
# UNIQUE NETWORKS: A METHOD TO IDENTIFY DISEASE-SPECIFIC REGULATORY NETWORKS FROM MICROARRAY DATA

A thesis submitted for the degree of *Doctor of Philosophy*

by

Valeria Bo

Department of Computer Science

December 2014

# Abstract

The survival of any organism is determined by the mechanisms triggered in response to the inputs received. Underlying mechanisms are described by graphical networks that can be inferred from different types of data such as microarrays. Deriving robust and reliable networks can be complicated due to the microarray structure of the data characterized by a discrepancy between the number of genes and samples of several orders of magnitude, bias and noise. Researchers overcome this problem by integrating independent data together and deriving the common mechanisms through *consensus network* analysis.
Different conditions generate different inputs to the organism which reacts triggering different mechanisms with similarities and differences. A lot of effort has been spent into identifying the commonalities under different conditions. Highlighting similarities may overshadow the differences which often identify the main characteristics of the triggered mechanisms. In this thesis we introduce the concept of study-specific mechanism. We develop a pipeline to semi-automatically identify study-specific networks called *unique-networks* through a combination of consensus approach, graphical similarities and network analysis.

The main pipeline called UNIP (Unique Networks Identification Pipeline) takes a set of independent studies, builds gene regulatory networks for each of them, calculates an adaptation of the sensitivity measure based on the networks graphical similarities, applies clustering to group the studies who generate the most similar networks into *study-clusters* and derives the consensus networks. Once each study-cluster is associated with a consensus-network, we identify the links that appear only in the consensus network under consideration but not in the others (unique-connections). Considering the genes involved in the unique-connections we build Bayesian networks to derive the *unique-networks*. Finally, we exploit the inference tool to calculate each gene prediction-accuracy across all studies to further refine the unique-networks. Biological validation through different software and the literature are explored to validate our method.
UNIP is first applied to a set of synthetic data perturbed with different levels of noise to study

the performance and verify its reliability. Then, wheat under stress conditions and different types of cancer are explored. Finally, we develop a user-friendly interface to combine the set of studies by using AND and NOT logic operators.

Based on the findings, UNIP is a robust and reliable method to analyse large sets of transcriptomic data. It easily detects the main complex relationships between transcriptional expression of genes specific for different conditions and also highlights structures and nodes that could be potential targets for further research.

# Acknowledgements

First, I would like to thank my supervisor Allan Tucker for his constructive supervision, encouragement and patience. Without his guidance, in both work and life, this work would have never been completed. It has been a privilege and a pleasure to work with him.

I would also like to thank my second supervisor Stephen Swift for his invaluable advice and support.

I am grateful to the members of the Rothamsted Research Centre Mansoor Saqi and Artem Lysenko. They have been great partners in collaboration.

Thanks to Dimah Habash for her help using MapMan and to Tanya Curtis for the interesting discussions and the collaboration on the wheat data analysis.

To my colleagues (past and present) Neda, Cici, Chandrika, Helga, Mahir, Ovidiu and many others for keeping the department a pleasure to work in.

Thanks to Shav, Valentina and Valentina, Jessica, Massimo, Nicola and Gian Paolo for these amazing 3 years.

A big thanks to Pam, for always being there for me.

Last but not least a special thanks to my family for the love and incredible support over all these years.

# Publications

The following publications have resulted from the research presented in this thesis:

- Bo V, Tucker A. Integrating Gene Regulatory Networks to identify cancer-specific genes. Submitted to AMIA - Joint Summit 2015. Accepted.
- Bo V, Curtis T, Lysenko A, Saqi M, Swift S, Tucker A. Discovering Study-Specific Gene Regulatory Networks. PloS one, 2014
- Bo V, Lysenko A, Saqi M, Habash D, Tucker A. Integrating Multiple Studies of Wheat Microarray Data to Identify Treatment-Specific Regulatory Networks. Advances in Intelligent Data Analysis XII, 2013
- Bo V, Lysenko A, Saqi M, Tucker A. Exploring the variation in gene regulatory networks: a study in wheat. European Conference on Computational Biology (ECCB), 2012

# Contents

**List of Figures** **x**

**List of Tables** **xiv**

**Glossary** **xvi**

**1 Introduction** **1**

1.1 Motivation . . . . . 1

1.2 Thesis contributions . . . . . 4

1.3 Thesis outline . . . . . 5

**2 Background** **6**

2.1 Gene Expression Analysis . . . . . 6

2.2 Microarrays . . . . . 8

2.3 Analysis of microarray data . . . . . 12

2.4 Gene selection . . . . . 14

2.5 Gene Regulatory Networks . . . . . 16

2.5.1 Boolean Networks . . . . . 18

2.5.2 Correlation Networks . . . . . 18

2.5.3 Bayesian Networks . . . . . 19

2.6 Identifying Gene Regulatory Networks structure . . . . . 20

2.7 Module Analysis . . . . . 21

2.8 Construction of robust regulatory networks . . . . . 22

2.9 Incorporating expertise . . . . . 24

2.10 Integration of multiple data . . . . . 25

2.11 Conclusions . . . . . 26

**3 Key Concepts** **27**

3.1 Co-expression . . . 27
3.2 Clustering . . . 29
3.3 Scale free vs Random graphs . . . 31
3.4 Weighted Gene Correlation Network Analysis . . . 32
3.4.1 WGCNA networks applied to wheat . . . 34
3.5 Modelling GRNs using Glasso . . . 38
3.5.1 Inverse covariance and partial correlation . . . 38
3.5.2 Lasso . . . 38
3.5.3 Graphical lasso . . . 39
3.5.4 Glasso implementation in R . . . 40
3.5.5 Glasso networks applied to wheat . . . 42
3.6 Modelling GRNs using Bayesian Networks . . . 46
3.6.1 Model selection . . . 47
3.6.2 D-separation, Markov property and conditional independence . . . 49
3.6.3 Bayesian Network Inference Algorithms . . . 50
3.6.4 Prediction . . . 51
3.6.5 Application to gene expression profiles . . . 52
3.7 Conclusion . . . 52

**4 Analysis of synthetic data** **53**

4.1 Introduction . . . 53
4.2 Methods . . . 54
4.2.1 Single study glasso network . . . 56
4.2.2 Graph similarity . . . 56
4.2.3 Consensus networks and unique-connections . . . 56
4.2.4 Unique Networks . . . 57
4.2.5 Bayesian unique-networks . . . 58
4.2.6 Prediction accuracy . . . 59
4.2.7 Biological support . . . 59
4.2.8 Biclustering . . . 59
4.3 Data structure . . . 60
4.4 Results on simulated data . . . 66
4.4.1 Unique-networks and intermediate results . . . 67
4.4.2 Prediction accuracy and final results . . . 69

4.5 Comparison with Biclustering . . . 74
4.6 Discussion . . . 74

**5 Analysis of Real Data** **77**
5.1 Introduction . . . 77
5.2 Pipeline adaptation to real datasets . . . 78
5.2.1 Variables selection . . . 78
5.2.2 Consensus and unique networks . . . 79
5.2.3 Biological support . . . 79
5.3 Wheat results . . . 80
5.4 Wheat comparison . . . 89
5.4.1 Comparison with Bicluster . . . 89
5.4.2 Comparison with WGCNA . . . 90
5.5 Biological validation - literature . . . 90
5.6 Fusarium results . . . 94
5.6.1 Comparison with WGCNA: . . . 95
5.7 Discussion . . . 98

**6 Cancer data and logic application** **100**
6.1 Introduction . . . 100
6.2 Method description . . . 101
6.2.1 Variable selection . . . 102
6.2.2 Genecards validation and probability score . . . 103
6.2.3 Logic and GUI . . . 103
6.3 Results and applications . . . 104
6.3.1 Identification of unique-genes through GeneCards . . . 109
6.3.2 Gene-a-la-carte and source selection . . . 110
6.4 Interface description - Logic . . . 111
6.5 Discussion . . . 115

**7 Conclusions** **116**
7.1 Thesis contributions . . . 116
7.1.1 Unique-networks . . . 116
7.1.2 Unique Network Discovery Pipeline . . . 116
7.1.3 Application to several datasets . . . 117
7.1.4 Unique genes and probability score . . . 117

7.1.5 Logic Application . . . . . 118
7.2 Limitations . . . . . 118
7.3 Further work . . . . . 119
7.3.1 Next Generation Sequencing . . . . . 119
7.3.2 Application to different kind of data . . . . . 120
7.3.3 Static vs dynamic data . . . . . 120
7.3.4 Improvement of the Graphical User Interface . . . . . 120

**A Additional tables and results** **121**
A.1 Chapter 5 additional tables . . . . . 121
A.2 Chapter 6 additional tables . . . . . 128

**References** **147**

# List of Figures

1.1 A simple gene regulatory network model (Steele 2010) . . . . . 2

2.1 The '*central dogma*' of gene expression, enunciated by F. Crick in 1958, summarized in its essential steps. The process involves a transcription phase, which transcribe one single DNA strand into messanger RNA, and a translation phase, which translate the mRNA strand into a polypeptide chain. This image was taken from Steiner (2014). . . . . 7

2.2 The figure shows a graphical representation of the steps required for the microarray technique. Image taken from Grigoryev (2011). . . . . 9

3.1 Contingency table . . . . . 29

3.2 Example of how to calculate true positives, false positives and false negatives between two networks. . . . . 30

3.3 Scale-free plot. The figure on the left hand side show the distribution of the connectivity (k), while the one on the right represent the relation between $k$ and $p(k)$ in logarithmic scale highlighting that the slope is close to -1. . . . . 36

3.4 Scale independence. Each plot shows the variation of $R^2$ for different values of $\beta$ (power) for each single study under analysis. The red horizontal line identifies the threshold set at 0.8. Above which $R^2$ satisfies the scale-free criteria therefore the corresponding value of $\beta$ can be used in the soft-thresholding procedure. . . . 37

3.5 Network built with glasso and parameter $\rho = 0.005$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names. . . . . 43

3.6 Network built with glasso and parameter $\rho = 0.010$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names. . . . . 44

3.7 Network built with glasso and parameter $\rho = 0.020$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names. . . . . 45
3.8 The figure shows the DAG of the Bayesian network with 4 random discrete valued gene variables and the conditional probability tables related to each node in the DAG. Note that G1=Gene1, G2=Gene2, G3=Gene3 and G4=Gene4 (Steele 2010). 47
3.9 The figure shows the probability of Gene 2 and Gene 3 being *on* or $off$ when it is observed that Gene 4 = *on*. Note that Gene 1 = G1, Gene 2 = G2, Gene 3 = G3 and Gene 4 = G4. . . . . 51

4.1 Pipeline overview. A schematic overview of the sequence of steps forming the pipeline. . . . . 55
4.2 Example of unique-connections construction approach. Given three study-clusters each with a corresponding consensus study-cluster, the unique-connections for study-cluster 1 are the set of connections that are unique for that consensus study-network and do not appear in consensus study-networks 2 and 3. Dashed connections indicate the connections that each network has in common with consensus study-network 1 and therefore will not be included in the unique-connections set. Genes not involved in any unique-connections will also be discarded (genes crossed out) . . . . . 58
4.3 Original structure of the Alarm network. . . . . 62
4.4 Original structure of the Insurance network. . . . . 63
4.5 Original structure of the Child network. . . . . 64
4.6 *Big matrix* constructed from the datasets generated from the three networks and six randomly generated datasets which represent the noise. The shaded regions indicate the non-noisy datasets generated from Alarm, Insurance and Child networks (respectively A, I and C in the figure). While R indicates random values (noise). . . . . 65
4.7 Study-clusters for the original data (0% of noise), 10%, 50% and 90% of noise. The studies' number highlighted with the same colour belong to the same cluster. 66

4.8 TPs and FPs vs noise **before** calculating the correct-prediction. The figures show the evolution of TPs and FPs vs noise in terms of nodes (variables involved in the discovered subnetworks) and connections between nodes. The green dotted lines indicate what is the original number of nodes. These are the partial results, prior to the filtering of the informative nodes based on the intra cluster correct-prediction accuracy (which are shown in Figure 4.9). . . . . 68

4.9 Intra cluster correct-prediction for simulated data. The figure shows the boxplots of the intra cluster correct-prediction (calculated within the same cluster using cross-validation) for the simulated dataset in the case of 0% of noise. . . . . 71

4.10 Intra cluster correct-prediction distribution for 10, 50 and 90% perturbation. The figures show the histograms of the intra cluster correct-prediction (calculated within the same cluster using cross-validation) for the simulated dataset for different levels of noise. . . . . 72

4.11 TPs and FPs vs noise **after** calculating correct-prediction. The graphs show the number of TPs and FPs nodes and connections detected at different levels of noise. Threshold set to 0.6. The dotted lines at the top of the graphs indicates the number of nodes in the relative original network. . . . . 73

4.12 The figures show the group of **samples** and **variables** respectively obtained using the bicluster method QuestMotif (Murali & Kasif 2003). Each bar represents a sample-group indicated with a number on the x-axis. The different colours indicate to which original network the samples in the sample-group truly belong to. The y-axis indicates the number of samples in Figure a and the number of variables in Figure b. . . . . 75

5.1 Network 1. Unique-Network for wheat under stress-enriched conditions in cluster 1. The highlighted genes (black and grey) have a *intra* prediction accuracy higher than 0.6, meaning that they have been predicted correctly at least 60% of the times by the remaining genes inside the same study-cluster. The network shows one big path starting with a highly predicted stress related gene (black genes number 29) connected through one gene (47) to a two more stress related genes all directly connected to highly predictive genes and another stress-related path involving the genes 41-53-23-25. . . . . 83

5.2 Network 2. Unique-Network for wheat under stress-enriched conditions in cluster 2. Grey nodes indicate highly predictive (average correct-prediction level higher or equal to 0.6) genes. Black nodes highlight highly predictive and stress related genes. This network presents a high number of highly predicted genes, but only one that is highly predicted and stress-related. . . . . 84
5.3 Network 3. Unique-Network for wheat under non-stress conditions in cluster 3. This network is composed of multiple smaller sub-networks not immediately related to each other. Few genes of those involved are still stress-related (black nodes) and almost the total of the them present high prediction (grey nodes). . . 85
5.4 Boxplot intra clusters prediction. The boxplots in each figure represents the *intra* (internal) cluster prediction-accuracy for each gene where the line indicates the average *inter*-clusters (external) prediction-accuracy. . . . . 88
5.5 Boxplot intra vs inter clusters correct-prediction. . . . . 89
5.6 Unique-Network for *Fusarium* cluster 2,5,6,7,13. In this figure grey background indicates highly predictive genes (average correct-prediction equal or higher than 0.6). Despite the lack of different conditions in the dataset, as explained in the text, still about a 1/3 of the genes selected are highly predictive. . . . . 96
5.7 Intra vs inter clusters prediction for *Fusarium*. . . . . 97

6.1 Bayesian unique-network for breast cancer. . . . . 105
6.2 Bayesian unique-network for ovarian cancer. . . . . 106
6.3 Bayesian unique-network for medullary-breast cancer. . . . . 107
6.4 Bayesian unique-network for lung cancer. . . . . 108
6.5 Internal (intra) vs External (inter) prediction accuracy for each study averaged among all genes involved in the related unique-network. . . . . 109
6.6 Left hand-side panel of the Logic Application interface. The figure shows the three loading buttons and the AND and NOT boxes for the studies logic combination. This example shows the case where the user wants to visualize the unique-connections and the list of related genes that study 1 AND 4 have in common but do not appear in study 2. . . . . 113
6.7 Right hand-side of the Logic Application interface. The figure shows both tabs of the results panel placed side by side. The first shows the unique-connections network and the other the table containing the correspondence between genes number in the network and real names. . . . . 114

# List of Tables


3.1 Study numbers, labels, number of samples and descriptions of the wheat microarray dataset. . . . . 35

4.1 Simulation studies generated independently from the three networks in consideration. . . . . 66

5.1 Study numbers, labels, number of samples and descriptions of the wheat microarray dataset. . . . . 81

5.2 Wheat Unique-Networks(U-N) biological process functions from Gene Ontology as described in Lysenko et al. (2011). IC values greater than 3 are considered to be biologically informative. . . . . 87

5.3 Study numbers, labels, number of samples and descriptions of the *Fusarium* microarray dataset. . . . . 94

5.4 *Fusarium* unique networks biological process functions from Gene Ontology as described in Lysenko et al. (2011). IC values greater than 3 are considered to be biologically informative. . . . . 97

6.1 Cancer datasets identification code, description and samples number. . . . . 102

6.2 Cancer datasets identification code, description and the p-values obtained from the t-test. . . . . 104

6.3 Parameters values, z-score and p-value for each study. . . . . 109

6.4 List of the identified unique-genes in each study. . . . . 111

A.1 Correspondence of genes numbers and affymetrix names together with the functions indicated by Mapman in unique-network 1 (stress-enriched) for the wheat dataset. . . . . 123

A.2 Correspondence of genes number sand affymetrix names together with the functions indicated by Mapman in unique-network 2 (stress-enriched) for the wheat dataset. . . . . 125
A.3 Correspondence of genes number sand affymetrix names together with the functions indicated by Mapman in unique-network 3 (non-stress) for the wheat dataset.127
A.4 Correspondence of genes numbers, affymetrix names and symbols for the **breast** cancer dataset. . . . . 134
A.5 Correspondence of genes numbers, affymetrix names and symbols for the **ovarian** cancer dataset. . . . . 138
A.6 Correspondence of genes numbers, affymetrix names and symbols for the **medullary breast** cancer dataset. . . . . 142
A.7 Correspondence of genes numbers, affymetrix names and symbols for the **lung** cancer dataset. . . . . 146

# Glossary

- **Consensus-study network or Consensus network:** is the consensus network built for a study-cluster.
- **Sample:** indicates the measurements of all the genes when the organism is subjected to an experimental condition.
- **Study:** is the collection of samples measured under the same experimental conditions.
- **Study-cluster:** is the group of studies that present a similar network structure and therefore are cluster together by k-means algorithm.
- **Unique-connections:** list of edges that exist in the consensus-study network in consideration, but not in the other consensus-study networks.
- **Unique-genes:** list of genes involved in one unique-network but not in the others.
- **Unique-network:** given the consensus networks for all study-clusters, we first identify the unique-connections and considering only the genes involved in the unique-connections we build the Bayesian networks for each study-cluster. It represents the sub-network(s) that is specific for that study-cluster and does not appear in any of the others.

# Chapter 1

# Introduction

## 1.1 Motivation

Organisms of any level of complexity (from bacteria to mammalian) developed during evolution, a large set of internal mechanisms either for the normal functioning or in response to external or internal stimuli that differ from the normal activity. While many mechanisms, necessary for survival, carry on mostly unchanged under all conditions the organism is subjected to (e.g. cell metabolism), others are triggered or modified only when some event external or internal to the organism (environmental changes, stress, cancer, etc.) happens. Organisms' mechanisms, in general, involve large numbers of interactions between thousands of genes resulting in highly complex networks.

For the past decade bioinformaticians have focused their attention to discover the regulatory mechanisms that govern organisms. Despite the giant steps in the area still a lot of knowledge is hidden in the data waiting to be revealed.

Thanks to the constant improving of techniques, machine procedures and data storage more and more data are now publicly available either as microarray or as Next Generation Sequencing (NGS).

While next generation sequencing seems likely to completely replace microarrays in the near future, the large amount of these data available and its precious source of information is not to be wasted.

Along with the large increase of data, new computational tools have been developed to decrypt the information hidden in them. At present, a popular area of research is the understanding of the mechanisms underlying an organism generally achieved by the modelling of Gene Regulatory Networks (GRNs).

GRNs represent the underlying mechanisms of gene regulation in various cellular processes and describe how genes influence the activity of other genes. This is necessary to comprehend cells activity and furthermore to explore the functioning of diseases. An altered condition, in fact, can be detected from a change in the ordinary mechanism pattern. Building GRNs helps biologists in better understanding genetic conditions and identifying genes of particular interest for further experiments.

A simple example of a generic GRN is shown in Figure 1.1 (Steele 2010). The network clearly shows that the expression of Gene1 infuences the expression of both Gene2 and Gene3 by producing the transcription factor proteins that activate their expression. Then, the expression of Gene2 and Gene3 infuence the expression of Gene4 in the same way.

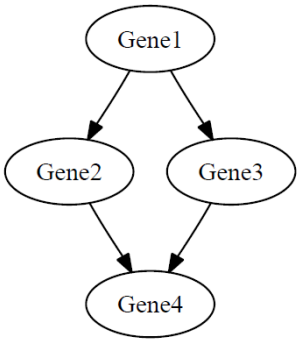



Figure 1.1: A simple gene regulatory network model (Steele 2010)

Publicly available databases contain an enormous amount of gene expression data for numerous organisms and across various experimental conditions waiting to be explored (Rustici et al. 2013, Geer et al. 2009). Genes expression measurements across one or a set of independent studies provide information about the underlying regulatory relationships between genes. Several methods have been developed over the years to infer GRNs from microarray data. Clustering techniques allows to group co-regulated genes to use as a basis for learning GRNs models. However, simple clusters are not able to reveal the more complex structure of the gene regulation process. Therefore, a group of more complex analysis techniques for reverse-engineering GRN models (build the model from the data) have been implemented for the task. This thesis focuses on two technique in particular, Glasso (Friedman et al. 2008) and Bayesian networks (Nielsen & Jensen 2009, Friedman et al. 2000).

Glasso goes beyond the simple pairwise correlations between genes. It estimates sparse graphs by deriving the inverse covariance matrix using the lasso penalty to make it as sparse as possible.

Bayesian networks are a popular and successful method, able to represent the network both qualitatively (with a network graph) and quantitatively (probability distributions that quantify the strength of influences and dependencies between nodes/variables in the network graph) and thus are relatively easy to interpret by non-technical people.

In the past decade, researchers have been focusing on what regulatory mechanisms different experimental conditions have in common i.e. given a set of different type of tumours and their GRN models researchers identify what are the gene regulatory mechanisms that the set of tumours have in common. This represents a valuable information when it comes to understanding tumours or other diseases. In fact, tumours affect different organs and have different levels of aggressiveness, but they still belong to the same generic class and therefore must have some commonalities that show in the GRNs. On the other hand highlighting the commonalities often overshadows the differences, what makes each disease unique and easier to detect and therefore to cure it. Hence, in this research we aim to discover the differences between set of studies. We develop a pipeline called UNIP (Unique Network Identification Pipeline) to semi-automatically identify mechanisms that are specific/unique for one or a set of studies.

