2nd IEEE Computer Society Bioinformatics Conference*, page 104–113, 2003. 4

[92] Ingenuity Pathway Analysis. Ingenuity Systems - IPA. http://www.ingenuity.com/, 2012. 76

[93] H. Jeong, S.P. Mason, A.L. Barabasi, and Z.N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411:41–42, 2001. 74

[94] D. Jiang, C. Tang, and A. Zhang. Cluster analysis for gene expression data: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, 16(11):1370–1386, 2004. 11, 31

[95] B. Jin and X. Lu. Identifying informative subsets of the Gene Ontology with information bottleneck methods. *Bioinformatics*, 26(19):2445–2451, August 2010. 18

[96] D.S. Johnson, A. Mortazavi, R.M. Myers, and B. Wold. Genome-wide mapping of in vivo protein-DNA interactions. *Science*, 316(5830):1497, 2007. 12

[97] K.S. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972. 21

[98] M.P. Joy, A. Brock, D.E. Ingber, and S. Huang. High-betweenness proteins in the yeast protein interaction network. *Journal of Biomedicine and Biotechnology*, 2005(2):96–103, 2005. 74

[99] M. Kanehisa, M. Araki, S. Goto, M. Hattori, M. Hirakawa, M. Itoh, T. Katayama, S. Kawashima, S. Okuda, T. Tokimatsu, and Y. Yamanishi. KEGG for linking genomes to life and the environment. *Nucleic acids research*, 36:D480–4, January 2008. 13

[100] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, and M. Hirakawa. From genomics to chemical genomics: new developments in KEGG. *Nucleic acids research*, 34, January 2006. 13

[101] M. Kanehisa, S. Goto, Y. Sato, M. Furumichi, and M. Tanabe. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic acids research*, 40(D1):D109–114, November 2011. 13

[102] S.A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969. 49

[103] F. Képès. *Biological networks*, volume 3. World Scientific Pub Co Inc, 2007. 9, 10

[104] H. Kitano. Computational systems biology. *Nature*, 420(6912):206–10, November 2002. 3

[105] H. Kitano. Looking beyond the details: a rise in system-oriented approaches in genetics and molecular biology. *Current genetics*, 41(1):1–10, April 2002. 3

[106] H. Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002. 3, 10

[107] H. Kitano. Biological robustness. *Nature Reviews Genetics*, 5(11):826–837, 2004. 10

[108] C.A. Klein. Gene expression sigantures, cancer cell evolution and metastatic progression. *Cell cycle (Georgetown, Tex.)*, 3(1):29, 2004. 4

[109] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT Press, 2009. 49

[110] E. V. Kriventseva, W.g Fleischmann, E. M. Zdobnov, and R. Apweiler. CluSTr: a database of clusters of SWISS-PROT+TrEMBL proteins. *Nucleic Acids Research*, 29(1):33–36, 2001. 18

[111] R. Küffner, T. Petri, P. Tavakkolkhah, L. Windhager, and R. Zimmer. Inferring Gene Regulatory Networks by ANOVA. *Bioinformatics*, 2012. 50

[112] Kanehisa Laboratories. KEGG: Kyoto Encyclopedia of Genes and Genomes. http://www.genome.jp/kegg/. 13

[113] H Lage and M. Dietel. Involvement of the dna mismatch repair system in antineoplastic drug resistance. *J Cancer Res Clin Oncol*, 125(3):156–165, 1999. 77

[114] M.C. Lagomarsino, B. Bassetti, G. Castellani, and D. Remondini. Functional models for large-scale gene regulation networks: realism and fiction. *Mol. BioSyst.*, 5(4):335–344, 2009. 47

[115] P. P Le, A. Bahl, and L. H. Ungar. Using prior knowledge to improve genetic network reconstruction from microarray data. *Silico Biol.*, 4(3):335–353, 2004. 4

[116] I. Lee, S.V. Date, A.T. Adai, and E.M. Marcotte. A probabilistic functional network of yeast genes. *science*, 306(5701):1555, 2004. 48