The main aim of the research presented in here is to identify the study-specific mechanisms and the genes involved in them for a given set of conditions.
Previous researches in the literature focus on integrating data from a set of independent studies to infer more robust models and detect mechanisms that are common to multiple experimental conditions. In this work instead, we recognize the importance of shared mechanisms but we realise that identify what is specific of each experimental condition leads to a better a quicker diagnostic as well as to a cure. Hence, we introduce and develop the concept of *unique-network* through the implementation of a pipeline to semi-automatically identify study specific networks and genes.

The aim of the research presented in this thesis is the integration of a set of independent studies on the same organism to discover study-specific gene regulatory networks. Using a combination of cluster analysis, graphical similarities and prediction accuracy our method identifies reliable and robust (sub)networks *unique* for one or a set of conditions.
In this introductory chapter we fully explain the motivations, aims, and contributions of this thesis.

## 1.2 Thesis contributions

The main contributions of this thesis are:

- **A full formal definition of *unique-networks*.** First, the generic concept of *uniqueness* and its importance are explained, then a formal mathematical definition is derived in terms of graphical structure.
- **Development of an algorithm to generate unique-networks.** A set of algorithms are combined in a specific and justified sequence to read multiple microarray files and identify the related unique-networks.
- **Implementation of a pipeline for the discovery of study-specific gene regulatory networks.** We implement a sequence of steps involving gene selection, clustering technique and graph similarity measure.
- **Exploration of the performances of the pipeline on a synthetic dataset.** In order to analyse the robustness and reliability of the unique network identification pipeline developed in this work it has been considered necessary to first evaluate the pipeline performance using a dataset originated from a well-defined and synthetically created network.
- **Validation of the pipeline on multiple real datasets.** Application of the pipeline to a combination of wheat and cancer studies.
- **Identification of unique-genes.** Following the same line of thought that brought us to explore unique-networks we further develop a method to detect those genes involved uniquely in the condition under study.
- **Unique-genes validation using statistical score.** Measurement of the unique-genes significance trough the use of a statistical score.
- **Creation of a Graphical User Interface to perform different combination of studies using AND and NOT logic operators.** Finally, a basic application has been developed to ease the use of part of the process described by non-technical users.

## 1.3 Thesis outline

This thesis is organized as follows.

Chapter 2 explores the state of the literature and the gaps to fill. It first defines the concept of gene expression, then it explains the different microarray technologies used to record these data and the techniques to analyse it. It moves then to a comprehensive analysis of the algorithms developed in the literature to correctly process these data and reveal the information hidden in it.

Chapter 3 explores the state-of-the-art concepts used for this work. It focuses on standard techniques such as co-expression and clustering and later moves to investigate the most reliable and robust methods to build Gene Regulatory Networks (GRNs), some of which are later employed in this thesis.

Chapter 4 introduces the concept of unique-networks and describes, in details, the pipeline which is the primary focus of this thesis and then studies its performances when applied to a set of synthetic independent datasets perturbed with different levels of noise.

Chapter 5 illustrates the changes implemented to adapt the main pipeline to real world problems and explores the results using real datasets obtained under different conditions in wheat and *Fusarium*.

Chapter 6 describes how we apply the pipeline to another set of real data focusing on four different studies of cancer and develop a user-friendly interface to combine the studies using AND and NOT logic operators. Also, it explores the new concept of unique-genes and a method to integrate historical knowledge to detect the most informative uniquely-involved genes.

Finally, Chapter 7 summarizes our findings, identifies advantages and disadvantages and explores future improvements and developments.

# Chapter 2

# Background

This chapter reports the state of the literature regarding gene expression analysis. The first part, describes the different microarray techniques employed to collect gene expression data and explores the state-of-the-art algorithms used to read it. This is to gain an insight into the advantages and disadvantages of these techniques.
The second part of this chapter highlights the problems related to the analysis of gene expression and explores past and present studies that use data mining and machine learning techniques to get a better understanding of the gene regulatory mechanism. The main focus remains on the analysis using Gene Regulatory Networks (GRNs).
This chapter identifies gaps in the literature that we fill with this work.

## 2.1 Gene Expression Analysis

In 1958 F. Crick enunciated what is now called the *'central dogma'* of gene expression. This theorem explains how the information included in the nucleotide sequence of DNA (genes) inside a cell are translated into polypeptide chains (proteins). It determines the structure and capabilities of cells and organisms (Hartl & Jones 2009) and it is vital for their survival. The 'central dogma' is an extremely sophisticated mechanism involving several intricate steps. Although, for this thesis purposes, a simple and schematic view of the entire process is explained and shown in Figure 2.1.

The first step is called *transcription* and includes two phases, both happening inside the nucleus of the cell. To start with, the RNA-polymerase uses the nucleotide sequence of a segment of a single strand DNA as a template to create a complementary RNA strand. Immediately afterwards the RNA strand goes through some chemical modifications that return the messenger

RNA (mRNA). The next phase, called *translation* resides within a specialized organelle - the ribosome. In eukaryotes, mRNA molecules leave the nucleus and travel to the cytoplasm, where the ribosomes are, while in prokaryotic organisms this is not necessary. Through the ribosome, the nucleotide sequence of the mRNA is translated into a specific sequence of amino acids which generates a polypetide chain (protein).

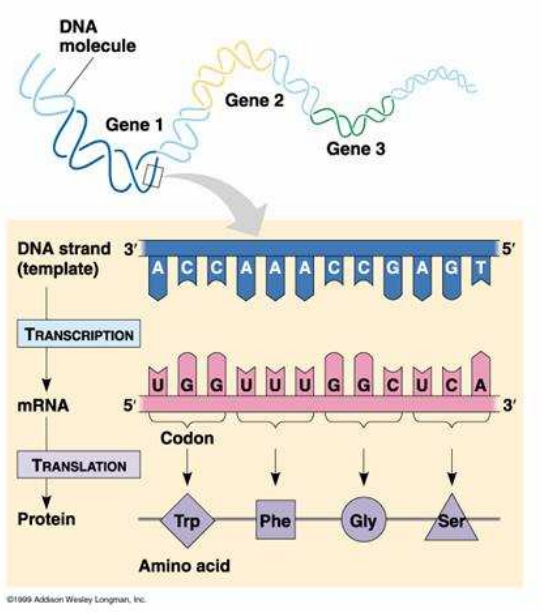



Figure 2.1: The '*central dogma*' of gene expression, enunciated by F. Crick in 1958, summarized in its essential steps. The process involves a transcription phase, which transcribe one single DNA strand into messanger RNA, and a translation phase, which translate the mRNA strand into a polypeptide chain. This image was taken from Steiner (2014).

This whole process is the phenotypic manifestation of one single or multiple genes and is also called *gene expression*.

Regulation of gene expression proved to be an essential process for the development of cells and organisms, therefore, it remains a central topic of research.

Gene expression activity, measured in terms of gene expression level (how much a gene is expressed), is regulated at the transcription step through either signals internal to the cell, according to cell type or stage in the cell cycle, or in response to external stimuli (Hartl & Jones

2009). To explore gene expression activity, the cell or organism needs to be first subjected to the experimental condition of interest and only then gene's expression level is measured. Each experimental condition is repeated multiple times to generate a collection of multiple samples related to each gene called *gene expression profile*. Two techniques known as *Microarray* (DeRisi et al. 1997) and *Next Generation Sequencing* (Shendure & Ji 2008) can be used to measure gene expression activity. Next Generation Sequencing is newer (introduced in the early/mid-2000s by the 454 Corporation) and, in certain cases, more appropriate (Morozova & Marra 2008). On the other hand, Microarray is less expensive and easier to analyse, it requires less laboratory analysis and it produces less data to process. Furthermore, researchers still feel more comfortable using microarray given the familiarity they have with it. Last but not least, while next generation sequencing will likely soon replace microarrays for expression analysis, the large amount of unexplored microarray data produced in the last two decades will be useful to researchers for many years to come. Considering all these factors, in this work we focus on using data obtained from microarray techniques. However the method proposed in here can be adapted, with some preprocessing steps, to the use of Next Generation Sequencing data.

## 2.2 Microarrays

Microarray analysis is a practical and time-saving laboratory tool that allows biologists to collect thousands of individual gene sequences in parallel to study gene expression and gene variation in any given cell type, time, set of conditions or treatments (*Scitable* 2014).
It was used for the first time to study the yeast genome in DeRisi et al. (1997), for two purposes:

1. investigate the temporal program of gene expression accompanying the metabolic shift from fermentation to respiration;

2. identify genes whose expression was affected by deletion of the transcriptional co-repressor TUP1 or over-expression of the transcriptional activator YAP1.

The statement 'Gene A is expressed' indicates that the segment of DNA encoding gene A is identified by a specific protein, called a transcription factor (TF) which triggers the transcription process (see Figure 2.1) and transcribe gene A into the corresponding mRNA strand, called transcript. The array of mRNA transcripts produced in a particular cell is called transcriptome. While the genome (the array of DNAs) is stable, the transcriptome is more sensitive and actively changes depending on many factors including cell's cycle stage and environmental conditions. Microarray, then, analyses changes in the transcriptome by measuring the abundance of mRNA molecules (expression level) present in the cells sample taken at that time. To do this, it

hybridises known DNA molecules (genes) with the complementary mRNA sequence extracted from the cell.

There are different types of microarrays that measure the gene expression levels in different ways, which we refer to as different platforms. The most common is the two-channel hybridised array which compares the gene expression levels in the same cell but in two different samples collected under different conditions. Usually one is the control and the other is the sample under specific conditions, but it can also represent the comparison of two different samples (sample A and sample B).

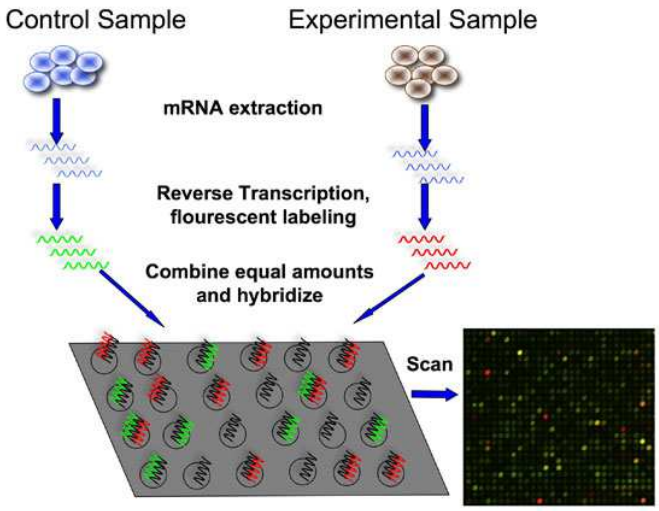



Figure 2.2: The figure shows a graphical representation of the steps required for the microarray technique. Image taken from Grigoryev (2011).

As shown in Figure 2.2, DNA molecules are printed in a glass or polymer microscope slide called DNA array, DNA chip or gene chip. Each attached molecule, referred to as spot or feature, encodes one single gene. A single DNA array may contain spots in the order of tens of thousands.

mRNA (transcriptome) is extracted from both samples, converted into cDNA and labelled with a different fluorescent dye based on which sample it comes from, on the same DNA array. Usually it is used red for one sample and green for the other. mRNA sample will hybridize to the complementary DNA segment (cDNA) previously attached to the spots on the array. Then, samples are washed away to allow only those mRNA segments that strongly paired strands will have enough hybridization strength to remain attached to the DNA array. After the washing-off a laser is used to determine the amount of fluorescence emitted by the dye-labelled mRNA at each spot. The total strength of the signal depends on the number of sample sequences bound

to the sequences in the spot. In the case of the two-channels hybridised array the fluorescence is measured twice (one for each sample). The emitted colour will be green if sample A is present, red if sample B is, yellow if both are, and black (no fluorescence) if neither of them are present. The identity of the gene is known by its position on the array.
Different microarray platforms use the same principle of complementary DNA / mRNA but the techniques to reveal the expression level may vary.

In single-channel arrays (van Bakel & Holstege 2007) (e.g. Affymetrix 'GeneChip' and Illumina 'Bead Chip') each sample is collected and labelled with only one colour on separate DNA arrays, consequently the value measured is the absolute expression value. Comparisons between different experimental conditions are done similarly as in the two-channel by comparing the signals obtained from each microarray. Clearly then, it is necessary to collect multiple samples from different experiments to compare the expression levels under different conditions. The single channel array, obviously, requires as many hybridizations as many samples we need to compare, but an anomalous sample does not affect the other samples. Also, it allows an easier comparison of DNA arrays from different studies as long as the batch effect (technical variation) is well handled. Therefore, when, as in our case, different experimental studies are compared, the single-channel array is preferred.

Whatever technique is used (two or one -channel), the following steps are carried out in the image processing: the result of hybridization is a DNA array (or multiple DNA arrays) that needs to be read. Most microarray scanners provide a software which will scan the array and extract the fluorescence intensities for each spot in it (Causton et al. 2009). First, to identify the spots on the array, avoiding artefacts or contaminants on the slides (e.g. scratches or dust), the software applies a process called *gridding*. The *gridding* requires the user to identify the approximate locations of subgrids which are used as reference points to place the grid. Then, to improve the grid placement, the centre-of-mass for each spot is calculated and the grid position arranged.

After the spot locations have been identified, the expression levels need to be inferred based on the spot fuorescence intensities (Quackenbush 2001). The built-in software usually returns a set of statistics that represent the spot such as mean, median and intensity of the spot. In the case of one-channel array, a common measure, called background-subtracted median, that consists of subtracting the median of the spot intensity with the median of the background is returned.
In the two-channel array, instead, we want to capture the relative change in a gene between two conditions. Therefore, the ratio of the intensity in the first sample over the intensity in the second sample is calculated. The ratio is a straightforward way to measure changes in expression,

as those genes that do not have a change in their expression between the two conditions will have a ratio of 1. However, if a gene has a two-fold increase in expression in the query sample compared to the reference sample, the expression ratio will be 2. But, if a gene has a two-fold decrease in expression, the expression ratio will be 0.5. Then a logarithmic transformation at base 2 is applied to reflect the right scale. These measurements are called intensity log ratios or expression levels.

Finally, microarray data needs to be adjusted for systematic variation (variation in the technology) so that measurements from different samples can be directly compared. The most common and simple method is to apply the scale normalisation, where the data range is adjusted by a constant factor across all spots. This is a simple scaling procedure that consists of subtract a normalization factor L from all the log ratio data

$$M_i' = M_i - L$$

where $M_i$ is the log ratio of the ith gene, and $M_i'$ is the normalised log ratio. Other more complex methods for normalisation can be applied such as linear regression, lowess normalization (logically weighted linear regression) (Cleveland 1979), loess normalization (a generalization of lowess), and so on.
In the work presented in this thesis, microarray datasets from different studies are collected from online databases (e.g. Affymetrix). These datasets have been applied a preprocessing step consisting of Robust Multichip Average method (Irizarry et al. 2003) followed by redundancy adjusted Pearson correlation coefficient calculated according to the method described in Obayashi et al. (2011).

Apart from the many qualities, microarrays also have some important limitations.

- Microarray expression datasets often come from different microarray platforms which measurement units may vary inducing bias (Shi et al. 2006, Tan et al. 2003);
- Studies may come from different laboratories where data are collected with different measurement biases based on the different experimental conditions. Thus, variations across samples and different experiments induce biological and experimental noise respectively. The lack of reproducibility leads to a lack of reliability;
- Microarray datasets are composed of a very large amount of genes (in the order of thousands) and very few samples (in the order of tens or hundreds). This is usually refered to as *curse of dimensionality* (Bellman et al. 1961, Somorjai et al. 2003) which makes it very difficult to identify reliable regulatory interactions.

In 2005 a new group of techniques have been developed (Margulies et al. 2005, Shendure et al. 2005) called Next Generation Sequencing (NGS) which proved to be more accurate. Although, microarrays still remain, in most cases, researchers' preference due to a less complicated sample preparation, minor costs especially for large number of samples and greater ease of use and analysis. For these reasons in this work we focus on using datasets derived from microarray platforms.

## 2.3 Analysis of microarray data

Once the microarray experiments have been collected and the results have been normalized, the next step is to explore the expression data to discover interesting patterns and relationships amongst genes and between experimental conditions (studies). For example, genes with similar behaviour or genes with interesting expression patterns (e.g. they are active in certain studies but not in others). The Microarray process is generally repeated several times, under the same experimental condition, to keep experimental bias under control. Each repetition is called sample and all the samples obtained under the same condition constitute a study. The microarray results are easily represented by a matrix containing the list of genes as rows and the samples as columns.

To perform useful and robust analysis it is often necessary to integrate several experimental conditions (study) to build the gene expression matrix, where each entry $M_{ij}$ is the expression level (intensity log ratio), for gene $i$ in the $j$th array (sample). The columns of the matrix represent different samples and different groups of samples represent distinct experimental conditions. Rows of the matrix represent genes expression profiles which show how the gene's expression changes across the studies. In some cases, if the samples are measured over time this shows how a gene's expression changes over time under a particular environmental condition. Otherwise, samples are simply split into different classes (e.g. healthy and diseased) and show the difference between the gene expression profiles across the different classes.

As explained in Section 2.2, microarrays are the major source of data for collecting gene expression levels in an organism, in certain conditions and at a specific time. The popularity of this technique is due to its ability to describe the expression of thousands of genes measured simultaneously under the experimental condition under analysis. The number of genes is exceptionally high (in the order of thousands) but the number of samples is very low with tens or at best hundreds of them. Depending on the complexity of the query mechanism the amount of samples are, very often, not enough to robustly learn a network model of the underlying behaviour. This computational issue is well known as the *curse of dimensionality*. Merging

together a broader collection of data has the potential to reduce the dimensionality gap between samples and variables and to produce gene regulatory models that are more robust and have greater confidence. Therefore, researchers increase the number of samples by bringing multiple studies together. However, in such situations bias and inter-platforms variabilities are likely to lead to spurious dependencies, resulting in models that significantly overfit the data.

Extensive effort has been directed toward assessing the combination of differential expression measurements across different platforms. Steele & Tucker (2008) bring together multiple datasets from different platforms to learn from and implement different methods to aggregate the knowledge between the datasets. Specifically, the authors developed two main approaches based on at which stage of the modelling process the aggregation is applied. In Pre-learning aggregation, first, data is scale normalized to allow combination and then a model is learnt from the combined dataset. The other method, instead, is called Post-learning and it splits in two different algorithms. Meta-Analysis learns a model from each dataset and then combines the models through statistical confidences attached to networks edges. Consensus Bayesian Networks identify consensus network features across all datasets. Despite the computational simplicity of the pre-learning aggregation method, simple normalization is not suitable for microarray because of the typical high level of noise caused by the use of different platforms. On the other hand, while Meta-analysis generalizes very well, Consensus Bayesian Network is too sensitive to poorly performing input networks.

In general two main techniques exist: meta-analysis and cross-platform. While cross-platform involves a direct comparison between expression measurements obtained from different platforms, meta-analysis combines the results of intra-platform comparisons at a higher level. Meta-analysis techniques are useful tools, but they can only combine the results of studies that have tested the same hypothesis or undergone the same experimental condition, and cannot easily be applied to investigate new hypotheses from existing data. An extensive and detailed comparison of the main available techniques can be found in Rudy & Valafar (2011). The authors compare cross-platform normalization methods based on inter-platform concordance and on the consistency of gene lists obtained with transformed data. To measure the effectiveness of each method, they use adapted statistics based on scatter and ROC (Fawcett 2006) -like plots. Given the complexity of the problem, in this research only microarray data produced by the same platform are integrated.

## 2.4 Gene selection

Returning to the microarray dataset typical structure, the large number of features/genes expressed (in the order of thousands) combined with only few samples (in the order of tens) makes the analysis and comprehension of each gene's function(s) and mechanism(s) difficult and confusing. Furthermore, eliminating irrelevant or redundant genes will certainly improve the accuracy of classification or prediction (Tabus & Astola 2005). This forces researchers to reduce the number of variables in consideration using dimensionality reduction techniques. The overall goals of variable/gene selection (Saeys et al. 2007) are to:

- avoid overfitting (poor predictive performance due to overly complex model of the data),
- render following processing faster and computationally easier,
- help in understanding the mechanisms underlying the data.

Dimensionality reduction is a broad area of research with many applications. It is possible to distinguish two main categories:

- *Feature extraction*
- *Feature selection*

In *feature extraction* the data represented in a high dimensional space is transformed into a space of fewer dimensions that reproduce most of the variability of the original data set. One famous example of this technique is Principal Component Analysis (PCA)(Pearson 1901) which uses an orthogonal transformation to convert a set of correlated variables into a set of values of linearly uncorrelated variables called principal components.

*Feature selection*, instead, aims to select a subset of variables from the original dataset to investigate further. This category allows the reduction of the dimensionality without corrupting the original representation of the variables. It preserves the original structure of the data and simplifies the interpretability. Various approaches have also been developed according to unsupervised and supervised learning within the classification context. The methods can be organized in three categories:

- *Filter techniques*: look at the intrinsic properties of the data, calculates a feature relevance score and discard the features with a low score. Each variable is considered separately;
- *Wrapper methods*: include the model hypothesis search within the feature search. Several subsets of features are generated and each evaluated by training and testing a specific classification model;

- *Embedded techniques*: the search method is built into the classifier and can be seen as a search in the combined space of feature subsets and hypotheses.

A very large number of all these techniques have been developed in the last few decades returning a large pool of choices (Saeys et al. 2007, Moreau & Tranchevent 2012). The simplest techniques, to discover differentially expressed genes, are parametric methods based on ANOVA, a modification of the t-test (Fox & Dimmic 2006) and Bayesian frameworks (Baldi & Long 2001) or non-parametric methods (model free) such as Wilcoxon rank-sum test (Thomas et al. 2001) and between-within classes sum of squares (Dudoit et al. 2002).

Data analysis, especially in the case of big data, incurs two types of error: type I and type II. Type I error commonly associated with the number of false positives indicates that a given condition is present when it is not (a gene is found relevant but it is not). Type II error, on the other hand, is associated with false negatives and indicates that a given condition is not present when instead, it is (a discarded gene that is actually relevant). These two errors are extremely dangerous and can lead to erroneous results and discoveries. Gene selection algorithms want to minimize the number of false positives (type I error) and of false negatives (type II error). Both are explored in Dudoit et al. (2003). The chance of committing some Type I errors increases with the number of hypotheses tested. For example, a p-value of 0.01 for one gene among a list of several thousands is no longer a significant finding, in fact it is very likely that even such a small p-value will occur by chance under the null hypothesis when considering such a large set of genes as in microarray datasets. A popular solution to type I error is to keep under control the false discovery rate (FDR) (Benjamini & Hochberg 1995). Four FDR controlling procedures are described in Reiner et al. (2003).

More methods for gene selection and extraction are available on Bioconductor (Gentleman et al. 2004). Well-known algorithms are MMD (Weiliang et al. 2008) which proposes a Marginal Mixture Model that directly models the marginal distribution of transformed gene profiles in the GeneSelectMMD package (Morrow et al. 2012) or in the GeneSelector package (Slawski & Boulesteix 2009) which generates a list of ranked genes (based on a choice of 14 different methods) and then derives the final ranking by examining perturbed versions of the original data set, e.g. by leaving samples, swapping class labels, generating bootstrap replicates or adding noise. One popular technique is to apply a modification of Principal Component Analysis (PCA) as in Wang & Gehan (2005), where they explore a method in which they apply the PCA to determine the essential dimensionality and then returns the genes in the dataset that are the closest to the essential dimensionalities (principal components). Last but not least another increasingly popular technique is Gene Set Enrichment Analysis (GSEA) which focuses on gene

set. That is, groups of genes that share common biological function, chromosomal location or regulation (Subramanian et al. 2005) followed by Gene Set Variation Analysis a GSE method that estimates variation of pathway activity over a sample population in an unsupervised manner (Hänzelmann et al. 2013).

When it comes to selecting genes something that we want to do is avoid repetition, meaning selecting genes with the same or similar functions. Genes with similar functions still can behave and respond differently based on the experimental condition they are subjected to but they confound when it is necessary to reduce the dimensionality (number of variables). One idea is to identify groups of genes rather than single genes using clustering techniques and use one representative of the group as the selected gene. Most of the analyses commonly attempted are based on clustering algorithms which locate groups of genes with similar expression patterns over a set of experiments. These approaches are based on the well known concept of *guilt-by-association* (GBA) (Altshuler et al. 2000, Oliver 2000) which is a statistical rule of thumb that states that we can reliably predict the function of a gene or protein if its correlated genes or other proteins connected through protein-protein interaction share similar functions. Such analysis has proven to be useful in discovering genes that are co-regulated and/or have similar functions. Peer et al. (2001) focus on genome-wide expression profile of genetic mutant, providing a wide variety of measurements of cellular responses to perturbations, and uses clustering to group genes of similar functions. Furthermore, they discover inter-cluster interactions between weakly correlated genes and uncover finer intra-cluster structure among correlated genes. This procedure allows the identification of highly promising general hypothesis useful to biologists although it cannot recover all interactions. Despite the expectation towards this concept Gillis & Pavlidis (2012) disapprove the use of GBA for function prediction. The authors specifically explore the application of the GBA concept on gene networks. Given that networks commonly include a substantial number of false positive connections, it is a very serious problem to generalize the use of Gene Ontology (GO) terms (Ashburner et al. 2000) combined with the GBA principle to predict new genes' functions.

## 2.5 Gene Regulatory Networks

Gene expression array data can be used to:

1. Measure if one gene expresses differently under different conditions (control vs. treatment conditions);

2. Explore common functionalities, interactions, etc. between clusters of genes;

3. Infer the underlying regulatory regions and gene/protein networks (*gene regulatory networks*) responsible for an observed behaviour (Baldi & Long 2001).

Since the purpose of this thesis is to exploit gene expression data to infer study-specific regulatory relationships among sets of genes, we focus now on the description of gene regulatory networks, what is special about them and why they are so difficult to infer.

A Gene Regulatory Network (GRN) represents the collection of DNA segments in the cell and their interactions, which controls the abundance of gene-product (Karlebach & Shamir 2008). The outcome of gene expression is the production of proteins which can be categorized into structural proteins, enzymes and transcription factors (TFs). Structural proteins confer rigidity and flexibility to the different biological components, enzymes catalyze chemical reactions and TFs, as the name says, are factors that induce the transcription stage. These proteins are particularly interesting, in fact, they are produced by the gene expression but they also induce the process or inhibit it by binding to the promoter region at the start of the DNA sequence of that gene. Therefore, regulatory interaction framework goes both direction from genes to proteins and from proteins to genes. This interaction can be even more complex if the TF activates or represses the expression of the same gene/s from which it is produced.

Since a GRN is the representation of how genes interact together and TFs are regulation process inductors/inhibitors and genes products, we can represent how genes interact together through gene expression and the regulation process. For example, if gene B is activated by a protein (TF) that is produced by the activation of gene A, we can easily say that A influences B and we can represent it as A $\rightarrow$ B. Because TFs can regulate the expression of more than one gene and each gene can be regulated by more than one TF in combination or under different conditions, we can say that in regulatory networks each gene may interact with both TFs and produced genes.

Building GRN models to gain insight into gene regulation is an increasingly popular topic of research. Understanding the mechanism underlying gene expression helps biologists for multiple reasons:

1. Identify possible disruptions of gene expression in some cell,

2. Investigate gene regulation interactions in a much cheaper and time-saving technique than wet lab experiments,

3. Identify pathways that can be tested experimentally and which would have not been considered otherwise.

Using data to learn a model of a gene regulatory network is called *reverse engineering*. Several techniques have been developed over the years to derive GRNs from data through reverse engineering. Each resulting model with its pros and cons highlight different aspects of the mechanism under study. Among the many developed techniques the most popular are described in the following section.

### 2.5.1 Boolean Networks

Kauffman (1969) introduce the concept of *Boolean networks*. These networks are system of binary variables, each with two possible states of activity ('on' and 'off') and with a boolean function which determines the topology (connectivity) of the set of variables (nodes in the network).
Considering the system as a discrete time series, the state of the network at time $t + 1$ is determined by each variable state at time $t$ according to the corresponding boolean switching function. So, boolean networks are a particular kind of sequential dynamical systems, where time and states are discrete.
These networks are related to cellular automata (Wolfram 1983) which are defined with an homogenous topology, i.e. a single line of nodes, a square or hexagonal grid of nodes or an even higher-dimensional structure, with the difference that each variable (node) may have more than two possible states (and hence not be boolean).
Dynamical systems contain thousands or millions of variables each in a different state. Many cellular and biochemical process exhibit a sigmoidal (S-shaped) response which are often properly idealized by 'on-off' systems. The simplification to an 'on-off' switching system allows researchers to study such enormously complex systems whose problems are often intractable using continuous nonlinear differential equations (Kauffman 1993).

### 2.5.2 Correlation Networks

*Correlation networks* is a broad category that goes from the simple calculation of the correlation coefficient between variables to the well known Weighted Gene Co-expression Network Analysis (WGCNA) (Zhang et al. 2005). This is a data mining method, based on pairwise correlations between variables. It works very well with high dimensional data and has led to broad application of this technique to study biological networks. It allows the identification of modules (clusters), intramodular hubs and nodes belonging to that module, the relationships between

co-expression modules, and the comparison of the topology of different networks. WGCNA also works as a data reduction technique, as a clustering method (fuzzy clustering), as a feature selection method, as a framework for integrating complementary (genomic) data , and as a data exploratory technique. WGCNA incorporates traditional data exploratory techniques, but its intuitive language and analysis framework makes it more popular than standard analysis technique. Since it uses network methodology and can integrate different genomic data sets, it is used as a systems biology (or genetic) data analysis method. Furthermore, selecting intramodular hubs in consensus modules, makes WGCNA a good meta analysis techniques (a class of method to contrast and combine results from different studies to identify patterns among study results, differences, or other interesting relationships that may come to light in the context of multiple studies). A full description of WGCNA method is given in Chapter 3.

### 2.5.3 Bayesian Networks

A *Bayesian Network (BN)* (Nielsen & Jensen 2009, Friedman et al. 2000) is a probabilistic graphical model that represents a set of random variables and their conditional dependencies using a directed acyclic graph (DAG). It is a representation of a joint probability distribution. Formally, Bayesian networks consist of $G$, a DAG, whose nodes represent random variables $X_1, ..., X_n$ and $\theta$, a conditional distribution table for each variable, given its parents in $G$. Edges represent conditional dependencies, non-connected nodes represent variables that are conditionally independent of each other. These two components combined together specify a unique distribution on $X_1, ..., X_n$. The graph $G$, representing conditional independence assumptions, allows to decompose the joint distribution reducing the number of parameters. In fact, the graph $G$ encodes the Markov Assumption:

*Each variable $X_i$ is independent of its nondescendants, given its parents in $G$.*

which means that when we apply the chain rule of probabilities and properties of conditional independencies, the joint distribution that satisfies the Markov Assumption can be decomposed into the product form:

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i | \mathbf{Pa}^G(X_i))$$

where $\mathbf{Pa}^G(X_i)$ is the set of parents of $X_i$ in $G$.

BNs are a popular method for multiple reasons: they enable the combination of highly dissimilar types of data (i.e., numerical and categorical) into a common probabilistic framework, without unnecessary simplification; they easily cope with missing data; and they naturally

weight each information source according to its reliability. Furthermore, in contrast to black-box predictors BNs are readily interpretable as they represent relationships using conditional probability distributions (Jansen et al. 2003) and thanks to their structure they are easily interpretable by biologists.

## 2.6 Identifying Gene Regulatory Networks structure

The structure of gene regulatory networks captures the relationships between genes, including correlation. The knowledge of the correct structures of gene networks is very important for characterizing the complex roles of all individual genes and the relationships between the many systems in an organism.

Network reconstruction has largely focused on physical protein interactions and so represents only a subset of biologically important relations. Thus, Lee et al. (2004) construct a more accurate and extensive gene network by considering functional, rather than physical associations. Gene-gene linkages are probabilistic values representing functional coupling between genes. Only some of the links represent direct protein - protein interactions, the rest are associations not mediated by physical contact, such as regulatory, genetic, or metabolic coupling that represent functional constraints satisfied by the cell during the course of the experiments.

Meinshausen & Bühlmann (2006) and Shojaie & Michailidis (2010), more generically, try to estimate the skeleton of Direct Acyclic Graphs (DAGs) where the variables exhibit a natural ordering. They exploit graph theoretic properties of DAGs and reformulate the likelihood as a function of adjacency matrix of the graph. To estimate the adjacency matrix of high dimensional DAGs, they use both lasso and adaptive lasso penalties.

Scutari & Nagarajan (2011), instead, propose a statistically-motivated estimator for the confidence threshold minimizing the L1 norm between the cumulative distribution function of the observed confidence levels and the cumulative distribution function of the confidence levels of the unknown network structure describing the true dependence structure.

One more approach is described in Zhang X. et al. (2012) where a novel method PCA-CMI (Path Consistency Algorithm and Conditional Mutual Information) is proposed for inferring GRNs from gene expression data by taking into account the non-linear dependencies and sparse structure of GRNs. The algorithm is able to distinguish direct regulatory relationships from indirect ones.

An important issue is the one of measuring the structural sustainability of the networks. This is analysed in Mueller et al. (2011) in which the authors develop an R package called QuACN to infer gene regulatory networks from microarray data and classify them by using topological

network descriptors provided in the package.

## 2.7 Module Analysis

No matter what technique is used to build GRNs, the goal for gene expression analysis is to reveal the structure of the transcriptional regulation process. Friedman et al. (2000) introduce an approach for analysing gene expression patterns that uncovers properties of the transcriptional program by examining statistical properties of dependence and conditional independence in the data. The algorithm is compared to clustering techniques and is able to discover relationships, interactions between genes other than positive correlation, and finer intra-cluster structure. Cell's mechanisms represented through gene regulatory networks are often organized as modules interacting with each other, where modules are a group (cluster) of genes co-regulated to different conditions.

Segal et al. (2003) develop an algorithm that, given as input a large pre-compiled set of candidate regulatory genes for the corresponding organism and a gene expression dataset, searches simultaneously for a partition of genes into modules and for a regulation program for each module that explains the expression behaviour of genes within. A regulation program specifies the behaviour of the genes in one module as a function of the expression level of a small set of the regulators (Transcription Factors and Signal Transduction Molecules) called module's 'regulators'. The procedure gives as output a list of modules of co-regulated genes and associated regulation programs (regulators and the conditions under which regulation occurs).

From this procedure a new class of model is derived called *Module Networks* which explicitly partitions the variables into modules, so that the variables in each module share the same parents in the network and the same conditional probability distribution. This procedure, significantly reduces the complexity of the model space as well as the number of the parameters. These reductions lead to more robust estimation and better generalization on unseen data.

Genes with correlated expression changes, over many conditions, are likely to be involved in similar functions or cellular processes (derived from Guilt-by-Association); these genes often also share DNA sequence elements, providing evidence that they are regulated by common transcription factors. Ideker et al. (2002) introduce an approach for screening a molecular interaction network to identify active sub-networks which are connected regions of the network that show significant changes in expression over particular subsets of conditions. The method they present combines rigorous statistical measure for scoring subnetworks with a search algorithm for identifying subnetworks with high score. The subnetworks are identified by different conditions, thus

genes don't have to be co-regulated over all conditions in order to group together. Because they consider only the significance of change, the algorithm may cluster together strongly repressed gene with an induced one and some genes may not belong to any cluster.

Microarrays measure not only expression levels of target genes, but also levels of genes encoding regulators Transcription Factors (TFs) and Signalling Proteins (SPs). TFs are specific proteins that bind to regulatory sequences on the DNA of target gene and work together to ensure the correct amount of gene is being transcribed. The behaviour of TFs is controlled by the cell's environment through the action of signalling proteins (SPs). The combined network of Transcription Factors and Signalling Proteins forms a regulatory program controlling the expression of individual genes directly (by regulator TFs) and indirectly (by regulator SPs). Pe'er et al. (2006) exploit this by limiting the search to simple network structures in order to significantly reduce the space of possible networks, while highlighting the most relevant biological information. Only a small fraction of all potential regulators may, in fact, be active in a given data set. Only when a gene consistently scores high as a parent for many genes, we can believe it indicates a true signal. Since false positives are significantly more costly than false negatives, finding a robust set of key regulators whom are most strongly supported by the data is a more important goal then discovering their complete set of targets. Furthermore, simple networks result in successfully reconstructing biologically correct regulatory relations in more complex organisms.

Because of the high number of variables and the complexity of some organisms, sometimes it is important to focus not necessarily on the entire mechanism underlying the gene expression data, but simply on some subsets and relative subnetworks. Sachs et al. (2009) describe an approach to scaling up the number of variables that can be considered for structure learning. The algorithm starts with a set of preliminary experiments to determine which subset may be useful. This subset is called a Markov Neighbourhood and is detected for each variable. It consists of a variable's parents, children, and other co-parents of its children.

## 2.8 Construction of robust regulatory networks

In cases such as complex diseases the expression of many genes can be significantly altered resulting in a differentially expressed disease network module. The genes involved can directly correspond to the disease phenotype (i.e. driver genes) or can be closely related to it (e.g. first degree neighbours). While the remaining ones are often not directly related to the disease.

Because a disease is a mutation in the normal pattern in a gene expression profile, we expect the expression changes of the driver genes and their first degree neighbours to be more consistent

than all the expression of the other genes. Thus, the identification of accurate and reproducible disease biomarkers is an important challenge for gene expression analysis. One example is given by Yang et al. (2012) who develop a novel pathway based biomarker identification method that extracts the essential core module of disease from known biological networks.

All organisms have many mechanisms, necessary for their survival, that carry on mostly unchanged under any condition the organism is subjected to (e.g. cell metabolism). Other mechanisms, however, occur only when some event external or internal to the organism (environmental changes, stress, cancer, etc.) happens to trigger them. Some conditions might trigger similar mechanisms (more or less based on how similar the conditions are) that researchers robustly identify using consensus networks analysis (Taylor et al. 2009).

The use of clustering techniques for microarray analysis can suffer from lack of inter-method consistency in assigning related gene expression profiles to clusters. In Swift et al. (2004) the authors create a consensus set of clusters, exploring different methods of clustering in parallel, to improve the confidence in gene expression analysis coupled with statistically based gene functional analysis to identify novel genes. The partial agreement of the different clustering algorithms should reflect the clustering of highly similar gene-expression vectors regardless of the clustering methods used. The weighted kappa metric (Altman 1990) is used to measure the discordance between clustering algorithms. They apply a minimum agreement: rather than grouping variables on the basis of full agreement only, consensus clustering maximizes a metric which rewards variables in the same cluster if they have high cluster method agreement and penalizes variables in the same cluster if they have low agreement. Robust clustering, which assumes full agreement, is also useful since it increases the module confidence but also reduces the dimensionality of large gene expression datasets.

The integration of multiple datasets derived from related biological systems leads to more robust models. Consequently we expect the use of multiple datasets of increasing biological complexity to give a deeper insight of the fundamental underlying mechanisms. Anvar et al. (2010) explore the use of Naïve Bayes Classifiers (NBC) and Bayesian Network Classifiers (BNC) for predicting expression on independent datasets in order to identify informative genes and their connections using classifiers of differing complexity. First, genes are ranked based on their informativeness. After applying the different algorithms, regulatory interactions that are consistently found across multiple datasets are more likely to be fundamentally involved and are easier to find in dataset with less biological variation. They find out the regulatory networks trained on less complex biological systems could thus be used for the modelling of the more complex biological systems.

Isella et al. (2011) implement an R-Bioconductor package named Mulcom, a derivative of the

t-test, designed to compare multiple test groups individually against a common reference. The bottleneck in genomic research has recently moved from the production of high quality data to interpretation of the data and hypothesis generation. The development of precise hypotheses from a long list of gene candidates, in fact, can be challenging. One powerful approach that has been used to aid in the interpretation of candidate genes lists is called integrative analysis with complementary genome-scale data. The basic approach adopted by these methods is to identify sub-graphs with conservation at the protein sequence level as well as at the physical or functional level. This approach has been used to suggest core pathways that are conserved across species and to build confidence in individual protein - protein interactions based on the co-occurrence in multiple species. Conserved patterns are not likely to have occurred by chance, and they are enriched for known as well as novel stem cell and differentiation-related processes. Deshpande et al. (2010) describe a scalable approach for discovering conserved active sub-networks across species.

## 2.9 Incorporating expertise

A vast volume of data is being generated and knowledge (i.e. scientific papers) is accumulating. However, knowledge is only considered implicitly in the form of assumptions which can be neither precise nor quantitative. A practical approach to overcome problems derived from low data quality, noise and measurement errors typical of microarray datasets is to incorporate existing knowledge into a computational framework. This way statistical inference can increase the knowledge in the areas that are still lacking evidence and help construct more precise models.

As explained earlier, learning a Bayesian network (BN) structure means finding a DAG that best matches the data set, maximizing the posterior probability of a DAG given the data. This allows BNs to deal with inherent stochasticity in gene expression and with the noise brought by the microarray technology. In addition, BNs are naturally capable of integrating prior knowledge into the system. So, Gao & Wang (2011) incorporate prior knowledge into BN in a quantitative way to bias the Markov Chain Monte Carlo (MCMC) simulation of candidate structures and prove that BN with prior knowledge greatly benefit the performances.

Angelopoulos & Wessels (2011) exploit Logic Programming (LP) which is an attractive formalism for representing knowledge. They discuss Distributional Logic Programming (DLP) which is formalism for combining Logic Programming and probabilistic reasoning. Prior knowledge improves the resulting gene regulatory networks including knowledge that doesn't appear in the data. Gene-pair association scores describe the overlap in the contexts in which the genes are mentioned in a simple and clear format. Steele et al. (2009) transform this literature-based

gene association scores to network prior probabilities. Prior networks can fill the gap for genes that are not of particular focus of the expression data set. They explore the effect of varying the influence of the prior knowledge through different weights. An exceedingly low weight produces a network that doesn't learn enough information and the consequent structure is the result of the only knowledge hidden in the data. An extremely high weight, instead, implies a network which is the result of only the literature analysis and not of the gene expression data. The size of the prior weight is a tricky decision since it indicates the influence we want the prior to have on the final network and it necessarily differs from one study to another.

## 2.10 Integration of multiple data

Despite the enormous amount of genomic data, most of these data sources are not completely reliable due to noise and incompleteness. Because modern technologies generate a broad array of different data types, providing distinct but often complementary information, one way to overcome the unreliable data issue is to integrate heterogeneous data sources to improve the results' trustworthiness. Savage et al. (2010) implement an algorithm to integrate gene expression and transcription factor binding (ChIP-chip) data. The model uses a hierarchical Dirichlet process mixture model to allow data fusion on a gene-by-gene basis. This approach although successful performs integrative modelling of two datasets only. On the other hand Kirk et al. (2012) develop a method to integrate a significant number of datasets simulataneously and to captures the underlying structural similarity between the datasets. The authors create a Bayesian method for the unsupervised integrative modelling of multiple datasets and data types simultaneously, including time series data. In this approach, each dataset is modelled using a Dirichlet-multinomial allocation (DMA) mixture model, with dependencies between these models captured through parameters that describe the agreement among the datasets.

Shen et al. (2009) develop a joint latent variable model for integrative clustering called iCluster which incorporates flexible modelling of the associations between different data types and the variance-covariance structure within data types in a single framework, while simultaneously reducing the dimensionality of the datasets. A further application of this technique is shown in Shen et al. (2012).

Another integration example is described by Zhang J. et al. (2012) where two cancer datasets are compared (case and control). For each dataset gene-pair expression correlation is computed and then used to build a frequency table whose values are used to build a weighted gene co-expression frequency network. After this they identify sub-networks with similar members and

iteratively merge them together to generate the final network for both cancer and healthy tissue.

## 2.11 Conclusions

In this chapter we explained the main concepts related to gene expression and the different ways to measure and analyse it. We focus on microarray data due to its popularity among biologists and the consequent large volume of publicly available data. We tackled the problems related to the high discrepancy between the size of samples and the number of genes. Therefore we explore the various gene selection techniques developed over the paste two decades. We moved then to analyse the importance of building Gene Regulatory Networks (GRNs) from the data. How to detect similar behaviour within the genes and group them into module and derive genes' functions based on the "guilt-by-association" principle when reliable. Given the high level of noise and bias in microarray data, we explore how to construct robust GRNs using a whole set of different algorithms and we discuss the improvements that can be made incorporating expertise or integrating data.