[117] J.K. Lee, D.M. Havaleshko, H.J. Cho, J.N. Weinstein, E.P. Kaldjian, J. Karpovich, A. Grimshaw, and D. Theodorescu. A strategy for predicting the chemosensitivity of human cancers and its application to drug discovery. *Proceedings of the National Academy of Sciences*, 104(32):13086, 2007. 72, 73

[118] S. G. Lee, J. U. Hur, and Y. S. Kim. A graph-theoretic modeling on GO space for biological interpretation of gene clusters. *Bioinformatics*, 20(3):381–388, 2004. 18

[119] W.P. Lee and W.S. Tzou. Computational methods for discovering gene networks from expression data. *Briefings in bioinformatics*, 10(4):408–423, 2009. 4, 5, 47

[120] C. Lefebvre, G. Rieckhof, and A. Califano. Reverse-engineering human regulatory networks. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 2012. 47, 48

[121] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14):4781, 2004. 49

[122] S. Liang, S. Fuhrman, and R. Somogyi. REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. In *Pacific symposium on biocomputing*, pages 18–29, 1998. 49

[123] A. Liaw and M. Wiener. Classification and Regression by randomForest. *R News*, 2(3):18–22, 2002. 76

[124] W.K. Lim, E. Lyashenko, and A. Califano. Master regulators used as breast cancer metastasis classifier. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 504. NIH Public Access, 2009. 72

[125] D.R. Lorenz, C.R. Cantor, and J.J. Collins. A network biology approach to aging in yeast. *Proceedings of the National Academy of Sciences*, 106(4):1145, 2009. 50

[126] A.C. Lozano, N. Abe, Y. Liu, and S. Rosset. Grouped graphical Granger modeling for gene expression regulatory networks discovery. *Bioinformatics*, 25(12):i110–i118, 2009. 60, 62

[127] P. Lu, C. Vogel, R. Wang, X. Yao, and E.M. Marcotte. Absolute protein expression profiling estimates the relative contributions of transcriptional and translational regulation. *Nature biotechnology*, 25(1):117–124, 2006. 8

[128] S.P. Lu and S.J. Lin. Regulation of yeast sirtuins by NAD+ metabolism and calorie restriction. *Biochimica et Biophysica Acta (BBA)-Proteins & Proteomics*, 1804(8):1567–1575, 2010. 42

[129] J. Lyons-Weiler, S. Patel, M. Becich, and T. Godfrey. Tests for finding complex patterns of differential expression in cancers: towards individualized medicine. *BMC bioinformatics*, 5(1):110, 2004. 73

[130] Y. Ma, Z. Ding, Y. Qian, Y.W. Wan, K. Tosun, X. Shi, V. Castranova, E.J. Harner, and N.I. Guo. An integrative genomic and proteomic approach to chemosensitivity prediction. *International journal of oncology*, 34(1):107, 2009. 72, 75

[131] D. Machado, R.S. Costa, M. Rocha, E.C. Ferreira, B. Tidor, and I. Rocha. Modeling formalisms in Systems Biology. *AMB Express*, 1(1):45, 2011. 47

[132] M. Madan Babu, S.A. Teichmann, and L. Aravind. Evolutionary dynamics of prokaryotic transcriptional regulatory networks. *Journal of molecular biology*, 358(2):614–633, 2006. 4

[133] S. C. Madeira and A. L. Oliveira. Biclustering Algorithms for Biological Data Analysis: A Survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, 2004. 11, 32

[134] A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Favera, and A. Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*, 7(Suppl 1):S7, 2006. 48, 65

[135] A. Mazurie, S. Bottani, and M. Vergassola. An evolutionary and functional assessment of regulatory network motifs. *Genome Biology*, 6(4):R35, 2005. 60

[136] M.L. Metzker. Sequencing technologies—the next generation. *Nature Reviews Genetics*, 11(1):31–46, 2009. 51, 84

[137] HW Mewes, D Frishman, U. Güldener, G. Mannhaupt, K. Mayer, M. Mokrejs, B. Morgenstern, M. Münsterkötter, S. Rudd, and B. Weil. MIPS: a database for genomes and protein sequences. *Nucleic acids research*, 30(1):31, 2002. 27

[138] G. Miller. Wordnet A lexical database for English. *Communications of ACM*, 38(11):39–41, 1995. 30