The integration of multiple data is particularly useful when no prior knowledge is available on the organism under analysis or for that specific experimental condition. Researchers have built more robust GRNs by collecting multiple datasets from the same organism under several experimental conditions. But different experimental conditions trigger different mechanism inside the same organism which are hidden or even completely lost applying this procedure. Hence, in this research we want to develop a method to robustly identify GRNs that are specific for the experimental condition under consideration to highlight what makes each condition unique compared to the others.

In the next chapter we explore in details some of the techniques to build GRNs that will be used later for our method.

# Chapter 3

# Key Concepts

Analysis of the literature has shown a great deal of attention to the discovery of common underlying mechanisms between independent studies, especially in the case of similar diseases. However, biologists are now starting to recognize the importance of highlighting the differences. In this work we build a pipeline to semi-automatically identify the differences between various experimental conditions an organism is subjected to. This method takes as input a set of different independent studies to explore and detect the mechanisms (gene paths) that render each study (or group of similar studies) *unique* compared to the others. The entire procedure faces several theoretical and computational challenges explored and resolved in the remainder of this thesis.

Gene Regulatory Networks (GRNs) are a user-friendly representation of the underlying mechanisms of an organism, easily interpretable by non-technical people. Thanks to the explosion of publicly available data and the popularity of their use we choose to develop this method focusing on microarray datasets. However, as described in Chapter 2, the structure and the generation of microarrays make the creation of reliable and robust GRNs a difficult task.

In this chapter we illustrate the techniques developed over the last two decades to convert gene expression profiles into GRNs, that we exploit within the pipeline.

## 3.1 Co-expression

The term *co-expression* is used to indicate the simultaneous expression of two or multiple genes. Considering a set of genes and their expression profiles the main objective is to discover the interactions and relationships between them.

The easiest and most common technique used to determine the existence and quantify any

kind of relationship between two (or multiple) genes is to calculate the so called **Pearson correlation coefficient ($\rho$)** or simply correlation coefficient. Given two random variables $X$ and $Y$, Pearson coefficient is obtained dividing the covariance of the two variables by the product of their standard deviations:

$$\rho_{X,Y} = corr(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \tag{3.1}$$

The value of $\rho_{X,Y}$ varies between $-1$ and $+1$. If $\rho_{X,Y} \neq 0$ means that $X$ and $Y$ are dependent, $\rho_{X,Y} > 0$ indicates a direct relationship (if $X$ increases $Y$ increases simultaneously) and $\rho_{X,Y} < 0$ indicates an inverse relationship (if $X$ increases $Y$ decreases and viceversa).

Stuart et al. (2003) analyse DNA microarrays from different organisms and calculate the Pearson correlation coefficient of the expression profiles between every pair of genes in the microarray data sets for each organism in order to identify gene interactions that are evolutionarily conserved.

A more generic class of coefficients called **Rank correlation coefficients** measure the dependency between variables without requiring a linear relationship between them, unlike the Pearson's coefficient. Some examples are Spearman's $\rho$ (Pirie 1988), Kendall's $\tau$ (Abdi 2007) and Goodman and Kruskal's $\gamma$ (Goodman & Kruskal 1954) coefficients.

Another popular measure of the variables' mutual dependence is the **Mutual Information coefficient** (MI) defined for two discrete random variables, as:

$$I(x,y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{3.2}$$

where $p(x,y)$ is the joint probability distribution function of $X$ and $Y$, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of $X$ and $Y$ respectively. In the case of continuous random variables, the summation is replaced by the integral sign. MI is always non-negative and measures the information shared between $X$ and $Y$. In other words, it measures how much knowing one variable reduces the uncertainty about the other. If $I = 0$ then $X$ and $Y$ are independent which means they do not share any information. On one side the use of mutual information is better since it doesn't necessitate of linear relationships between variables, but its use in continuous data is complicated by the fact that it requires an estimate (explicit or implicit) of the probability distribution underlying the data (Kinney & Atwal 2014).

An application on real data is shown in Butte & Kohane (2000) who develop a technique that computes comprehensive pair-wise mutual information for all genes in a dataset and build the networks by setting a threshold mutual information and using only associations (links) at or above that threshold.

Although these coefficients measure the degree of statistical dependency between variables, it is

important to highlight that correlation of two or multiple variables does not implicate a causal relationship between the variables!

## 3.2 Clustering

Cluster analysis is intended as the approach of organizing objects into groups whose members are similar to each others and different (or less similar) to the objects in the other groups. Each group is called *cluster*. In data analysis and pattern discovery, clustering is a broad term that identifies fundamental techniques to extract underlying cluster structures (Baldi & Brunak 2001). The covariance or correlation matrix of the genes are a simple and popular example of a way to identify clusters of genes that are related to each other. A part from these, several more detailed algorithms have been developed for the clustering purpose. The choice of one over another depends on several factors such as the type of data we are dealing with and the information needed to retrieve.

Clustering algorithms are divided into two big categories: supervised and unsupervised. While supervised clustering infers the clusters using labelled training data, unsupervised clustering algorithms use unlabelled data.

K-means together with hierarchical clustering are the oldest and most popular unsupervised clustering algorithms of which several version have been developed over the decades.

K-means (Hartigan 1975) partitions the $n$ observations in $m$ dimensions (usually a similarity matrix), into $k$ (variable set manually by the user) clusters in a way that the within-cluster sum of squares is minimized. This algorithm run iteratively and is a NP-hard problem, therefore it is necessary to apply heuristic algorithm such as the one developed by Hartigan & Wong (1979).

Hierarchical clustering (Eisen et al. 1998) iteratively creates a dendrogram that assembles all elements into a tree, starting from the correlation or other similarity-type kind of matrices.

All clustering algorithms necessitate of a similarity-type of matrix. In this work we apply the sensitivity matrix. Sensitivity and specificity are two statistical measures highly used to measure the performance of a binary model. The example in Figure 3.1 shows a table, called contingency table, that summarizes the frequency distribution of the variables of a study that evaluates a clinical test to diagnose a specific condition.

| **Total population** | Condition positive | Condition negative |
|---|---|---|
| Test positive | True Positive | False Positive |
| Test negative | False Negative | True Negative |

Figure 3.1: Contingency table

Given this table, the sensitivity and specificity are defined as:

$$Sensitivity = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$
$$Specificity = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

which in other word is the probability of getting a positive test when the patient has the condition.

In our work we apply calculate the sensitivity to measure how two GRNs are similar to each other. Considering two networks $N_1$ and $N_2$, true positive is the number of connections that $N_1$ and $N_2$ have in common, false positive the number of connections that appear in $N_1$ but not in $N_2$ and false negative the number of connections in $N_2$ but not in $N_1$. A simple example is shown in Figure 3.2.

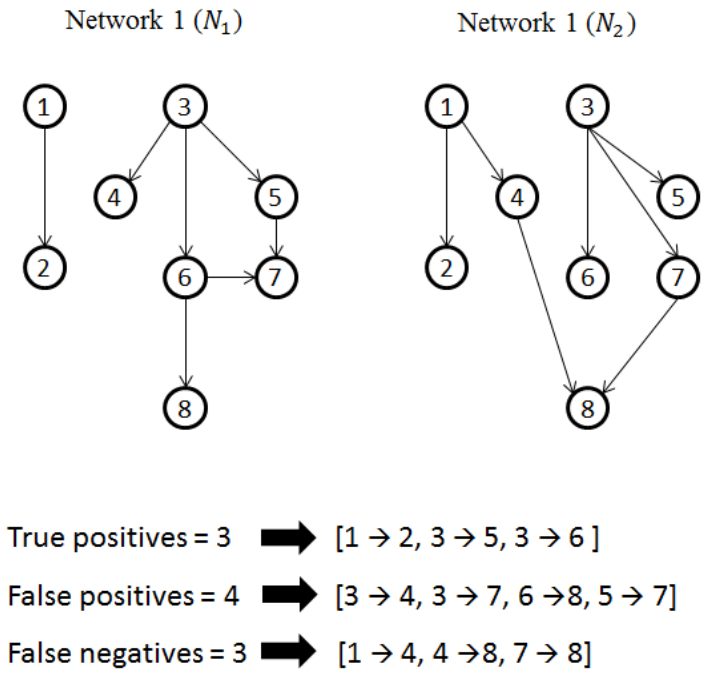



Figure 3.2: Example of how to calculate true positives, false positives and false negatives between two networks.

Doing this we obtain a sensitivity matrix that is then used as input to the k-means clustering algorithm (details and results are described in Chapters 4 and 5). A clustering algorithm should identify reliable and robust clusters meaning that they have to be representative of the data even when the observations are affected by noise or bias (systematic error). A related approach is called *consensus clustering* (Swift et al. 2004) which, given a set of clusters, aims to build

one single cluster to better represent the input ones. This is a technique often used to overcome bias, noise or even merge similar clusters to reduce the dimensionality. In the specific case of this thesis we applied consensus clustering in order to build networks that represent a larger group of experimental conditions.

## 3.3 Scale free vs Random graphs

A graph $G$ (Nagarajan et al. 2013) is a representation of objects whose relationship between each other is represented through links. Each object or variable is represented with a node also called vertex $V$ and the links that connect pair of vertices are called edges $E$. An edge that connects vertex A with vertex B is *directed* when it starts from A and ends in B (A $\longrightarrow$ B), and *undirected* when the direction of the link is not defined (A — B). Thus, a graph can be *directed*, *undirected* or *partially directed*. Finally, a graph is called *acyclic* when there are no cycles, meaning no edges that connect node A to itself.

A user-friendly representation of gene interactions is through a graph framework $G(V, E)$ where $V$ are the vertices or nodes and $E$ the edges that connect the vertices to each other. In the case of gene expression, the vertices are the genes and the edges represent the relationships between them. When we analyse complex networks we often have no information about their structure, so several models have been developed over the years. Erdős & Rényi (1959) build the first model based on a random approach. The model sets an edge between each pair of nodes (N) with equal probability $p$ and the probability of a vertex to have $k$ edges follows a Poisson distribution

$$P(k) = e^{-\lambda}\frac{\lambda^k}{k!}, \qquad \text{where } \lambda = N\binom{N-1}{k}p^k(1-p)^{N-1-k}$$

Later on, Watts & Strogatz (1998) describe the $small-world$ problem. Here, $N$ vertices form a one-dimensional lattice where each vertex is, first, connected to its own nearest and next-nearest neighbours and then, each edge is reconnected to a random vertex with probability $p$. This process generates connections in a way that decrease the distance between the vertices. In both models, the probability of finding highly connected nodes decreases exponentially with $k$ (number of edges), meaning that they are practically absent.

Contrary of these models, it has been noticed that real complex networks self-organize into a scale-free state: the probability $P(k)$ that a vertex in the network interacts with $k$ other vertices decays as a power law following:

$$P(k) \sim k^{-\gamma}$$

In Barabási & Albert (1999), the authors notice that two aspects of real networks are ignored

in random models.

First, the models assume the network starts with a fixed number (N) of vertices that are randomly connected, without modifying N. But real world networks continuously expand adding new vertices and connecting them to the already existing ones (e.g. publications in research literature).

Second, random network models assume that the probability that two vertices are connected is random and uniform, while most real networks show *preferential connectivity* which means that the probability that a new vertex connects to a new vertex is not uniform but is higher for already highly connected nodes (e.g. a highly cited paper have a higher probability, compared to an unknown one, to be cited by a new paper). A network model based on these two facts determines the scale-invariant distribution.

The model starts with $(m_0)$ vertices, at every time step a new vertex is added with $m \leq m0$ edges that link the new vertex to $m$ different vertices already in the network model.

Because of preferential attachment, the probability $P$ that a new vertex is connected to vertex $i$ depends on the connectivity $k_i$ of that vertex, so that $P(k_i) = \frac{k_i}{\sum_j k_j}$. After $t$ time steps, the model leads to a random network with $t + m_0$ vertices and $m \times t$ edges. This network becomes scale-invariant and the probability that a vertex has $k$ edges follows the power law. Furthermore, $P(k)$ is independent of time and of the system size $m_0 + t$, which indicates that despite its continuous growth, the system self-organizes into a scale-free stationary state. These models perfectly describe genetic or signalling networks. Although they are now stable, the growth can be seen as the evolutionary history.

## 3.4 Weighted Gene Correlation Network Analysis

A Correlation network represents the pairwise relationships between variables/genes. The network framework is so easy to understand, even for non-bioinformaticians that it is now a popular tool used to analyse complex networks resulting from large and high dimensional data such as microarrays.

WGCNA R software package (Langfelder & Horvath 2008) collects a set of R functions to perform various types of weighted correlation network analysis such as co-expression network analysis of gene expression data (Sengupta et al. 2009, Langfelder & Horvath 2012).

Given a $n \times m$ data matrix $X = [x_{ij}]$, where $i = 1, ..., n$ are the genes (nodes of the networks) and $l = 1, ..., m$ are the samples measurements. The $i$th row $x_i$ is called the $i$th gene expression profile across $m$ sample measurements.

A graphical network is mathematically identified by its adjacency matrix defined as a $n \times n$ ma-

trix whose elements $[a_{ij}]$ encode the connection strength between nodes $i$ and $j$ of the network. In the specific case of co-expression networks, this strength is obtained by, first, calculating the absolute value of the correlation coefficient (co-expression similarity) between the nodes profile $i$ and $j$: $s_{ij} = |corr(x_i, x_j)|$, and then, by applying a threshold that transforms the co-expression similarity matrix into the adjacency matrix. The type of threshold applied determines the type of network.

Hard thresholding, defined as:

$$a_{ij} = \begin{cases} 1 & \text{if } s_{ij} \geq \tau \\ 0 & \text{otherwise} \end{cases}$$

where $\tau$ is the hard threshold parameter, creates unweighted networks. Even though unweighted networks are widely used, they do not reflect the continuous factor of the underlying co-expression which may lead to loss of information. On the other hand, weighed network adjacency is obtained by setting a soft threshold that means raising the co-expression similarity to a power: $a_{ij} = s_{ij}^{\beta}$, with $\beta \geq 1$. This implies that the weighted adjacency $a_{ij}$ between two genes is proportional to their similarity on a logarithmic scale, $log(a_{ij}) = \beta \times log(s_{ij})$. To pick the right value of beta WGCNA uses a biologically motivated criterion called the **scale-free topology criterion** (Barabási & Albert 1999, Zhang et al. 2005). As opposed to the random graph model (Erdős & Rényi 1959), scale-free networks present only few highly connected nodes (hubs) and display a high degree of tolerance against errors (Barabási & Albert 1999). One way to prove the network has this topology is to plot $log_{10}p(k)$ vs $log_{10}(k)$, if an approximate straight line appears the scale-free topology is satisfied. Otherwise calculate the correlation between $log_{10}p(k)$ and $log_{10}(k)$ which represents the model fitting index $R^2$ of the linear model that regresses $log_{10}p(k)$ on $log_{10}(k)$. If $R^2 \approx 1$, then there is a straight line relationship between $log_{10}p(k)$ and $log_{10}(k)$.

So, given the power adjacency function in a weighted gene network $a_{ij} = |corr(x_i, x_j)|^{\beta}$, WGCNA considers $\beta$ values that lead to a network that approximately satisfies the scale-free topology.

An additional function available in the WGCNA package is to transform the adjacency function into the Topological Overlap Matrix (TOM) that detects subsets of nodes (modules) that are tightly connected to each other and minimizes the effects of noise and spurious associations. This is done by calculating the topological overlap dissimilarity measure (Ravasz et al. 2002) which evaluates the relative interconnectedness between pair of nodes. TOM is defined as:

$$\omega_{ij} = \frac{l_{ij} + a_{ij}}{min\{k_i, k_j\} + 1 - a_{ij}}$$

where $l_{ij} = \sum_u a_{iu} a_{ui}$ and $k_i = \sum_u a_{iu}$ is the node connectivity. In the case of hard thresholding

(unweighted networks) $\omega_{ij} = 1$ if the node with fewer connections is connected to the other node and all of its neighbours are also neighbours of the other node, $\omega_{ij} = 0$ otherwise. To generalise it for the case of weighted networks $a_{ij}$ accepts real numbers $0 \leq a_{ij} \leq 1$.
Since TOM $\Omega = [\omega_{ij}]$ provides a similarity measure, the topological overlap dissimilarity measure is 1 - TOM.

### 3.4.1 WGCNA networks applied to wheat

Biological networks show some characteristics that seems to satisfy the scale-free model organization. For example, these networks show a high degree of internal order that governs the cell's molecular organization (Barabasi & Oltvai 2004) rather than a random one and the growth criteria can be satisfied by their evolutionary history. Although scale-free framework seems to easily explain the complexity of biological networks, Khanin & Wit (2006) and Stumpf et al. (2005) suggest to use some caution.
In this work we decide to explore building networks beyond pairwise correlation but, given its success, we investigate WGCNA as a comparison of our pipeline abilities. The first pipeline performance study on real data is made on multiple studies of wheat. We focus on 16 independent studies downloaded from Array Express database (Rustici et al. 2013, Parkinson et al. 2007) of stress enriched and non-stress condition, each containing 61290 genes. Table 3.1 shows the studies and their corresponding number of samples and descriptions.
We first want to check if the entire system can be described by a scale-free network. So, after the studies are merged together we calculate the connectivity $k$ and then plot $k$ vs $p(k)$ to explore the nature of the datasets. Figure 3.3 shows on one side the histogram of the connectivity which denotes a high number of nodes with a low connectivity and lower one but still present for hubs and on the other the relation of $k$ vs $p(k)$ in logarithmic scale. As also highlighted in the figure title the value of $R^2$ is equal to 0.83 which we can consider close enough to 1, as well as the absolute value of the slope. On these first results we can deduce that the general underlying mechanism of these studies can be described through a scale free network.
Given that, we now want to build weighted co-expression networks, one for each wheat dataset and compare it afterwards with our pipeline results. For computational reasons, we first need to reduce the number of variables. First, the genes that are not part of the Gene Ontology database (Ashburner et al. 2000) and therefore not biologically known (yet), are discarded. Then, the standard deviation for each gene in each study across all samples is calculated and only the genes with sd $\geq 2$ in at least 4 of the 16 studies are finally selected for the rest of the analysis. The value of the sd threshold is defined by the user based on the number of genes that the user

believe can be reasonably analysed. The first step reduces the genes from 61290 to 21487, that after the second step are reduced to the final number of 67 genes. More details can be found in Chapters 4 and 5. For each study, once we build the co-expression similarity matrix we need to transform it into the adjacency matrix to define the final study-network. Since we are interested in directed networks, we choose to apply the soft-thresolding procedure which requires the selection of the parameter $\beta$ (power). Common practise requires to set $\beta = 6$ for signed networks and $\beta = 12$ for unsigned ones. Although, different studies imply different underlying mechanisms and possibly a different $\beta$ value. We explore a set of values for the parameter $\beta$ from 1 to 30 and analyse the effects.

Wheat Studies

| Study | Label | Samples | Description |
|---|---|---|---|
| 1 | E-MEXP-971 | 60 | Salt stress |
| 2 | E-MEXP-1415 | 36 | S and N deficient conditions |
| 3 | E-MEXP-1193 | 32 | Heat and Drought Stress |
| 4 | E-MEXP-1694 | 6 | Re-supply of sulfate |
| 5 | E-MEXP-1523 | 30 | Heat stress |
| 6 | E-MEXP-1669 | 72 | Different nitrogen fertiliser levels |
| 7 | E-GEOD-4929 | 4 | Study parental genotypes 2 |
| 8 | E-GEOD-4935 | 78 | Study 39 genotypes 2 |
| 9 | E-GEOD-6027 | 21 | Meiosis and microsporogenesis in hexaploid bread wheat |
| 10 | E-GEOD-9767 | 16 | Genotypic differences in water soluble carbohydrate metabolism |
| 11 | E-GEOD-12508 | 39 | Wheat development |
| 12 | E-GEOD-12936 | 12 | Effect of silicon |
| 13 | E-GEOD-11774 | 42 | Cold treatment |
| 14 | E-GEOD-5937 | 4 | Parental genotypes 2 biological replicates from SB location |
| 15 | E-GEOD-5939 | 72 | 36 genotypes 2 biological replicates from SB location |
| 16 | E-GEOD-5942 | 76 | Parental and progenies from SB location |

Table 3.1: Study numbers, labels, number of samples and descriptions of the wheat microarray dataset.

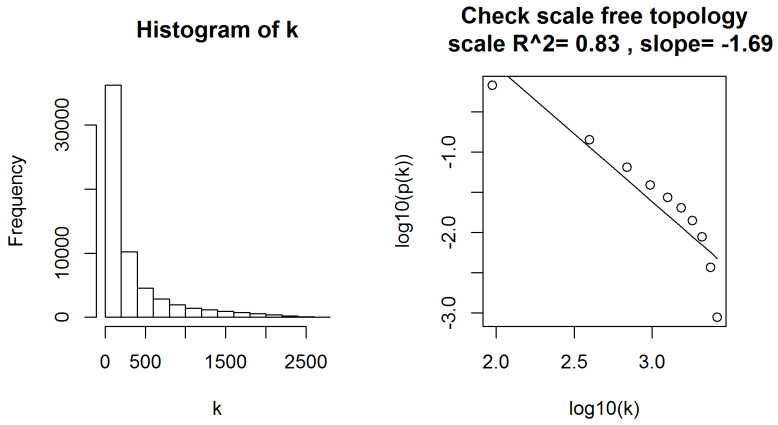



Figure 3.3: Scale-free plot. The figure on the left hand side show the distribution of the connectivity (k), while the one on the right represent the relation between $k$ and $p(k)$ in logarithmic scale highlighting that the slope is close to -1.

Figure 3.4 shows the variation of $R^2$ in correspondence to different values of $\beta$. As previously explained the closer to 1 $R^2$ gets the better it is. Therefore, in each study, we select the first value of beta that corresponds to $R^2 \geq 0.8$. Although, in the figure, many studies never reach the threshold 0.8, leaving us with no $\beta$ to select. This may be due to the low number of samples available per study or even to the set of reduced genes. Once the values of $\beta$ are chosen, we calculate the adjacency matrices that are going to define the study networks, one per study, which in turn are used as a base to create unique networks to compare with the ones obtained by our pipeline. The unique networks resulting from the WGCNA procedure can be seen in Chapter 5.

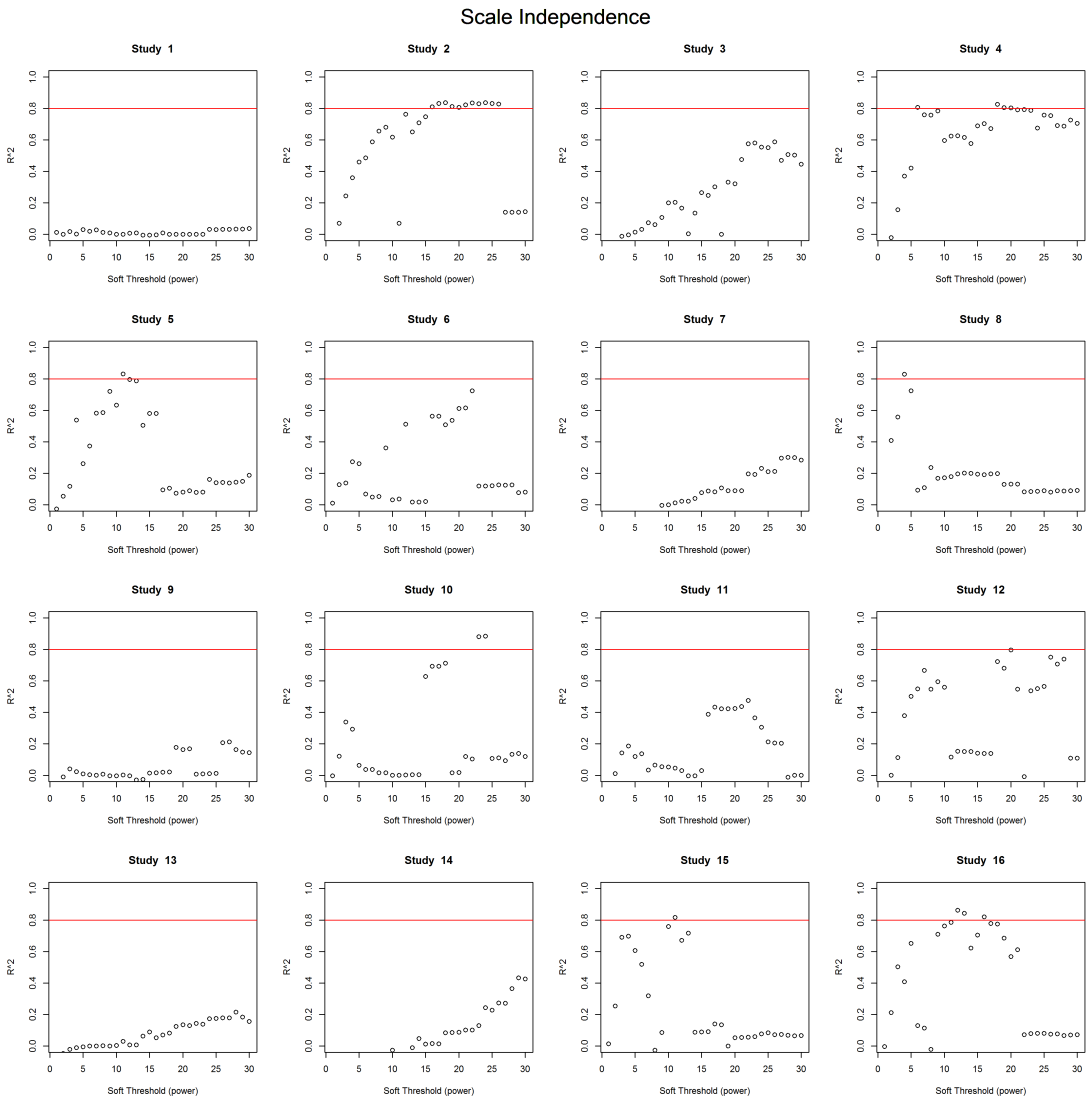



Figure 3.4: Scale independence. Each plot shows the variation of $R^2$ for different values of $\beta$ (power) for each single study under analysis. The red horizontal line identifies the threshold set at 0.8. Above which $R^2$ satisfies the scale-free criteria therefore the corresponding value of $\beta$ can be used in the soft-thresholding procedure.

## 3.5 Modelling GRNs using Glasso

### 3.5.1 Inverse covariance and partial correlation

The covariance is a statistical measure that, as well as the correlation coefficient, defines the degree of similarity between two random variables. Given two random variables $x$ and $y$, $cov(x, y) = \sigma(x, y) = E[(x - E[x])(y - E[y])]$. A covariance value higher than 0 indicates similar behaviour between the variables (e.g. given two variables A and B when A grows, B grows as well), while a value lower than 0 indicates an opposite behaviour (when A grows, B decrease and vice versa).
Because the covariance strongly depends on the data, correlation normalizes the covariance dividing it with by the product of the standard deviations of the variables in consideration (see formula 3.1). This creates a dimensionless coefficient that ease the comparison of datasets with different scale. On the other hand, the inverse of the covariance matrix (also called concentration or precision matrix) is a matrix whose elements are interpreted in terms of partial correlation. Partial correlation measures the degree of association between two random variables (same as correlation and covariance), but with the effect of the controlling random variables removed. It aims at finding correlation between two variables after removing the effects of other variables. This analysis avoids spurious correlations (i.e. correlations explained by the effect of other variables) but reveals hidden ones (i.e. correlations masked by the effect of other variables).
Given two random variables $x$ and $y$ linearly related to variable $z$:

$$x = Az + B + d_x$$
$$y = Cz + D + d_y$$

the partial correlation coefficient $r_{xy.z}$ is defined as the correlation coefficient between the residuals $d_x$ and $d_y$. Again, this coefficient varies between -1 and 1. The variables are conditionally independent, given all the other variables, if the partial correlation coefficient is equal to zero and conditionally dependent, given all the other variables, otherwise (Lauritzen 1996).

### 3.5.2 Lasso

Another way of seeing this problem is through regression. Given the data $(x^i, y_i), i = 1, 2, ..., N$, where $x_i = (x_{i1}, ..., x_ip)^T$ and $y_i$ are respectively the regressors and response for the $i$th observation, the ordinary least squares (OLS) estimates are obtained by minimizing the residual squared error. However, this method incurs two drawbacks:

- *Prediction accuracy*: the OLS estimates often show low bias but large variance;

- *Interpretation*: the large number of predictors complicates the interpretation of the results.

The prediction accuracy can be improved by shrinking or setting to 0 some coefficients. Sacrificing bias to reduce the variance of the predicted values may improve the overall prediction accuracy. On the other hand interpretation can be refined selecting a smaller subset of predictors which exhibits the strongest effects.

To improve the OLS estimates, two methods have been developed which became quite popular: subset selection and ridge regression. Subset selection is a discrete process in which regressors can either be conserved or discarded from the model. Small changes in the data often lead to select very different models which reduces its prediction accuracy. On the other hand ridge regression is a continuous process that shrinks the coefficients. The method is more stable but because it does not set any coefficients to exactly 0, the interpretation is still complicated.

In Tibshirani (1996), the authors introduce a new method called *Least Absolute Shrinkage and Selection Operator* (LASSO) which novelty is that it shrinks some coefficient and set others to exactly zero. This, of course, ease the interpretation of the model and improve the prediction accuracy. Given the predictor variables $x^i$ and the responses $y_i$, it is assumed that the responses are conditionally independent given the predictors and the predictors are standardized e.g. $\sum_i x_{ij}/N = 0$ and $\sum_i x_{ij}^2/N = 1$.

Let $\widehat{\beta} = (\widehat{\beta}_1, ..., \widehat{\beta}_p)^T$, the lasso estimate $(\widehat{\alpha}, \widehat{\beta})$ is defined by

$$(\widehat{\alpha}, \widehat{\beta}) = \arg\min\Big\{ \sum_{i=1}^{n} \big(y_i - \alpha - \sum_{j=1}^{p} \beta_j x_{ij}\big)^2 \Big\} \qquad \text{subject to} \sum_{j=1}^{p} |\beta_j| \leq t. \tag{3.3}$$

Where $t \geq 0$ is the tuning parameter which determines how much penalization is applied to the estimates. For all possible $t$, the solution for $\alpha$ is $\widehat{\alpha} = \overline{y}$ but the derivative of a constant is zero, so we can assume $\overline{y} = 0$ without loss of generality and then omit $\alpha$. The structure of the formula 3.3 suggests it is a quadratic programming problem with linear inequality constraints. Now, considering $\widehat{\beta}_j^0$ the full least squares estimates and $t_0 = \sum |\widehat{\beta}_j^0|$, all values of $t < t_0$ shrink the solutions towards 0, and some of them to exactly 0.

### 3.5.3 Graphical lasso

Consider a $p$-dimensional multivariate normal distributed random variable $X = (X_1, ..., X_p) \sim \mathcal{N}(\mu, \Sigma)$. If the covariance matrix $\Sigma$ is non singular, the conditional independence structure of the distribution can be represented through a graphical model $\mathcal{G} = (\Gamma, E)$ where $\Gamma = \{1, ..., p\}$ is the set of nodes and $E$ the set of edges in $\Gamma \times \Gamma$. An edge $(a, b)$ (between between $a$ and $b$) exists in $E$ if and only if $X_a$ is **conditionally dependent** on $X_b$, given all remaining variables. Consequently, each pair of variables non included in the edge set is conditionally independent,

given the remaining variables, and correspond to a zero in the inverse covariance matrix (Lauritzen 1996) (see section 3.5.1). Following this line of thoughts Dempster (1972) are the first to use the covariance approach to build a graph (conditional independence restrictions) for a set of i.i.d observations. The covariance selection needs the discrete optimization of an objective function for which greedy forward and backward search are employed, but the complexity of the procedures makes it computationally impractical even for small sized graphs.

In Meinshausen & Bühlmann (2006) the authors introduce a new procedure that explores the Lasso for neighbourhood selection. They estimate a sparse graphical model by fitting a lasso model consecutively to each node in the graph, using the others as predictors.
The Lasso (Tibshirani 1996) has a parsimonious property which means that when predicting a variable $X_a$ given the remaining variables $\{X_k : k \in \Gamma(n) \setminus \{a\}\}$, the vanishing lasso coefficient estimates identify asymptotically the neighbourhood of a node $a$ in the graph.
Given the matrix $X[n \times p(n)]$ which contains $n$ independent observations of $X$, so that the columns $X_a$ correspond for all $a \in \Gamma(n)$ to the vector of $n$ independent observations of $X_a$. The Lasso estimate $\hat{\theta}^{a,\lambda}$ of $\theta^a$ is:

$$\hat{\theta}^{a,\lambda} = \underset{\theta:\theta_a=0}{\operatorname{argmin}}(n^{-1}\|X_a - X\theta\|_2^2 + \lambda\|\theta\|_1)$$

where $\|\theta\|_1 = \sum_{b\in\Gamma(n)} |\theta_b|$ is the $l_1$-norm (Minkowski distance with exponent = 1) of the coefficient vector. The neighbourhood estimate (parametrized by $\lambda$) is defined by the non-zero coefficient estimates of the $l_1$-penalized regression,

$$\hat{ne}_a^{\lambda} = \{b \in \Gamma(n) : \hat{\theta}_b^{a,\lambda} \neq 0\}.$$

Large values of the penalty reduce the size of the estimated set while small values increases it. The method is computationally very efficient and is consistent even for high-dimensional settings/graphs. Although it is an approximation to the exact problem.

### 3.5.4 Glasso implementation in R

To summarize what was previously said, the problem of identifying the structure of a network can be solved by estimating the relationships between variables. In the case of undirected graphs it is the same as learning the structure of the conditional independence graph (CIG), which in the specific case of Gaussian random variables, means to identify the zeros of the inverse covariance matrix (also called precision or concentration matrix). Given a $p$-dimensional normally distributed random variable $X$, assuming that the covariance matrix is non-singular, the conditional independence structure of the distribution can be represented by the graphical

model $G = (N, E)$ where $N = (1, ..., p)$ is the set of nodes and E is the set of edges in $N \times N$. If an edge (pair of variables) is not in the set E it means that the two variables are conditionally independent given the other variables. This corresponds to a zero in the inverse covariance matrix.

The R package *glasso* (*graphical lasso*) (Friedman et al. 2014) estimates sparse graphs by applying the lasso penalty to the inverse covariance matrix. Continuous data are described by a multivariate Gaussian distribution with mean $\mu$ and covariance $\Sigma$, therefore the variables $i$ and $j$ are conditionally independent, given the other variables, if the $i, j$th element of $\Sigma^{-1}$ is zero. This method aims to simultaneously strengthen the connections between variables and reduce the number which is done by applying the lasso with $L_1$ penalty.
The problem is to maximize the penalized log likelihood:

$$\log det\Theta - tr(S\Theta) - \rho \left\|\Theta\right\|_1 \tag{3.4}$$

where $\Theta = \Sigma^{-1}$, $S$ is the empirical covariance matrix and $\left\|\Theta\right\|_1$ is the $L_1$ norm (the sum of the absolute values of the elements of $\Sigma^{-1}$) and $\rho$ is the regularization parameter.
Banerjee et al. (2008) demonstrate the problem to be convex and solve it estimating $\Sigma$ instead of $\Sigma^{-1}$. Specifically, given $W$ the estimation of $\Sigma$ they optimize over each row and corresponding column of $W$ following a block coordinate descent manner. Given $W$ and $S$ as:

$$W = \begin{pmatrix} W_{11} & w_{12} \\ w_{12}^T & w_{22} \end{pmatrix}, \qquad S = \begin{pmatrix} S_{11} & s_{12} \\ s_{12}^T & s_{22} \end{pmatrix} \tag{3.5}$$

then

$$w_{12} = \underset{y}{\operatorname{argmin}}\{y^T W_{11}^{-1} y : \|y - s_{12}\|_\infty \leq \rho\} \tag{3.6}$$

which, since the problem is convex, is equivalent to solve

$$\min_\beta \{\frac{1}{2}\|W_{11}^{1/2}\beta - b\|^2 + \rho\|\beta\|_1\} \tag{3.7}$$

where $b = W_{11}^{1/2} s_{12}$. If $\beta$ solves 3.7, then $w_{12} = W_{11}\beta$ solves 3.6. The structure of this formula looks like the lasso, so the fast coordinate descent algorithm (Friedman et al. 2007) is applied which makes a good solution of the lasso problem. The usual lasso estimates takes as input $S_{11}$ and $s_{12}$. To solve 3.7, glasso instead uses $W_{11}$ and $s_{12}$. Then it updates $W$ and cycle through all the variables until convergence. The parameter $\rho$ can be a scalar (typical situation) or a $p \times p$ matrix, if $\rho = 0$ means no regularization (Friedman et al. 2008, Meinshausen & Bühlmann 2006). This algorithm is extremely fast even for high dimensional datasets such as microarrays.

### 3.5.5 Glasso networks applied to wheat

Given the covariance matrix, *glasso* returns the inverse covariance matrix calculated applying the lasso (L1) penalty. The penalty parameter $\rho$ can be tuned by the user based on how sparse the matrix needs to be. The more sparse the matrix is the less number of edges the resulting networks will have but also the strength and reliability of these edges are highly improved. Hence, the tuning parameter $\rho$ can be chosen based on the number of connections the user is happy to work with.

In this work we want to identify underlying mechanisms that are specific for the study or group of studies under analysis. Of course, we expect only a reduced set of genes to be involved in the *unique-mechanisms*, therefore we need to considerably reduce the possible number of edges and consequently of genes involved in the network paths.

If $\rho = 0$ means no regularization (no penalty), we need to choose a value above zero. Traditionally $\rho$ is chosen between 0.010 and 0.020, but again the user can modify it depending on the needs.

For each study and for all the datasets explored in this work we applied different values of the tuning parameter, in wheat $\rho = 0.020$ in Fusarium $\rho = 0.010$ and for the cancer datasets $\rho = 0.050$. The resulting networks for each study are called in here glasso-networks. A practical example of the effect on wheat of different values of $\rho$ is shown in Figures 3.5, 3.6 and 3.7. The first network built with $\rho = 0.005$ shows a highly connected structure which is reflected in the histogram of the nodes *degree* (number of connections a node has to other nodes). The histogram shows a quite distributed distribution with very few nodes with 0 or maximum degree. The Figure 3.6 describes the network built with parameter $\rho = 0.010$. In here the network is less connected and consequently the adjacency matrix more sparse. The histogram in fact shows a higher number of nodes with degree equal to zero. Finally, Figure 3.7 is extremely sparse with only very few connections and degree distribution shifted towards zero. These results can be easily generalized for any other networks and datasets.

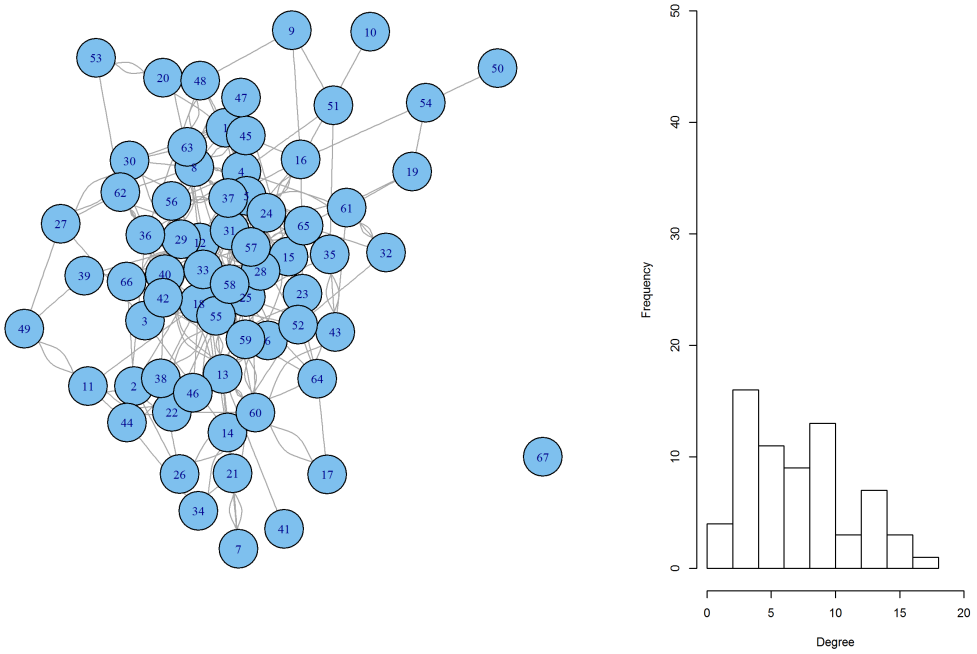



Figure 3.5: Network built with glasso and parameter $\rho = 0.005$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names.

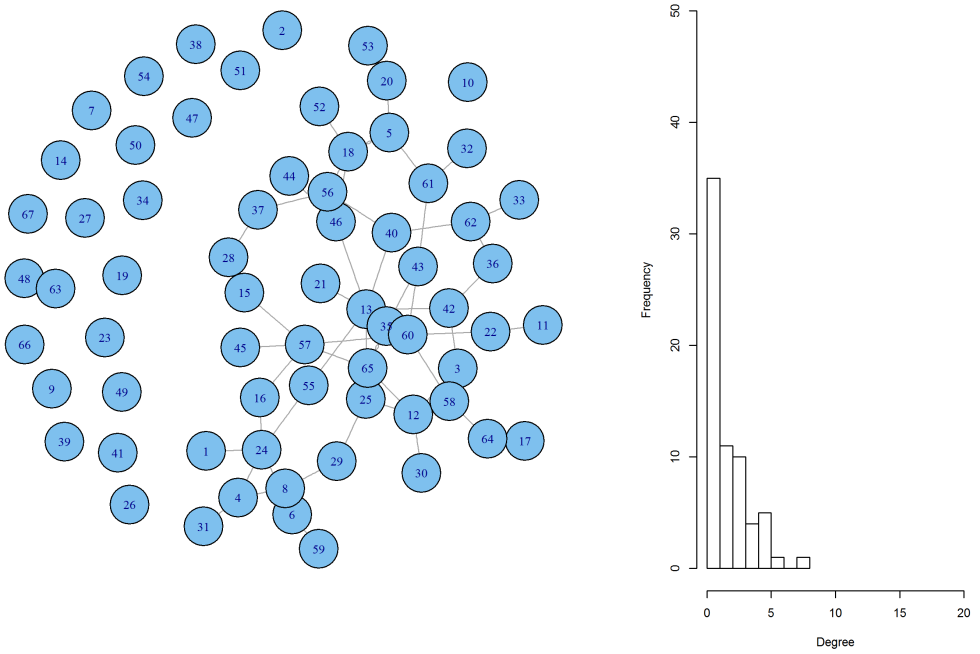



Figure 3.6: Network built with glasso and parameter $\rho = 0.010$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names.

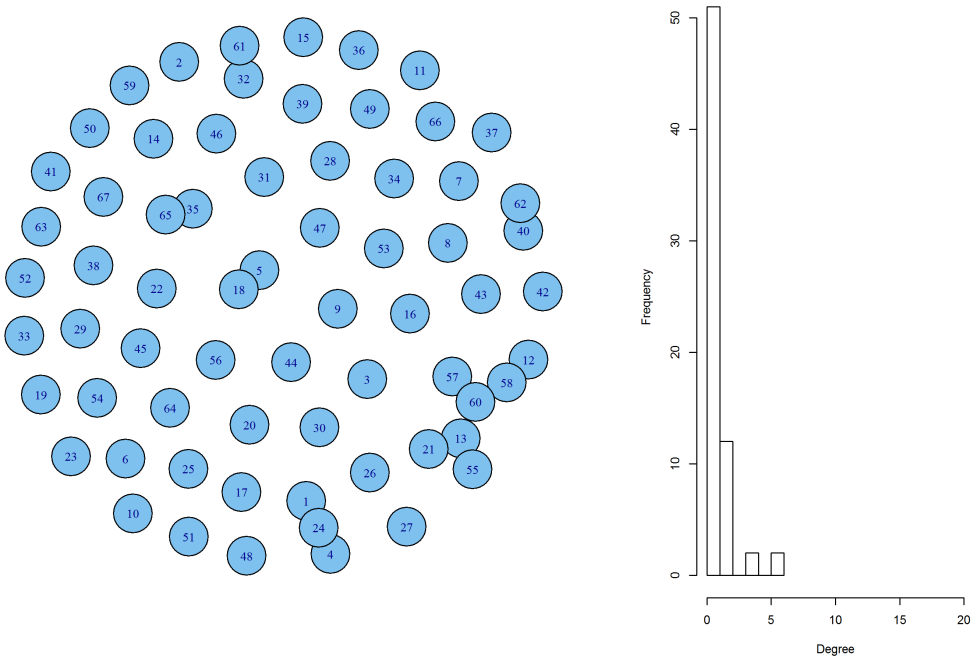



Figure 3.7: Network built with glasso and parameter $\rho = 0.020$ for the first study of the wheat dataset and corresponding histogram of nodes degree. The numbers in the network represent genes names.