[139] R. Mobini, B. Andersson, J. Erjefält, M. Hahn-Zoric, M. Langston, A. Perkins, L. Cardell, and M. Benson. A module-based analytical strategy to identify novel disease-associated genes shows an inhibitory role for interleukin 7 receptor in allergic inflammation. *BMC systems biology*, 3(1):19, 2009. 5

[140] J. Montojo, K. Zuberi, H. Rodriguez, F. Kazi, G. Wright, SL Donaldson, Q. Morris, and GD Bader. GeneMANIA Cytoscape plugin: fast gene function predictions on the desktop. *Bioinformatics*, 26(22):2927–2928, 2010. 44

[141] D. O. Morgan. *The Cell Cycle, Principales of control.* NewScience Press, 2007. 24

[142] C. J. Mungall. Obol: integrating language and meaning in bio-ontologies. *Comparative and Functional Genomics*, 5:509–520, 2004. 30

[143] C. J. Mungall, M. Bada, T. Z. Berardini, J. Deegan, A. Ireland, M. A. Harris, D. P. Hill, and J. Lomax. Cross-product extensions of the Gene Ontology. *Journal of Biomedical Informatics*, 2009. 18

[144] MU Naidu, GV Ramana, PU Rani, IK Mohan, A Suman, and Roy P. Chemotherapy-induced and/or radiation therapy-induced oral mucositis–complicating the treatment of cancer. *Neoplasia*, 6(5):423–431, 2004. 77

[145] H Ogata, S. Goto, K. Sato, W. Fujibuchi, H. Bono, and M. Kanehisa. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 27(1):29–34, January 1999. 13

[146] S. Ohno and al. So much "junk" DNA in our genome. In *Brookhaven symposia in biology*, volume 23, page 366, 1972. 7

[147] I.M. Ong, J.D. Glasner, and D. Page. Modelling regulatory pathways in E. coli from time series expression profiles. *Bioinformatics*, 18(suppl 1):S241–S248, 2002. 49

[148] CH Ooi and P. Tan. Genetic algorithms applied to multi-class prediction for the analysis of gene expression data. *Bioinformatics*, 19(1):37–44, 2003. 75

[149] H. Parkinson, M. Kapushesky, N. Kolesnikov, G. Rustici, M. Shojatalab, N. Abeygunawardena, H. Berube, M. Dylag, I. Emam, A. Farne, E. Holloway, M. Lukk, J. Malone, R. Mani, E. Pilicheva, T. F Rayner, F. Rezwan, A. Sharma, E. Williams, X. Z. Bradley, T. Adamusiak, M. Brandizi, T. Burdett, R. Coulson, M. Krestyaninova, P. Kurnosov, E. Maguire, S. G. Neogi, P. Rocca-Serra, S. A. Sansone, N. Sklyar, M. Zhao, U. Sarkans, and A. Brazma. ArrayExpress update–from an archive of functional genomics experiments to the atlas of gene expression. *Nucleic acids research*, 37(Database issue):D868–72, January 2009. 11

[150] P. Pavlidis and P. Poirazi. Individualized markers optimize class prediction of microarray data. *BMC bioinformatics*, 7(1):345, 2006. 73

[151] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan Kaufmann, 1988. 49

[152] H. Pearson. What is a gene? *Nature*, 441:399–401, 2006. 7

[153] K. Pearson. Notes on the history of correlation. *Biometrika*, 13(1):25–45, 1920. 53

[154] C.A. Penfold and D.L. Wild. How to infer gene networks from expression profiles, revisited. *Interface Focus*, 1(6):857–870, 2011. 60

[155] J. Peng, J.E. Elias, C.C. Thoreen, L.J. Licklider, and S.P. Gygi. Evaluation of multidimensional chromatography coupled with tandem mass spectrometry (LC/LC-MS/MS) for large-scale protein analysis: the yeast proteome. *Journal of proteome research*, 2(1):43–50, 2003. 64

[156] R.G. Pensa and J.F. Boulicaut. Constrained co-clustering of gene expression data. In *Proceedings SIAM SDM*, pages 25–36, 2008. 33