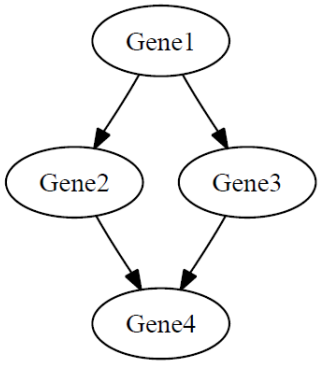
## 3.6 Modelling GRNs using Bayesian Networks

Bayesian Networks (BNs) are a tool that combines statistics with graph theory to capture relationships/dependencies among independent variables. The dependencies are qualitatively visualised through graph-based structures (networks) while the strength of the relationships are quantitatively represented by conditional probability tables or distributions for discrete and continuous data respectively. The graph based structure coupled with the conditional probability tables make BNs extremely easy to interpret by biologists and other non-technical people, consequently they have become exceptionally popular for the analysis of biological data.
A Bayesian Network (Pearl 1988, Heckerman et al. 1995, Nagarajan et al. 2013) is defined as: a probabilistic graphical model that encodes a joint probability distribution of a set of random variables. It consists of:

- Directed Acyclic Graph (DAG) where the vertices/nodes are the random variables/genes and the edges represent the conditional relationships between them;
- Conditional probability (distributions or tables) for each variable (continuous or discrete) given the parents in the graph.

A simple example of a generic Bayesian network adapted from the Sprinkler network (Murphy 2001) is shown in Figure 3.8. The DAG shows 4 nodes, each representing a gene with discrete values (on-off). The links between the nodes indicate that Gene 1 directly influences Gene 2 and Gene 3 which in turn influence Gene 4. The conditional probability tables quantify the strength of each link.
Bayesian Networks are called Dynamic BNs when modelling time series data, and Static BNs or simply BNs otherwise. In this research we explore microarray data with no temporal information, therefore we concentrate on describing static BNs and their applications.
The process of learning a Bayesian Network from the data (Koller & Friedman 2009) consists ot two steps:

- model selection
- parameter estimation.




| p(G1=on) | p(G1=off) |
|---|---|
| 0.50 | 0.50 |

| G1 | p(G2=on) | p(G2=off) |
|---|---|---|
| off | 0.50 | 0.50 |
| on | 0.80 | 0.20 |

| G1 | p(G3=on) | p(G3=off) |
|---|---|---|
| off | 0.75 | 0.25 |
| on | 0.25 | 0.75 |

| G2 | G3 | p(G4=off) | p(G4=on) |
|---|---|---|---|
| off | off | 0.90 | 0.10 |
| on | off | 0.20 | 0.80 |
| off | on | 0.20 | 0.80 |
| on | on | 0.10 | 0.90 |

Figure 3.8: The figure shows the DAG of the Bayesian network with 4 random discrete valued gene variables and the conditional probability tables related to each node in the DAG. Note that G1=Gene1, G2=Gene2, G3=Gene3 and G4=Gene4 (Steele 2010).

### 3.6.1 Model selection

Model selection can be considered as the qualitative step. It consists on learning the BN structure, that is to identify a graphical model which best fit the data given as input. Although several algorithms have been developed over the years, they all fall under three main categories: *constraint*-based, *score*-based and *hybrid*.

**Constraint-based** algorithms aim to learn the graph structure by exploring conditional independences between variables. The first algorithm of this class is called Inductive Causation (IC) and was implemented by Pearl et al. (1991).
IC firstly determines the skeleton of the network identifying all the connections between the variables regardless of its direction through statistical tests for conditional independence. If there is an edge between variables $A$ and $B$ it means that $A$ and $B$ are dependent and cannot

be independent given any subset of the variables. Then, it searches for the v-structures (two non-adjacent nodes are not independent conditional on a third one). Last IC iterates on each arc and derives the orientation to obtain the completed partially directed acyclic graph.
The IC algorithm is impracticable for real world problem due to the exponential number of possible combination of conditional independencies. Therefore, have been developed computationally more reasonable algorithms such as: PC algorithm (Glymour et al. 2001) where a backward selection from a saturated (fully connected) graph is applied; Grow-shrink (GS) (Margaritis 2003), Incremental Association (IAMB) (Tsamardinos et al. 2003), Fast Incremental Association (Fast-IAMB) (Yaramakala & Margaritis 2005) and interleaved Incremental Association (Inter-IAMB) (Tsamardinos et al. 2003).

**Score-based** learning algorithms generally consist in building a set of possible networks each with a corresponding score reflecting how well it fits the data and then select the one with the higher score. The first and most popular of these class of algorithms is hill-climbing which can be performed with either random restart or tabu search (Bouckaert 1995). Hill climbing initiates an empty network and then apply the operations *add*, *remove* and *reverse* to the edges. At each iteration a score is computed to determine if the new network fit the data better than the previous one. The algorithm stops when there is no more improvement and the final network is selected.
The score is usually calculated with the Bayesian Information Criterion (BIC) (Schwarz et al. 1978), but another popular option is the Akaike Information Criteria (AIC) (Akaike 1974). The BIC is a combination of a log likelihood model and a penalization term which penalizes complicated models against simpler ones:

$$BIC = logP(\theta) + logP(\theta|D) - 0.5 \times k \times log(n)$$

where $\theta$ represents the model, $D$ the data, $k$ the number of parameters and $n$ the number of observations (sample size). Similarly, the AIC shows a penalization term of $2k$ instead of $0.5 \times k \times log(n)$
Because the BIC takes into account the number of observations ($n$) it is more suitable to use in the case of microarray data.
Other score-based algorithms apply genetic algorithms (Larranaga et al. 1997) or simulated annealing (Bouckaert 1995) to overcome issues with local optima.

**Hybrid structure** algorithms are, as the name suggests, a combination of model selection and score-based. Some examples are Sparse Candidate algorithm (SC) (Friedman et al. 1999)

and the Max-Min hill-climbing algorithm (MMHC) (Tsamardinos et al. 2006).

### 3.6.2 D-separation, Markov property and conditional independence

Once the structure of the network has been detected it is necessary to move our attention to the quantitative aspect of the network analysis.
Before we explain how to learn the parameters of the network we need to illustrate few concepts fundamental in Bayesian networks analysis.
The Directed Acyclic Graph (DAG) of a BN represents the set of conditional independence relationships, which are explained by the directed separation (d-separation) criterion illustrated in Pearl (1988). Given three subsets of disjoint nodes $V_1$, $V_2$, $V_3$ in a DAG, $V_3$ 'd-separate' $V_1$ from $V_2$ if among all the arcs between $V_1$ and $V_2$ there is one node v that satisfies one of the following:

- v has converging arcs (all the arcs from the adjacent nodes point to v) and none of v or its children are in $V_3$;
- v is in $V_3$ and does not have converging arcs.

Considering a generic network, the Markov blanket of a node is the structure including the node's parents, its children and the other parents of its children. The node is dependent from the nodes in the Markov blanket and independent from all other nodes in the network. This implies that it is possible to calculate the distribution of each node in the network by simply considering the joint distribution of the variables in the Markov Blanket. We declare $B$ as a BN with respect to the graph $G$ if each node in the network is conditionally independent of all other nodes, given its Markov blanket. Consequently, we define the Markov property of BNs:

*each variable is conditionally independent of its non-descendants given its parents.*

The Markov property of Bayesian networks allows to represent the global distribution of the network $X$ as the product of the conditional probability distributions that are the local distributions associated with each variable $X_i$. This is a direct application of the chain rule (Korb & Nicholson 2003) so that for discrete random variables, the factorization of the joint probability distribution $P(X)$ is given by:

$$P(X) = \prod_{i=1}^{p} P_{X_i}(X_i|\pi_{X_i}) \tag{3.8}$$

where $\pi_{X_i}$ is the set of parents of $X_i$. Equally, for continuous random variables, the joint density function $f(X)$ is given by:

$$f(X) = \prod_{i=1}^{p} f_{X_i}(X_i|\pi_{X_i}) \tag{3.9}$$

In addition, Markov blankets ease the comparison of Bayesian networks using graphical models based on undirected graphs called Markov networks (Whittaker 2009). A DAG, in fact, can be transformed in the undirected graph of the Markov Network by applying the *moralization* transformation (Nagarajan et al. 2013) which links non-adjacenct parents that share a common child. The obtained graph is called *moral graph* (Castillo 1997). Furthermore, because the local distributions involve only fewer variables compare to the whole network, it also reduces the curse of dimensionality problem.

Although BNs are defined using terms of conditional independence, there is no implication that the arcs should represent cause-effect relationships. It could be argued that a 'good' BN represents the causal structure of the data it is describing (Pearl et al. 2009), however in this research the links between variables are not considered as such.

The aim, now, is to calculate the parameters of the conditional probability distribution/tables for each node in the network, that best fit the data. Given a probability distribution $X$ and a dataset $D = x_1, x_2, ..., x_n$ we want to learn a set of parameters $\theta$ for $X$ that maximizes the likelihood ($L(\theta)$) that the data D comes from X. There are two main approaches to estimate the parameters either through classic bayesian estimation or maximum likelihood: $\arg\max_\theta L(\theta) = \arg\max_\theta P(D|\theta) = \arg\max_\theta \prod_{i=1}^{n} P(x_i|\theta)$. Although this simplification, parameters estimation for high dimensional data such as microarrays, may still be problematic.

### 3.6.3 Bayesian Network Inference Algorithms

Once the structure of the networks has been chosen and the parameters learnt one more function of bayesian networks that is extremely useful to researchers is $inference$. Bayesian inference also called probabilistic reasoning or belief updating allows to determine the state of a set of variables given the state of others as evidence. The peculiarity is that it evaluates the evidence and assign the state value no matter if that state has been observed already or not. This function is, then, crucial in terms of reducing the number of additional experiments.

The strength of inference is then to determine the state of a variable beyond the observations, computing the posterior probabilities or densities (Pearl 1988, Koller & Friedman 2009). Given the Bayesian network $B$ with graphical structure $G$ and parameters $\Theta$ we want to analyse the effect of a new evidence $\mathbf{E}$ on the distribution of $\mathbf{X}$ using the knowledge encoded in $B$, which means to analyse the posterior distribution: $\mathrm{P}(\mathbf{X}|\mathbf{E},B) = \mathrm{P}(\mathbf{X}|\mathbf{E},G,\Theta)$.

Considering the example in Figure 3.8 we want to infer, using the *logic sampling* method (Henrion 1988), the values of Gene 2 and Gene 3 when it is observed that Gene 4 = *on*. It is clear from Figure 3.9 that both genes have a much higher probability of being *on* (green bars) when Gene 4 is observed to be expressed (*on*).

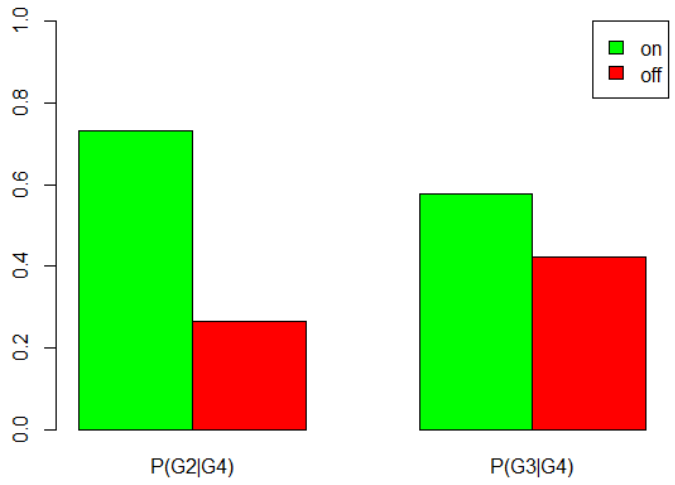



Figure 3.9: The figure shows the probability of Gene 2 and Gene 3 being *on* or $off$ when it is observed that Gene 4 = *on*. Note that Gene 1 = G1, Gene 2 = G2, Gene 3 = G3 and Gene 4 = G4.

### 3.6.4 Prediction

Inference can be used to evaluate the performance of a BN by predicting a node values on a new independent dataset. In fact, if a BN predicts better it means that there is less overfitting on the training dataset and the BN can be considered more robust and reliable.

The prediction can either be done on continuous or discrete data. While on discrete data BN can predict assigning a specific value, on continuous it can only indicate a range. Continuous data requires calculation of conditional probability distribution which are computationally more costly and less precise in the prediction. Therefore, it is often better to compute inference in a discrete environment.

Although, inference can measure the performance of the network, to measure the goodness of inference it is usually calculated the *prediction accuracy* which is the proportion of variables' (genes) values that have been predicted correctly. In this research the prediction accuracy is

calculated after applying the leave one out cross validation (LOOCV) technique on each dataset under analysis and the others (externals). Given the $m$ studies and $n$ genes within each dataset LOOCV uses $m$-1 studies as a training set and the remaining one as test set.

### 3.6.5 Application to gene expression profiles

Because of the way Bayesian networks present information and knowledge they are a popular tool also among non technical practitioners. This of course make them extremely practical in the analysis of all kind of biological data including gene expression profiles. The very first attempt to exploit BNs in biology was done by Friedman et al. (2000) by building a framework for discovering interactions between genes based on multiple expression measurements (microarrays). Later Hartemink et al. (2002) develop a method for elucidating genetic regulatory networks using Bayesian networks and genome-wide data describing gene expression and transcription factor binding location. Gyftodimos & Flach (2002) implement a specific case of BNs called Hierarchical Bayesian Networks to deal with structured data and allow representation of complex hierarchical domains. Pe'er et al. (2006) identify a constrained family of Bayesian network structures suitable for gene expression data and implement a search algorithm that utilizes these structural constraints to find high scoring networks from data. In Sachs et al. (2005) they perturb cellular signaling networks and simultaneously measure multiple phosphorylated protein and phospholipid components in thousands of individual primary human immune system cells. They then apply Bayesian networks to identify both the traditional pathways but also predict novel pathways that they verified experimentally.

## 3.7 Conclusion

Several algortihms and combination of them have been developed over the last few decades to detect the best possible model to build Gene Regulatory Networks from the data. Each method with its strengths and weaknesses can be applied to discover the underlying mechanism and the relationship hidden in the data under analysis.
In the next chapter we describe a novel approach that explores some of the methods described in this chapter in order to identify study-specific gene regulatory networks. In addition validation techniques are applied to refine and support our findings.

# Chapter 4

# Analysis of synthetic data

## 4.1 Introduction

Organisms of any level of complexity (from bacteria to mammalian) developed a large set of internal mechanisms during evolution, either the normal functioning or as a response to external or internal stimuli that differ from normal activity. While many mechanisms, necessary for survival, carry on mostly unchanged under all conditions the organism is subjected to (e.g. cell metabolism), others are triggered or modified only when some event external or internal to the organism (environmental changes, stress, cancer, etc.) happens.
Organisms' mechanisms, in general, involve large numbers of interactions between thousands of genes resulting in highly complex networks. However, all the necessary information are usually fully explained by only a few genes and much smaller networks. Therefore, networks with many thousands of connections can be rightly reduced in size by few orders of magnitude without loss of information (Gillis & Pavlidis 2012).
Some conditions might trigger similar mechanisms (more or less based on how similar the conditions are) that researchers identify using consensus networks analysis that identifies links in common over a number of studies (Swift et al. 2004). Highlighting the similarities, though, can overshadow or even hide what is unique and typical to one specific condition. Biologists are clearly interested in what these similarities are but they are also interested in identifying the condition-specific mechanisms/gene-paths of which knowledge will help in their detailed understanding.

The novelty of our approach is the ability to semi-automatically identify subnetworks that are unique to a number of independent studies (*unique-networks*). Identification of unique networks can lead to a better understanding of behaviours relevant to that condition. We

develop a pipeline that, given microarray raw input data from multiple independent studies, firstly selects a subset of relevant genes, groups the studies with similar mechanisms, identifies the mechanisms that are specific for each group of studies ($study - cluster$), and validates them biologically and statistically through *inter* and *intra* study-cluster prediction.

In this chapter we present the pipeline and evaluate its performances by using synthetic datasets. This chapter is organized as follows. Section 4.2 describes the pipeline. Section 4.3 explains the structure of the simulated data and the pre-processing. Section 4.4 shows the results. Section 4.5 compares the results obtained by using the biclustering technique. Finally Section 4.6 summarises and discusses the findings.

## 4.2 Methods

The pipeline described here, which we call UNIP (Unique Network Identification Pipeline) aims to discover what genes and the relationships between them are specific to the study or group of studies under consideration. To achieve this goal, we, first, identify the variables/genes that uniquely appear in the GRN of one study or one group of studies, and then derive study-specific gene regulatory networks (*unique-networks*). *Unique-networks* can be seen as the sub-GRNs specific to the group of studies. This helps biologists to identify what are the typical mechanisms that characterize one study rather than another.
To achieve this we need to sequentially go through a list of steps, each with a specific purpose. Figure 4.1 shows a schematic representation of the steps involved, each explained in the following sections.

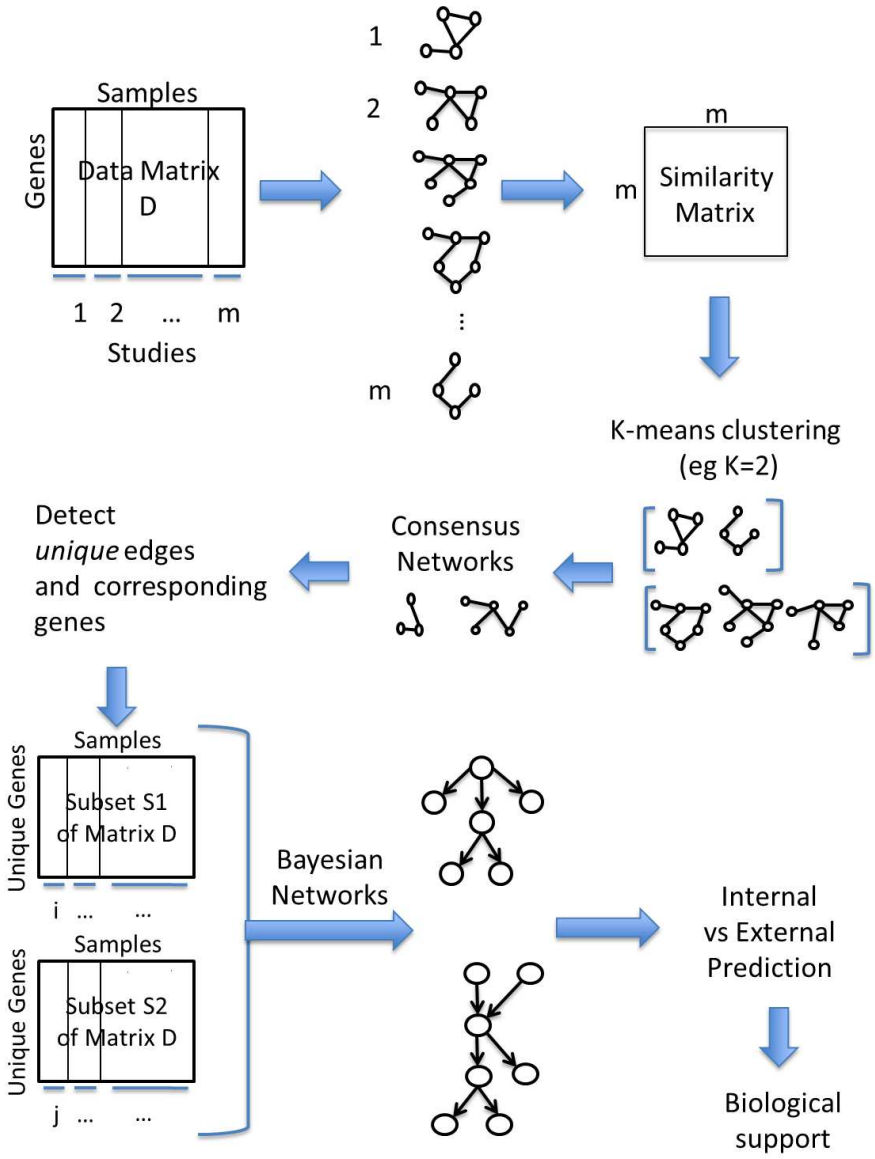



Figure 4.1: Pipeline overview. A schematic overview of the sequence of steps forming the pipeline.

### 4.2.1 Single study glasso network

Each organism has underlying mechanisms which apply under normal conditions. When the same organism is subjected to different conditions (stress, environmental changes, etc...) then it will need to respond to the change resulting in new paths of genes being highlighted. This results in new underlying mechanisms and/or changes in already active mechanisms. So, different experimental conditions can present different Gene Regulatory Networks (GRNs).
Given $m$ independent studies of the same organism each with the same genes but a different number of samples we merge them together in a data matrix $D$. As we want to identify networks that go beyond simple pairwise relationships, for each of the $m$ studies we build a Gene Regulatory Network (GRN) by either applying *glasso* (see Section 3.5.4) or *bayesian networks* (see Section 3.6) both being able to model more complex interactions.

### 4.2.2 Graph similarity

We integrate several microarray datasets in order to compare different studies. Some studies will still have some network paths in common (if the genes are regulating one another under those conditions). For example, heat stress and drought stress will have gene pathways in common with other stress-related studies. So, at this point of our pipeline the objective is to automatically detect mechanisms common to similar studies and cluster them using an adaptation of the sensitivity metric (Baldi & Brunak 2001) to obtain a restricted number of *study-clusters*. Given two networks, network 1 (NW1) and network 2 (NW2), the connections that two networks have in common are the true positives, those that are in NW1 but not in NW2 are the false positives and those not in NW1 but in NW2 are the false negatives. Therefore, we analyse the connections in common between two study-networks and build a contingency table. To verify the reliability of the clusters we compare the results with the description of the studies available when downloaded from public databases such as ArrayExpress (Parkinson et al. 2007). We explored a number of clustering techniques but found that k-means (Hartigan & Wong 1979) generated the most convincing *study-clusters*.

### 4.2.3 Consensus networks and unique-connections

In the process of identifying unique-networks we first build the consensus network for each study-cluster as a representative of the general mechanism for that group of studies (Steele & Tucker 2008). This step identifies the network pathways that are common to a $thr$ % of the networks in the study-cluster. Based on the data and the flexibility required we tune the threshold $thr$. $thr = 100\%$ implies full consensus, meaning that only those edges identified in

every single-network in the cluster are selected to build the consensus study-network. $thr < 100$ implies partial consensus and increases or decreases the size of the consensus study-network.

Once we have one *consensus-study network* per *study-cluster*, we select only those edges that exist in the consensus-study network in consideration, but not in the other consensus-study networks. We call these *unique-connections*. The resulting list of nodes involved in the unique-connections is used to build the *unique* Bayesian networks as explained in detail in the following sections.

### 4.2.4 Unique Networks

In the sections described above, we cluster the studies in $k$ groups each identifying one generic conditions. For each study-cluster a consensus network is constructed that represents the underlying gene regulatory mechanism(s) in common for that group of studies. This will allow us to build more robust GRNs for each study-cluster. As explored in Chapter 2, consensus networks together with consensus clustering are popular approaches but the focus of this research is to create and apply the concept of *unique-networks*.

Given a generic graph $G = (V, E)$. We have $m$ fixed graphs $G_i$ such that $G_i = (V, E_i)$ , where $V = 1, ..., n$ is the set of vertices(nodes) of the graph and $E_i$ the set of edges. $E_i = \{e_i\} = \{(u_{i1}, v_{i1}), ..., (u_{ik_i}, v_{ik_i})\}$, $k_i = \left|E_i\right|$ and $k_i \leq n(n-1)/2$. We define the unique function as $\Phi : G \mapsto G$, where, given $\hat{E}_i = \bigcup_{j=1, j\neq i}^{m} E_j$

**Definition 1:** We define a function $\Phi(G_i)$ such that $\Phi(G_i) : (V, \{e_i : e_j \in E_i \text{ and } e_j \notin \hat{E}_i\})$

In simple words, the unique function returns the *unique-connections* ($\Phi(G_i)$) that are the same set of edges in the consensus network ($G_i$) of the study-cluster ($i$) under consideration except those that also exist in the remaining consensus networks ($\hat{E}_i$). An example is explained in Figure 4.2. Given three consensus study-network, we are interested to identify the unique-connections for the first study cluster. The dashed lines in consensus study-networks 2 and 3 indicate the connections that each network has in common with consensus study-network 1 and therefore will not be included in the unique-connections set. Genes not involved in any unique-connections will also be discarded.

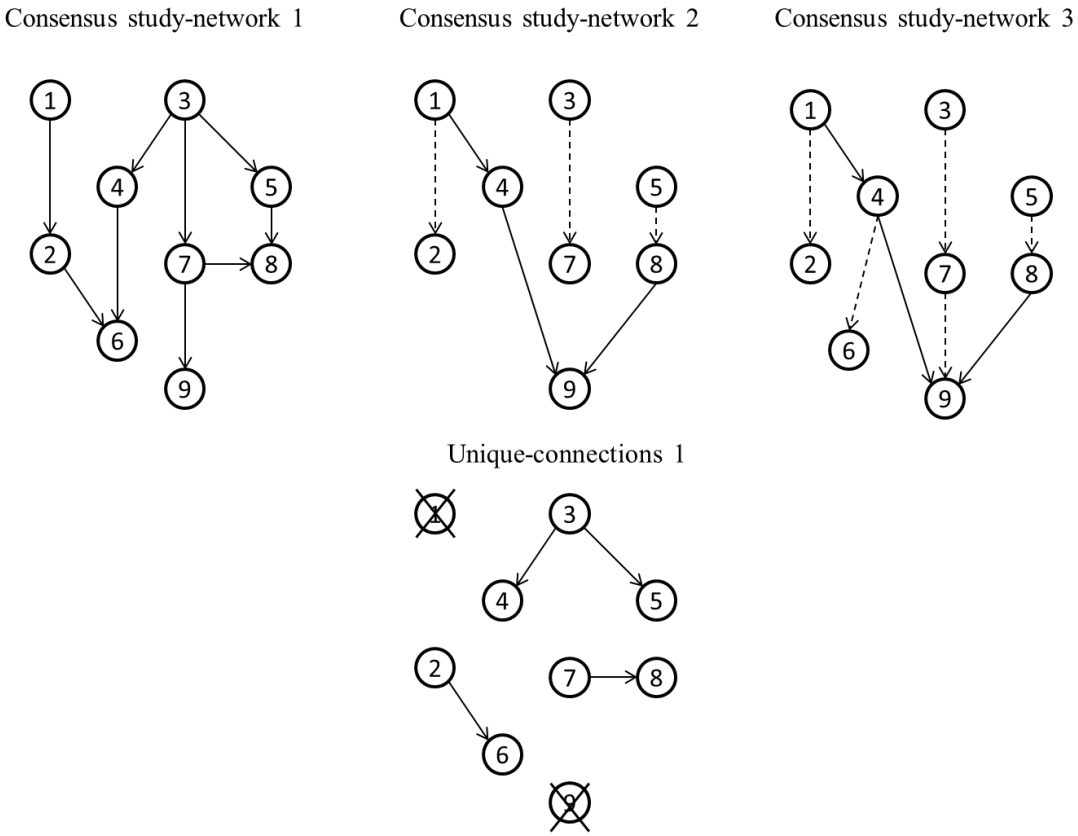



Figure 4.2: Example of unique-connections construction approach. Given three study-clusters each with a corresponding consensus study-cluster, the unique-connections for study-cluster 1 are the set of connections that are unique for that consensus study-network and do not appear in consensus study-networks 2 and 3. Dashed connections indicate the connections that each network has in common with consensus study-network 1 and therefore will not be included in the unique-connections set. Genes not involved in any unique-connections will also be discarded (genes crossed out)

### 4.2.5 Bayesian unique-networks

We choose to validate the networks through prediction using Bayesian Networks (BNs) which naturally performs this using inference (see Section 3.6.3). We want to compare accuracy for a network's own study cluster versus other clusters to highlight its 'uniqueness'. BNs (Heckerman et al. 1995, Friedman et al. 2000) are a class of graphical models that represent the probabilistic dependencies between a given set of random variables. A Bayesian network has a set of variables called nodes and a set of directed edges between variables called arcs. The nodes and arcs

together form a *directed acyclic graph* ($DAG$) $G$=($V$,$A$). Each variable in the network has an associated conditional probability table of itself given its parents. Having reduced the number of variables and samples by identifying the unique networks, we build one BN for each of the study-clusters previously identified based on the genes with unique edges in the consensus-study networks. To do this we use the *hill climbing* method (Bouckaert 1995) and the BIC score.

### 4.2.6 Prediction accuracy

Now that we have one *unique-network* per study-cluster, we are interested in finding the most predictive (how well it predicts other expression level values) and predictable (how well its expression level values are predicted) genes within (intra) and outside (inter) the study-clusters using the leave one out cross validation technique (refer to Section 3.6.4). The idea is that genes that are predictive or predicted better within the selected study-cluster than on other studies are more likely to be relevant to the *unique-network*. Given the $m$ studies and $n$ genes within each studies-cluster we use $m$-1 studies as a training set and the remaining one as test set. We employ the inference method described in Højsgaard (2012) which, given the $n$-1 genes, predicts the expression value of the one left out. Since, at this stage, we are dealing with discrete data we can compare the predicted value of the left out gene with its real value. The algorithm returns 1 if the real value and the predicted one correspond and zero otherwise. We do this within all the study-clusters and for all possible combinations of training and test sets of studies and genes. Finally, we average the amount of correctly-predicted values among the total predictions to obtain the *correct-prediction* for each gene.

### 4.2.7 Biological support

Having identified the study-clusters and, in turn, the study-specific mechanisms within the *unique-networks*, we explore the biological meaning behind them by using external tools such as Mapman (Thimm et al. 2004), the AIC-MICA method (Lysenko et al. 2011) or GeneCards encyclopaedia (Safran et al. 2010) as well as gaining the help of biologists expert in the field. For the case of synthetic data this analysis is not necessary as we are able to directly analyse the id of unique genes.

### 4.2.8 Biclustering

Part of our pipeline's purpose is to identify groups of genes involved in the unique mechanisms specific for a set of conditions (study-cluster) to build unique-networks. Therefore, we compare our results (the discovered clusters and their associated networks) with the closest technique

we found that already performs what we are trying to achieve. *Biclustering* techniques aim to cluster samples and genes simultaneously (Cheng & Church 2000) but it is important to highlight that biclustering works on each sample and not on the studies. There are various implementation variants in the literature for biclustering (Madeira & Oliveira 2004) but for this work we specifically choose a method called $Questmotif$ which is based on the framework described in Murali & Kasif (2003), for the simulated (categorical) datasets and the $BCS$ method for the real datasets of wheat and *Fusarium*. BCS is a state-of-the-art method that normalizes the data matrix and looks for checkerboard structures using the well-known technique of singular value decomposition in eigenvectors applied to both rows and columns (Kluger et al. 2003). Both BCS and Questmotif are implemented in the R package *biclust* (Kaiser et al. 2009).

## 4.3 Data structure

To explore, quantify and test the performance ability of UNIP it is necessary to apply it to well-known and easily modifiable datasets. Therefore we now examine and verify its performance on simulated data before we explore real microarray datasets (Chapters 5 and 6).

Following the steps described in Section 4.2 and the schematic flowchart in Figure 4.1, we first need to build a data matrix, resembling microarray characteristics, that will work as input for our pipeline. Synthetic data differs from real microarray datasets exhibiting no noise and a much smaller set of variables. While the first one is addressed adding random noise to the synthetic data, the latter is overcome knowing that real datasets always require a feature-selection preprocessing step that massively reduces the original set of variables to a number close to the dimension of synthetic data. Synthetic data variables are categorical while, on the other hand, real microarray datasets are continuous. For the purpose of this work the nature of the variables only slightly affects the pipeline which sees the use of bayesian network instead of glasso for categorical data, but does not affect the final results. Furthermore, discrete and categorical data are easier to work with when calculating the prediction accuracy. Bayesian network, in fact, returns a range of values when predicting a continuous variable but a distinct value for a discrete/categorical one.

From the Bayesian Network Repository (Scutari 2014), we select the networks: Alarm (Beinlich et al. 1989) (Figure 4.3), Insurance (Binder et al. 1997) (Figure 4.4) and Child (Spiegelhalter & Cowell 1992) (Figure 4.5) with 37, 27 and 20 nodes respectively. The possible number of states of the variables vary from 2 to 6. As a result, the chance to correctly predict them varies from $\frac{1}{2}$ to $\frac{1}{6}$. The variables in the alarm networks are categorical with a maximum of 4 possible states. Out of 37 variables, 13 have only two possible states, 17 have 3 possible states and only 7 have

4 possible states. Insurance network has 8 variables out of 27 with 2 possible states, 5 with 3, 12 with 4 and 2 with 5. Finally, Child on a total of 20 variables have a maximum of 6 possible states: 8 variables with 2, other 8 with 3, 1 with 4, 2 with 5 and 1 again with 6.
For each network we download the structure and the corresponding conditional probability tables and then simulate to randomly take 200 samples. At this point we have three datasets of sizes: $37 \times 200, 27 \times 200$ and $20 \times 200$. Each dataset is representative of a different underlying structure (much like a gene network under different experimental conditions).
The 84 total variables are far from the usual microarray number of variables (tens of thousands of genes), but in real dataset is usually required a pre-processing step which apply some feature selection technique to reduce the number of variables to a computationally reasonable number. Therefore, in this dataset we assume that the 84 total variables are already the results of the variable selection which we won't explore any further in this section, since this is not the main focus of our work.

Microarrays can be biased and noisy so we need to mimic this behaviour with our simulated data. Therefore, we perturb the datasets adding noise and creating what we will call from now on *big matrix*.
*Big matrix* represents our simulated data and is composed of 9 smaller matrices. Three matrices are the datasets sampled from the networks while the remaining six are randomly created based on the values of the original variables/nodes. If we consider the *big matrix* as a 3x3 block matrix composed of nine blocks, each row of the *big matrix* has one sampled dataset and two random ones. Figure 4.6 shows a representation of the *big matrix* where the capital letters A, I and C indicate the datasets of Alarm, Insurance and Child respectively while R represents random values (noise).
This structure simulates one organism in which specific group of genes are involved in the mechanism(s) of one or a group of conditions. So, in Figure 4.6 the genes (rows) from 1 to 37 are characteristics for the specific mechanism(s) described in the condition(s) represented by the samples (columns) 1 to 200.

In order to test the robustness of our pipeline we gradually introduce noise by swapping actual samples with random values. We first analyse the *big matrix* with no noise (0%). Then, we gradually introduce an increasing percentage (from 10% to 90% with intervals of ten) of random samples of the total (noise) and decide to focus on what we find to be the most revealing noise-levels: 10%, 50% and 90%.
We gather the 600 samples in 15 studies of 40 samples each so that each column-block of *big matrix* contains exactly 5 studies of 40 samples each and 84 variables/nodes (37+27+20). In Table 4.1 we show the correspondence of studies and original networks.

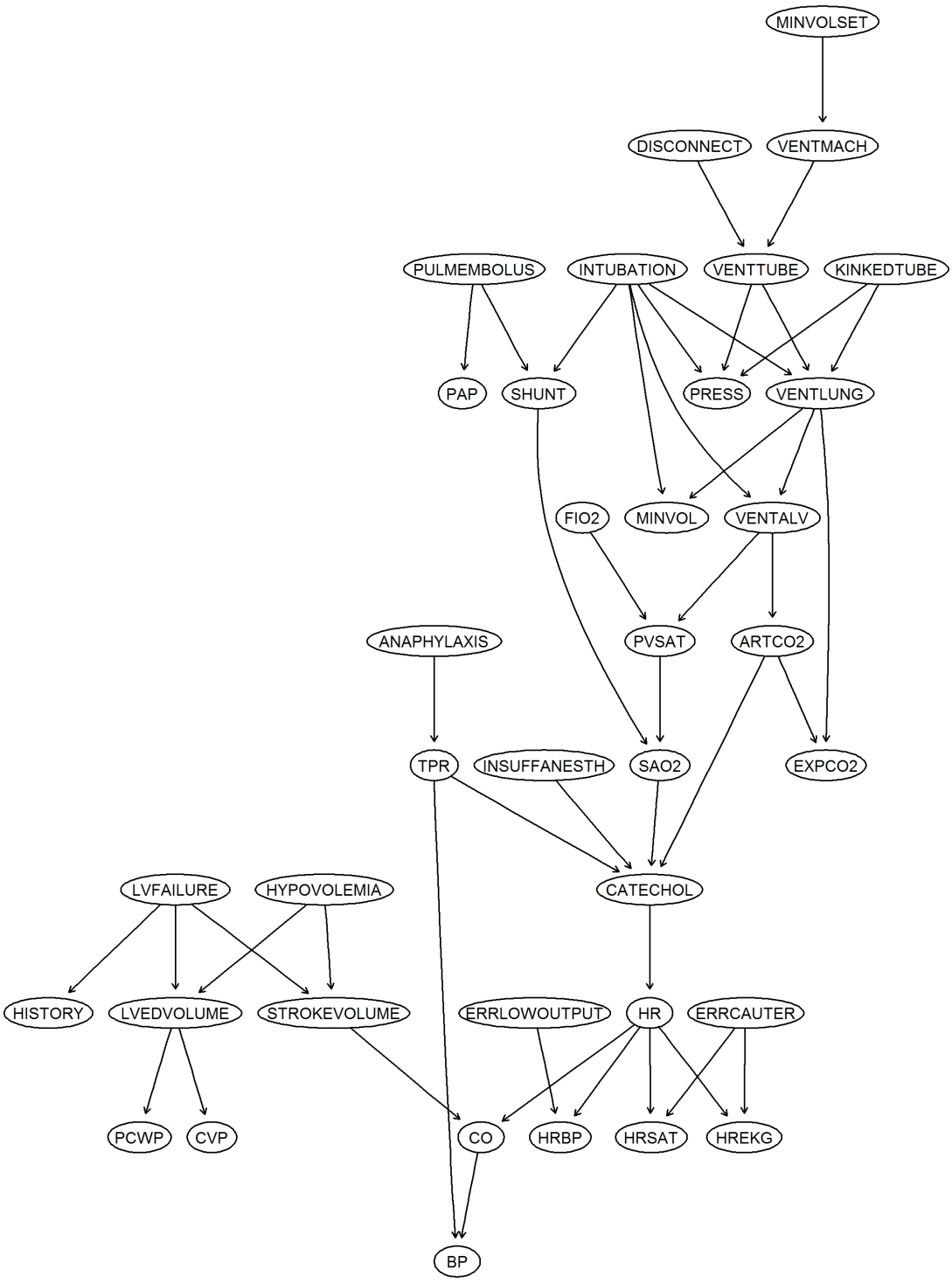



Figure 4.3: Original structure of the Alarm network.

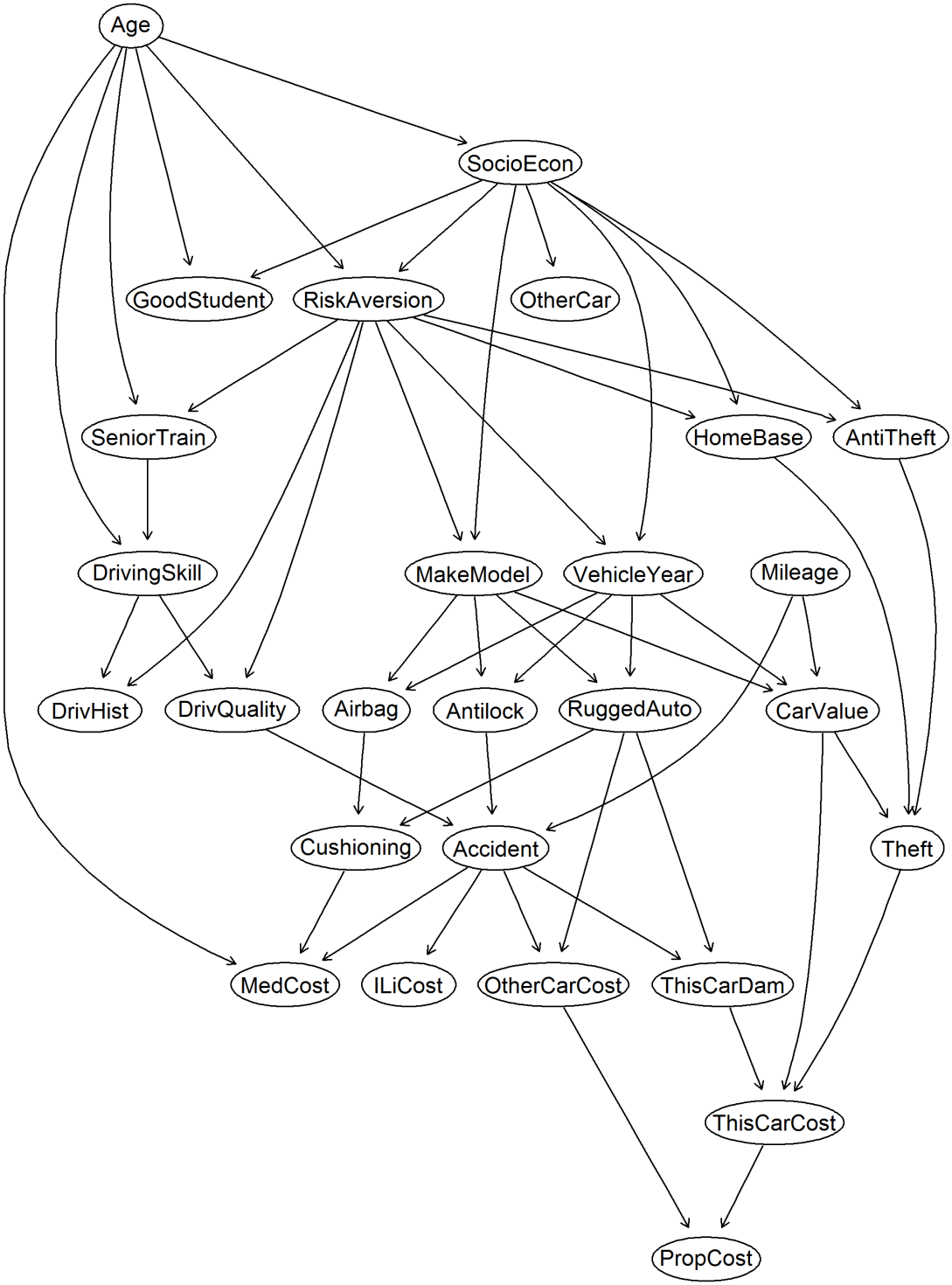



Figure 4.4: Original structure of the Insurance network.