[157] R.G. Pensa, J.F. Boulicaut, F. Cordero, and M. Atzori. Co-clustering numerical data under user-defined constraints. *Statistical Analysis and Data Mining*, 3(1):38–55, 2010. 33

[158] B.E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. d'Alche Buc. Gene networks inference using dynamic Bayesian networks. *Bioinformatics*, 19(suppl 2):ii138, 2003. 49

[159] C.A. Petri. *Kommunikation mit Automaten.* PhD thesis, PhD thesis, Technical University Darmstadt, 1962. 50

[160] E.M. Phizicky and S. Fields. Protein-protein interactions: methods for detection and analysis. *Microbiological reviews*, 59(1):94–123, 1995. 12

[161] A.P. Potapov, N. Voss, N. Sasse, and E. Wingender. Topology of mammalian transcription networks. *Genome Informatics Series*, 16(2):270, 2005. 74

[162] AS Prakasha Gowda, JM Polizzi, KA Eckert, and Spratt TE. Incorporation of gemcitabine and cytarabine into DNA by DNA polymerase beta and ligase. *Biochemistry*, 43(29):4833–4840, 2010. 77

[163] A. Prelić, S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, 22(9):1122, 2006. 38

[164] R.J. Prill, D. Marbach, J. Saez-Rodriguez, P.K. Sorger, L.G. Alexopoulos, X. Xue, N.D. Clarke, G. Altan-Bonnet, and G. Stolovitzky. Towards a rigorous assessment of systems biology models: the dream3 challenges. *PloS one*, 5(2):e9202, 2010. 54, 57, 65

[165] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. 11, 64

[166] C. Rangel, J. Angus, Z. Ghahramani, M. Lioumi, E. Sotheran, A. Gaiba, D.L. Wild, and F. Falciani. Modeling T-cell activation using gene expression profiling and state-space models. *Bioinformatics*, 20(9):1361–1372, 2004. 49

[167] V.N. Reddy, M.L. Mavrovouniotis, and M.N. Liebman. Petri net representations in metabolic pathways. In *Proc Int Conf Intell Syst Mol Biol*, volume 1, pages 328–36, 1993. 50

[168] D. Reiss, N. Baliga, and R. Bonneau. Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks. *BMC bioinformatics*, 7(1):280, 2006. 4, 33

[169] B. Ren, F. Robert, J.J. Wyrick, O. Aparicio, E.G. Jennings, I. Simon, J. Zeitlinger, J. Schreiber, N. Hannett, L.T Kanin, E. Volkert, C.J. Wilson, S.P. Bell, and Young R.A. Genome-wide location and function of DNA binding proteins. *Science*, 290(5500):2306, 2000. 12

[170] P. Resnik. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *J. Artif. Intell. Res. (JAIR)*, 11:95–130, 1999. 18

[171] G. Riddick, H. Song, S. Ahn, J. Walling, D. Borges-Rivera, W. Zhang, and H.A. Fine. Predicting in vitro drug sensitivity using Random Forests. *Bioinformatics*, 27(2):220, 2011. 72

[172] M.D. Robinson, J. Grigull, N. Mohammad, and T.R. Hughes. FunSpec: a web-based cluster interpreter for yeast. *BMC bioinformatics*, 3(1):35, 2002. 27

[173] D. J. Rogers and T. T. Tanimoto. A Computer Program for Classifying Plants. *Science*, 132:1115–1118, 1960. 33

[174] S. Rogers, M. Girolami, W. Kolch, K.M. Waters, T. Liu, B. Thrall, and H.S. Wiley. Investigating the correspondence between transcriptomic and proteomic expression profiles using coupled cluster models. *Bioinformatics*, 24(24):2894–2900, 2008. 8

[175] D.T. Ross, U. Scherf, M.B. Eisen, C.M. Perou, C. Rees, P. Spellman, V. Iyer, S.S. Jeffrey, M. Van de Rijn, M. Waltham, A. Pergamenschikov, J.C.F. Lee, D. Lashkari, D. Shalon, T.G. Myers, J.N. Weinstein, D. Botstein, and P.D. Brown. Systematic variation in gene expression patterns in human cancer cell lines. *Nature genetics*, 24(3):227–235, 2000. 75

[176] Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007. 72

[177] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975. 22

[178] A. Sandelin, W. Alkema, P. Engström, W.W. Wasserman, and B. Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic acids research*, 32(suppl 1):D91–D94, 2004. 12

[179] R.S. Savage, Z. Ghahramani, J.E. Griffin, J. Bernard, and D.L. Wild. Discovering transcriptional modules by bayesian data integration. *Bioinformatics*, 26(12):i158–i167, 2010. 4

[180] M. Schena, D. Shalon, R.W. Davis, and P.O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270(5235):467, 1995. 11

[181] U. Scherf, D.T. Ross, M. Waltham, L.H. Smith, J.K. Lee, L. Tanabe, K.W. Kohn, W.C. Reinhold, T.G. Myers, D.T. Andrews, D.A. Scudiero, M.B. Eisen, E.A. Sausville, Y. Pommier, D. Botstein, P.O. Brown, and J.N. Weinstein. A gene expression database for the molecular pharmacology of cancer. *nature genetics*, 24(3):236–244, 2000. 75, 78

[182] C. Schifanella, M.L. Sapino, and K.S. Candan. On context-aware co-clustering with metadata support. *Journal of Intelligent Information Systems*, pages 1–31, 2011. 32

[183] RL Schilsky, MJ Ratain, EE Vokes, NJ Vogelzang, J Anderson, and BA. Peterson. Laboratory and clinical studies of biochemical modulation by hydroxyurea. *Semin Oncol*, 19(9):84–89, 1992. 77

[184] T. Schlitt and A. Brazma. Current approaches to gene regulatory network modelling. *BMCBioinformatics*, 8, 2007. 48

[185] Saccharomyxes Genome Database (SGD). Saccharomyces Phenotype Terms. http://www.yeastgenome.org/cache/PhenotypeTree.html. 37

[186] SGD Project. Saccharomyces Genome Database. http://www.yeastgenome.org/. 13

[187] P. Shannon, A. Markiel, O. Ozier, N.S. Baliga, J.T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003. 44

[188] R. Shen, A. Chinnaiyan, and D. Ghosh. Pathway analysis reveals functional convergence of gene expression profiles in breast cancer. *BMC medical genomics*, 1(1):28, 2008. 73

[189] R.H. Shoemaker. The NCI60 human tumour cell line anticancer drug screen. *Nature Reviews Cancer*, 6(10):813–823, 2006. 71

[190] G. Smyth. limma: Linear Models for Microarray Data. In M. Gail, K. Krickeberg, J. Samet, A. Tsiatis, W. Wong, Robert Gentleman, Vincent J. Carey, Wolfgang Huber, Rafael A. Irizarry, and Sandrine Dudoit, editors, *Bioinformatics and Computational Biology solutions using R and Bioconductor*, Statistics for biology and health, pages 397–420. Springer New York, 2005. 11

[191] O.D. Sparkman and O.D. Sparkman. *Mass Spectrometry: Desk Reference*. Global View Publ, 2006. 12, 60

[192] V.J. Starai, H. Takahashi, J.D. Boeke, and J.C. Escalante-Semerena. Short-chain fatty acid activation by acyl-coenzyme A synthetases requires SIR2 protein function in Salmonella enterica and Saccharomyces cerevisiae. *Genetics*, 163(2):545–555, 2003. 42

[193] C. Stark, B.J. Breitkreutz, A. Chatr-Aryamontri, L. Boucher, R. Oughtred, M.S. Livstone, J. Nixon, K. Van Auken, X. Wang, X. Shi, T. Reguly, J.M. Rust, A. Winter, K. Dolinski, and M. Tyers. The biogrid interaction database: 2011 update. *Nucleic acids research*, 39(suppl 1):D698–D704, 2011. 12, 64

[194] J.E. Staunton, D.K. Slonim, H.A. Coller, P. Tamayo, M.J. Angelo, J. Park, U. Scherf, J.K. Lee, W.O. Reinhold, J.N. Weinstein, J.P. Mesirov, E.S. Lander, and T.R. Golub. Chemosensitivity prediction by transcriptional profiling. *Proceedings of the National Academy of Sciences*, 98(19):10787, 2001. 72

[195] D. Steinhauser, B.H. Junker, A. Luedemann, J. Selbig, and J. Kopka. Hypothesis-driven approach to predict transcriptional units from gene expression data. *Bioinformatics*, 20(12):1928–1939, 2004. 32

[196] J. Stelling, U. Sauer, Z. Szallasi, F.J. Doyle, and J. Doyle. Robustness of cellular functions. *Cell*, 118(6):675–685, 2004. 10

[197] G. Stolovitzky, R.J. Prill, and A. Califano. Lessons from the DREAM2 Challenges. *Annals of the New York Academy of Sciences*, 1158(1):159–195, 2009. 54, 57, 65

[198] J.M. Stuart, E. Segal, D. Koller, and S.K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249, 2003. 48

[199] D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, A. Roth, P. Minguez, T. Doerks, M. Stark, J. Muller, P. Bork, L.J. Jensen, and C. von Mering. The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic acids research*, 39(suppl 1):D561–D568, 2011. 12

[200] Y. Tamada, S. Kim, H. Bannai, S. Imoto, K. Tashiro, S. Kuhara, and S. Miyano. Estimating gene networks from gene expression data by combining Bayesian network model with promoter element detection. *Bioinformatics*, 19(Suppl. 2):ii227–ii236, 2003. 4

[201] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18(suppl 1):S136–S144, 2002. 32

[202] I.W. Taylor, R. Linding, D. Warde-Farley, Y. Liu, C. Pesquita, D. Faria, S. Bull, T. Pawson, Q. Morris, and J.L. Wrana. Dynamic modularity in protein interaction networks predicts breast cancer outcome. *Nature biotechnology*, 27(2):199–204, 2009. 72

[203] R. Thomas, D. Thieffry, and M. Kaufman. Dynamical behaviour of biological regulatory networks—I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bulletin of mathematical biology*, 57(2):247–276, 1995. 50

[204] V. Trevino and F. Falciani. GALGO: an R package for multivariate variable selection using genetic algorithms. *Bioinformatics*, 22(9):1154–1156, 2006. 75

[205] M.H.V. Van Regenmortel. Reductionism and complexity in molecular biology. *EMBO reports*, 5(11):1016, 2004. 3

[206] M. Vilela and G. Danuser. What's wrong with correlative experiments? *Nature Cell Biology*, 13(9):1011–1011, 2011. 60

[207] R. Viner, T. Zhang, S. Peterman, V. Zabrouskov, and T.F. Scientific. Advantages of the LTQ Orbitrap for protein identification in complex digests. *Thermo Scientific Application Note AN*, 386, 2007. 12, 60

[208] A. Visconti, F. Cordero, M. Botta, and R. A. Calogero. Gene Ontology rewritten for computing gene functional similarity. In *In Proceedings of the Fourth International Conferences on Complex, Intelligent and Software Intensive Systems, February 15-18 2010, IEEE Computer Society Press*, pages 694–699, 2010. 18

[209] A. Visconti, F Cordero, D. Ienco, and Pensa R. G. Coclustering under Gene Ontology Derived Constraints for Pathway Identification. In *Biological Knowledge Discovery Handbook: Preprocessing, Mining and Postprocessing of Biological Data*, page To appear. Wiley, 2012. 33

[210] A. Visconti, R. Esposito, and F. Cordero. Restructuring the gene ontology to emphasise regulative pathways and to improve gene similarity queries. *International Journal of Computational Biology and Drug Design*, 4(3):220–238, 2011. 24

[211] C. Vogel, G.M. Silva, and E.M. Marcotte. Protein expression regulation under oxidative stress. *Molecular & Cellular Proteomics*, 10(12), 2011. 64

[212] E.O. Voit. *Computational analysis of biochemical systems: a practical guide for biochemists and molecular biologists*. Cambridge Univ Press, 2000. 50

[213] J.Z. Wang, Z. Du, R. Payattakool, P.S. Yu, and C. Chen. A new method to measure the semantic similarity of GO terms. *Bioinformatics*, 23:1274–1281, 2007. 18

[214] Y.C. Wang and B.S. Chen. Integrated cellular network of transcription regulations and protein-protein interactions. *BMC systems biology*, 4(1):20, 2010. 60

[215] A.V. Werhli and D. Husmeier. Reverse engineering gene regulatory networks with Bayesian networks from expression data combined with multiple sources of biological prior knowledge. In *11th Annual International Conference on Research in Computational Molecular Biology (RECOMB 2007)*, 2007. 49

[216] M.L. Whitfield, G. Sherlock, A.J. Saldanha, J.I. Murray, C.A. Ball, K.E. Alexander, J.C. Matese, C.M. Perou, M.M. Hurt, P.O. Brown, and D. Botstein. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Molecular biology of the cell*, 13(6):1977–2000, 2002. 4

[217] E. Wingender, X. Chen, R. Hehl, H. Karas, I. Liebich, V. Matys, T. Meinhardt, M. Prüß, I. Reuter, and F. Schacherer. TRANSFAC: an integrated system for gene expression regulation. *Nucleic acids research*, 28(1):316–319, 2000. 12

[218] C.T. Workman and G.D. Stormo. ANN-Spec: a method for discovering transcription factor binding sites with improved specificity. In *Pac Symp Biocomput*, volume 2000, pages 467–478, 2000. 12

[219] Y. Yamaguchi, T. Narita, N. Inukai, T. Wada, and H. Handa. Spt genes: key players in the regulation of transcription, chromatin structure and other cellular processes. *Journal of biochemistry*, 129(2):185–191, 2001. 43

[220] E. Yeger-Lotem and H. Margalit. Detection of regulatory circuits by integrating the cellular networks of protein–protein interactions and transcription regulation. *Nucleic acids research*, 31(20):6053–6061, 2003. 60

[221] J. Yu, V.A. Smith, P.P. Wang, A.J. Hartemink, and E.D. Jarvis. Advances to bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, 20(18):3594–3603, 2004. 49

[222] N. Yu, J. Seo, K. Rho, Y. Jang, J. Park, W.K. Kim, and S. Lee. hiPathDB: a human-integrated pathway database with facile visualization. *Nucleic Acids Research*, 40(D1):D797–D802, 2012. 75

[223] A. Zelezniak, T.H. Pers, S. Soares, M.E. Patti, and K.R. Patil. Metabolic network topology reveals transcriptional regulatory signatures of type 2 diabetes. *PLoS computational biology*, 6(4):e1000729, 2010. 4

[224] Y. Zhang, J. Xuan, G. Benildo, R. Clarke, and H.W. Ressom. Reconstruction of Gene Regulatory Modules in Cancer Cell Cycle by Multi-Source Data Integration. *PloS one*, 5(4):e10268, 2010. 4

[225] Q. Zheng and X.J. Wang. GOEAST: a web-based software toolkit for Gene Ontology enrichment analysis. *Nucleic acids research*, 36(suppl 2):W358–W363, 2008. 25

[226] H. Zhou and F. Winston. NRG1 is required for glucose repression of the SUC2 and GAL genes of Saccharomyces cerevisiae. *BMC genetics*, 2(1):5, 2001. 43

[227] J. Zhu, Y. Chen, A. S. Leonardson, K. Wang, J. R. Lamb, V. Emilsson, and E. E. Schadt. Characterizing dynamic changes in the human blood transcriptional network. *PLoS computational biology*, 6(2):e1000671, February 2010. 60

[228] P. Zoppoli, S. Morganella, and M. Ceccarelli. TimeDelay-ARACNE: Reverse engineering of gene networks from time-course data by an information theoretic approach. *Bmc Bioinformatics*, 11(1):154, 2010. 65

[229] C. Zou and J. Feng. Granger causality vs. dynamic bayesian network inference: a comparative study. *BMC bioinformatics*, 10(1):122, 2009. 60

[230] C. Zou, C. Ladroue, S. Guo, and J. Feng. Identifying interactions in the time and frequency domains in local and global networks - A Granger Causality Approach. *BMC bioinformatics*, 11(1):337, January 2010. 60, 66