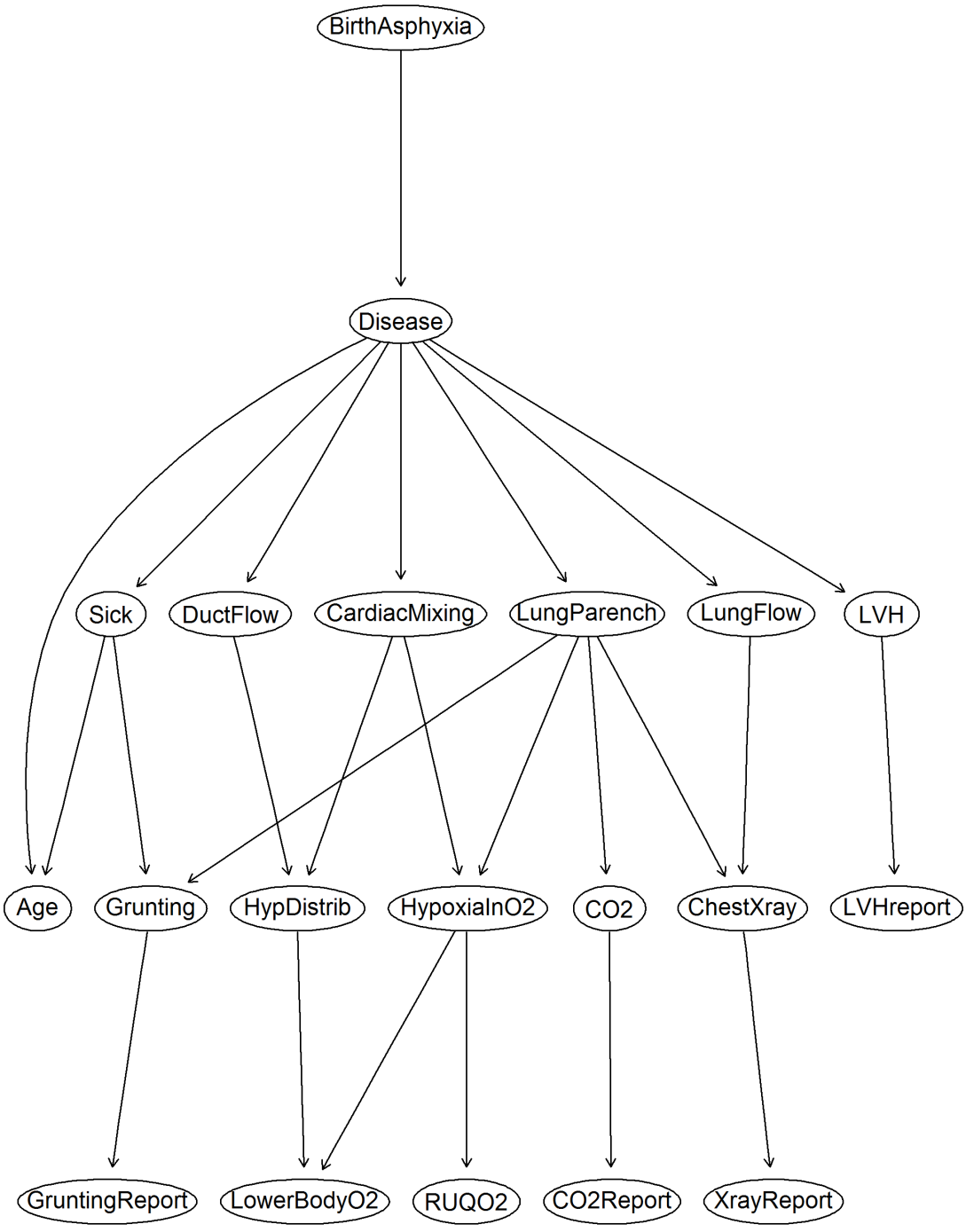



Figure 4.5: Original structure of the Child network.

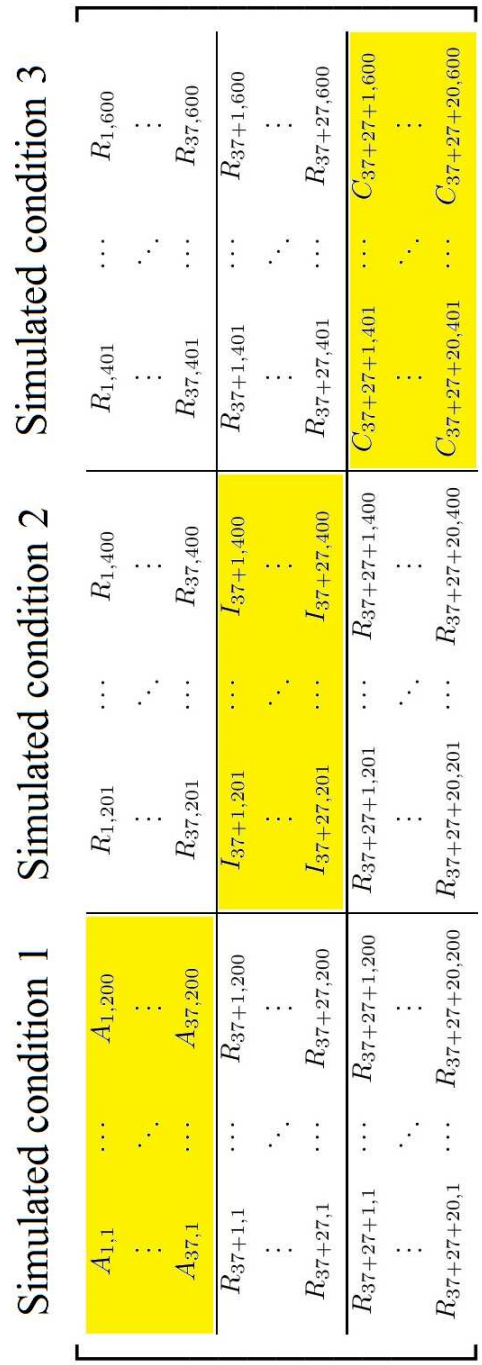



Figure 4.6: *Big matrix* constructed from the datasets generated from the three networks and six randomly generated datasets which represent the noise. The shaded regions indicate the non-noisy datasets generated from Alarm, Insurance and Child networks (respectively A, I and C in the figure). While R indicates random values (noise).

| Studies | Network |
|---|---|
| 1,2,3,4,5 | Alarm |
| 6,7,8,9,10 | Insurance |
| 11,12,13,14,15 | Child |

Table 4.1: Simulation studies generated independently from the three networks in consideration.

## 4.4 Results on simulated data

Once the *Bigmatrix* is set, the UNIP pipeline builds one Gene Regulatory Network (GRN) per study for a total of 15 networks. Here, due to the categorical nature of the data, we use Bayesian networks and the smaller number of variables rather than *glasso*. We learn the structure using the score-based approach hill climbing (Russell 2003) combined with the BIC score.
Ideally, this pipeline will cluster the studies as they belong to the original networks and detect, for each study-cluster, the variables that are truly involved in each of them. We calculate the graph similarity metric described in Section 4.2.2 and apply the k-means (Hartigan & Wong 1979) algorithm with $k = 3$ (3 is the number of original networks) to cluster the studies in the *bigmatrix*.
Figure 4.7 shows the clusters' arrangement for the original data and for the data with an increasing amount of noise (from 10% till 90%). While at 10% of the noise the study-groups detected by our pipeline reflect the real studies arrangement, an increase to 50% disrupts the process and shuffles the studies. As expected, the noisier the input is, the more mixed the study-groups are. 0% and 10% of noise are equivalently good and both perfectly separate the study into the real clusters. When the noise increases to 50% only two studies gets mixed in the wrong cluster. Finally, in the case of 90% of noise the clusters are extremely mixed with each other. This pre-analysis already gives us a good idea of how robust our unique network pipeline is per each level of noise.

| Noise % | Simulation Studies | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 [correct] | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 10 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 50 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 90 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

Figure 4.7: Study-clusters for the original data (0% of noise), 10%, 50% and 90% of noise. The studies' number highlighted with the same colour belong to the same cluster.

### 4.4.1 Unique-networks and intermediate results

For each cluster of networks (study-cluster) we build the consensus (where links in the network must exist in all networks for that cluster) and identify the unique-connections. Finally, considering only the genes involved in the unique connections we apply again Bayesian networks to obtain the unique-networks (where links must only occur in that cluster). *Big matrix* contains all 84 variables from all the three networks, which leads to the fact that all the unique study-cluster networks will most probably include variables and connections that do not belong to the original structure.

Given the unique-cluster networks, the next step in the pipeline is to compare each original network with the corresponding obtained *unique* one for each level of noise. These intermediate results are visualized in Figure 4.8 and show the ability of the pipeline, at this specific stage, to detect the true positive (TP) nodes and connections between nodes as the noise increases. TPs are the number of connections/nodes in the simulated network that are also in the corresponding original network, while FPs are the number of connections/nodes in the simulated network that are **not** in the original one. This, in Figure 4.1 corresponds to the step preceding the calculation of internal vs external prediction accuracy where non-predictive variables are filtered out. The number of both TPs and FPs nodes for all the clusters only slightly increase along with noise. This is due to the fact that at zero noise the pipeline manages to already select the majority of the correct nodes.

The connections, on the other hand, behave differently. The TPs constantly decrease: only slightly between no-noise and 10% but decrease much more for noise $\geq$ 50% with almost zero at 90% of noise. FPs, instead, tend to increase very slightly for lower percentages of noise in (Alarm and Insurance). Later, when the data becomes almost completely random, the algorithm recognizes the faulty information and massively decreases the number of connections detected to zero. One way to decrease the number of FPs, especially for the nodes, would be to increase the number of samples per study in the input dataset. Some tests, we have run, proved that samples need to be more than 200 which is an extremely rare case for microarray datasets.

To summarize, at this stage of the pipeline we discovered that for low levels of noise our pipeline can robustly identify unique-networks and what is more it is also resilient to moderate noise up to 50%. Very high levels of noise, however, appear to affect the TPs and FPs of the connection identification more than the node identification.

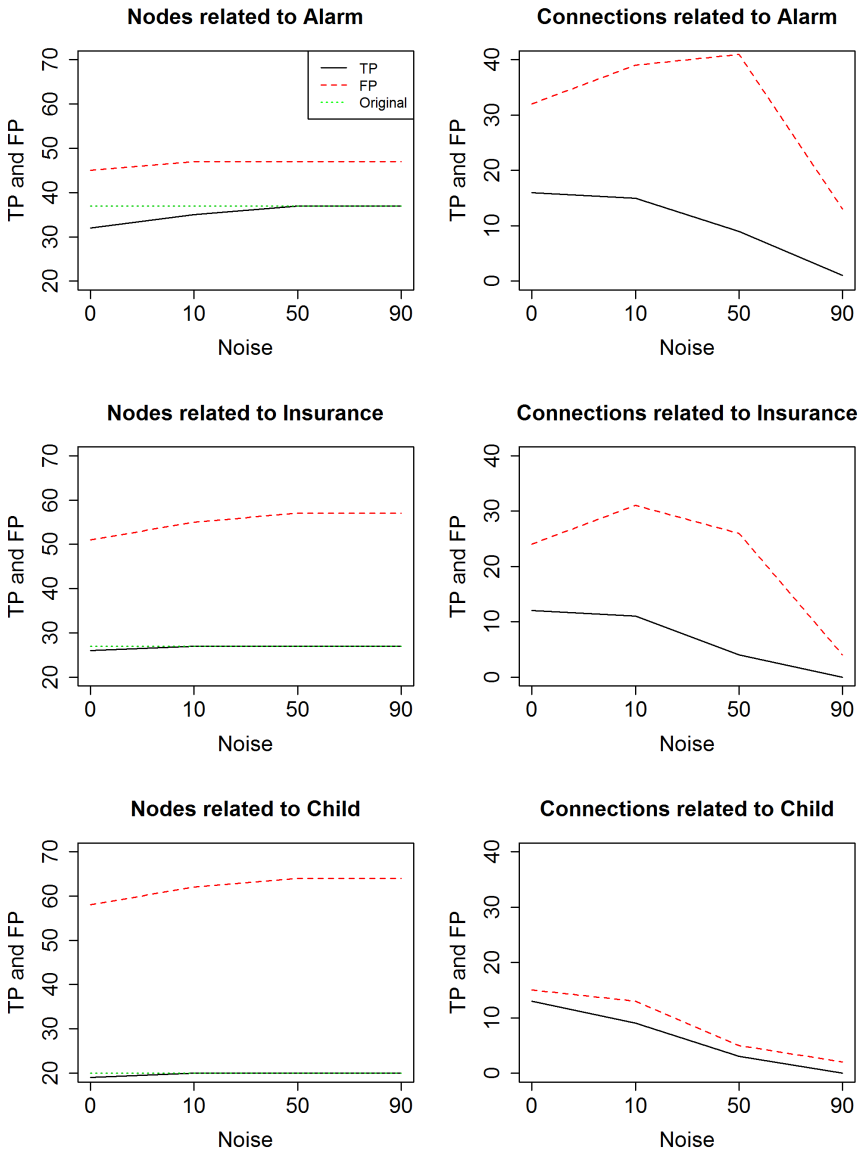



Figure 4.8: TPs and FPs vs noise **before** calculating the correct-prediction. The figures show the evolution of TPs and FPs vs noise in terms of nodes (variables involved in the discovered subnetworks) and connections between nodes. The green dotted lines indicate what is the original number of nodes. These are the partial results, prior to the filtering of the informative nodes based on the intra cluster correct-prediction accuracy (which are shown in Figure 4.9).

### 4.4.2 Prediction accuracy and final results

Finally, we calculate the *inter* and *intra* clusters prediction to validate the predictive power of the unique-subnetworks for datasets that are clustered together and to filter out any nodes that do not appear to be uniquely predictive to their study-cluster.

The possible number of states of the variables vary from 2 to 6. As a result, the chance to correctly predict them varies respectively from $\frac{1}{2}$ to $\frac{1}{6}$. So, to be able to say that one variable is predicting better than chance, its average correct-prediction across training and test sets has to be higher than its *accuracy by chance*.

The graphs in Figure 4.9 represent (in the case of 0 % noise) the boxplot of the average *correct-prediction* across training and test within each of the three study-groups, including all the variables involved in the unique network for that group. The study-clusters are listed in the titles and we can refer to table 4.1 to identify the networks they belong to. The variables involved in the unique networks for each group of studies are listed in the x axis. We clearly see groups of variables that stand out. The variables that truly belong to the corresponding real networks result in having an average accuracy above 0.6 which is significantly higher than their *accuracy by chance*. The circled variables are the ones with the highest correct-prediction and are likely to be the ones that are involved in the original networks.

Similarly, Figure 4.10 shows the distribution of the node's intra cluster correct-prediction when the noise is increased to 10, 50 and 90%. As we increase the noise, a number of things come to our attention. For lower percentages of noise, the variables' accuracy histogram shows one major peak at high correct-prediction values and another smaller peak at low correct-prediction values creating bimodal distribution. While the higher peak indicates the TPs, the lower one identifies the amount of FPs. An increase of noise, however, gives a more uniform distribution. Even for the highest level of noise there are still a good number of nodes with relatively high intra cluster (within the same *study-cluster*) correct-prediction levels. This gives us confidence that even for the noisiest datasets, the pipeline is still capable of identifying key variables. Although the clusters become incorrect they contain enough correct studies to learn predictive models.

Following the flowchart, we now select the variables that truly are involved in the network mechanism setting a threshold for the accuracy (Section 4.2.6 - Prediction accuracy). Different thresholds return a different number of TPs and FPs. Results show that for a threshold accuracy of **0.6** we obtain the best combination of TPs act while the number of TPs is very high, the number of FPs is reduced to zero. Which means that calculating the intra cluster correct-prediction allows to discard all the variables that are not involved in the original network. Figure 4.11 shows the behaviour of FPs and TPs as the noise increases, this is compared to

Figure 4.8 before filtering unpredictive variables.

As expected, when we increase the noise TPs' trend decreases while FPs slightly increases. The noisier the data are, the more difficult it is to set a threshold for the variables. The reasons for this are twofold: because the trend of FPs is higher and because both trends reach zero very quickly. Even if the number of TPs detected by the pipeline decreases when the noise level exceed 0.5, the number of FPs remains close to zero for all level of noise. This shows that even for extremely noisy and biased input data, the pipeline is still able to detect variables that are highly important.

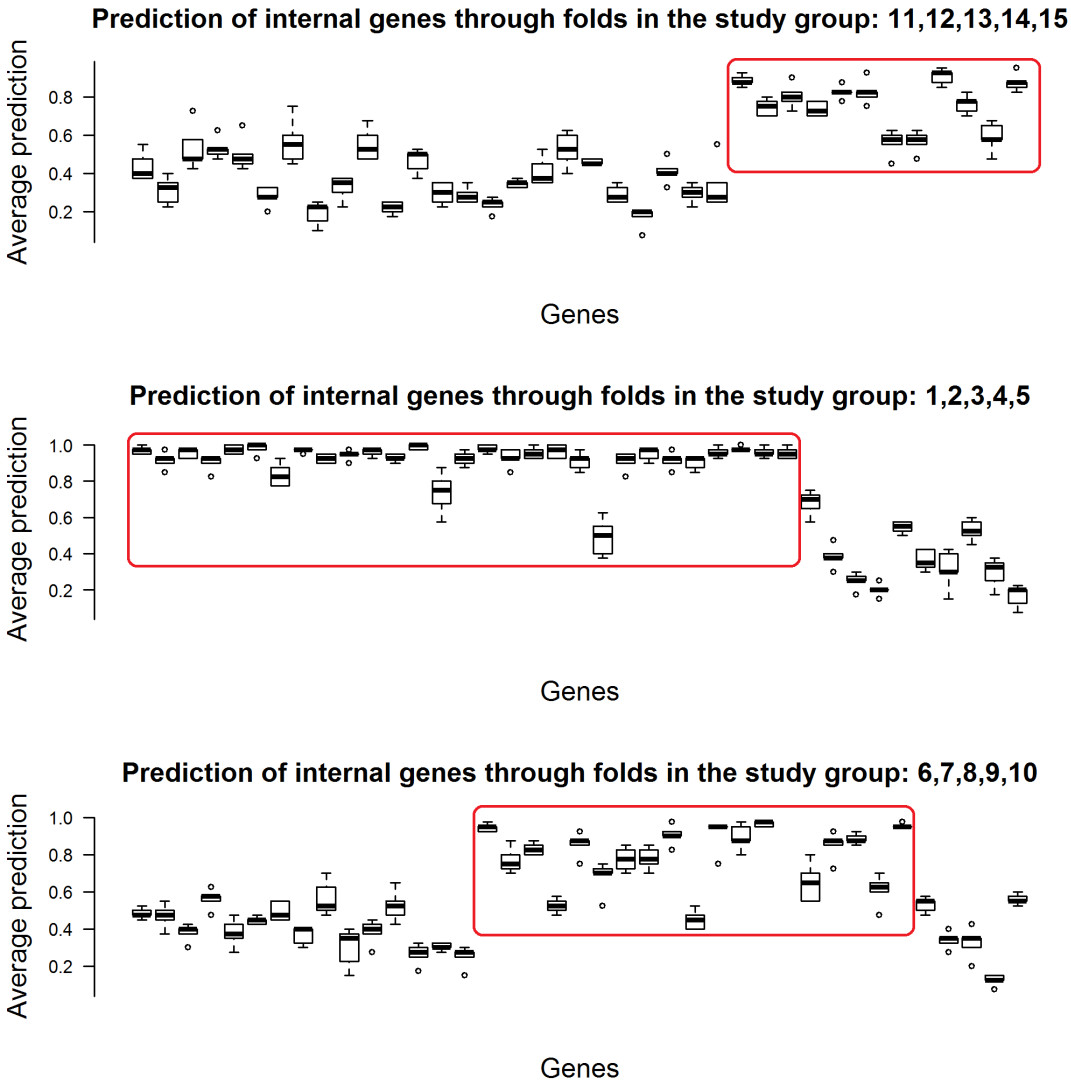



Figure 4.9: Intra cluster correct-prediction for simulated data. The figure shows the boxplots of the intra cluster correct-prediction (calculated within the same cluster using cross-validation) for the simulated dataset in the case of 0% of noise.

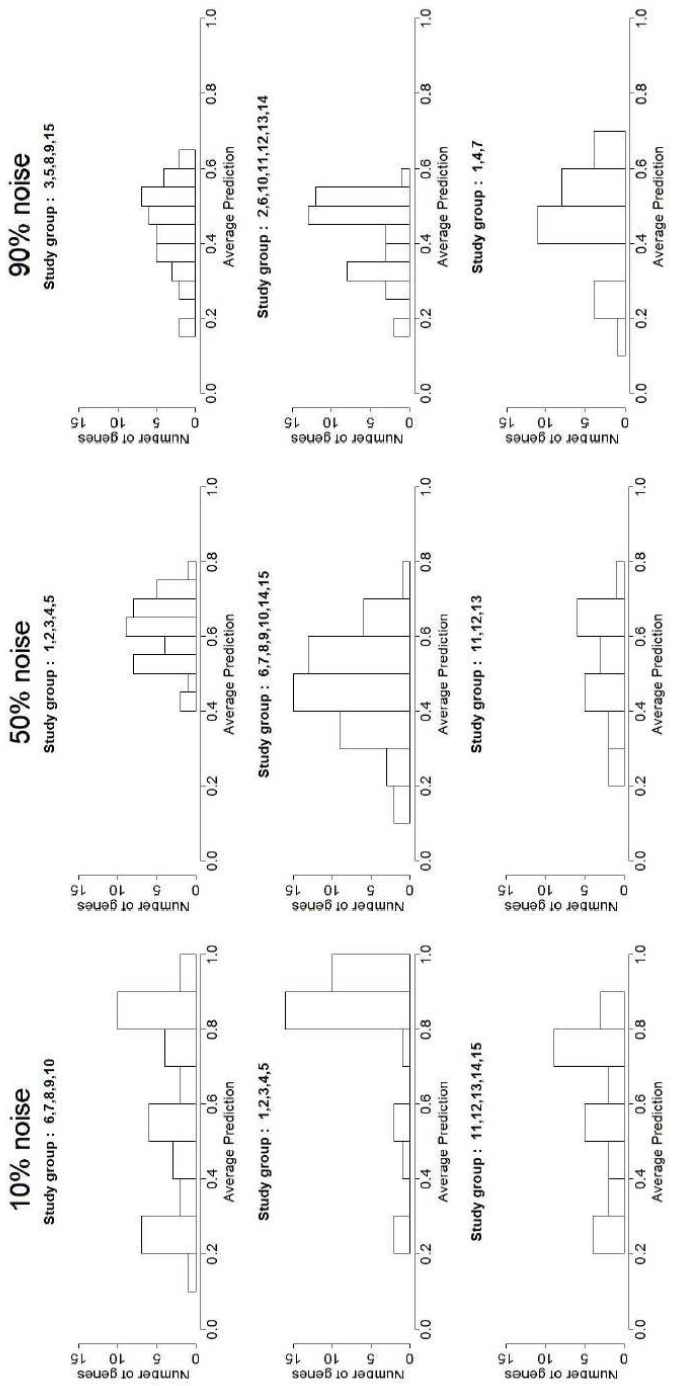



Figure 4.10: Intra cluster correct-prediction distribution for 10, 50 and 90% perturbation. The figures show the histograms of the intra cluster correct-prediction (calculated within the same cluster using cross-validation) for the simulated dataset for different levels of noise.

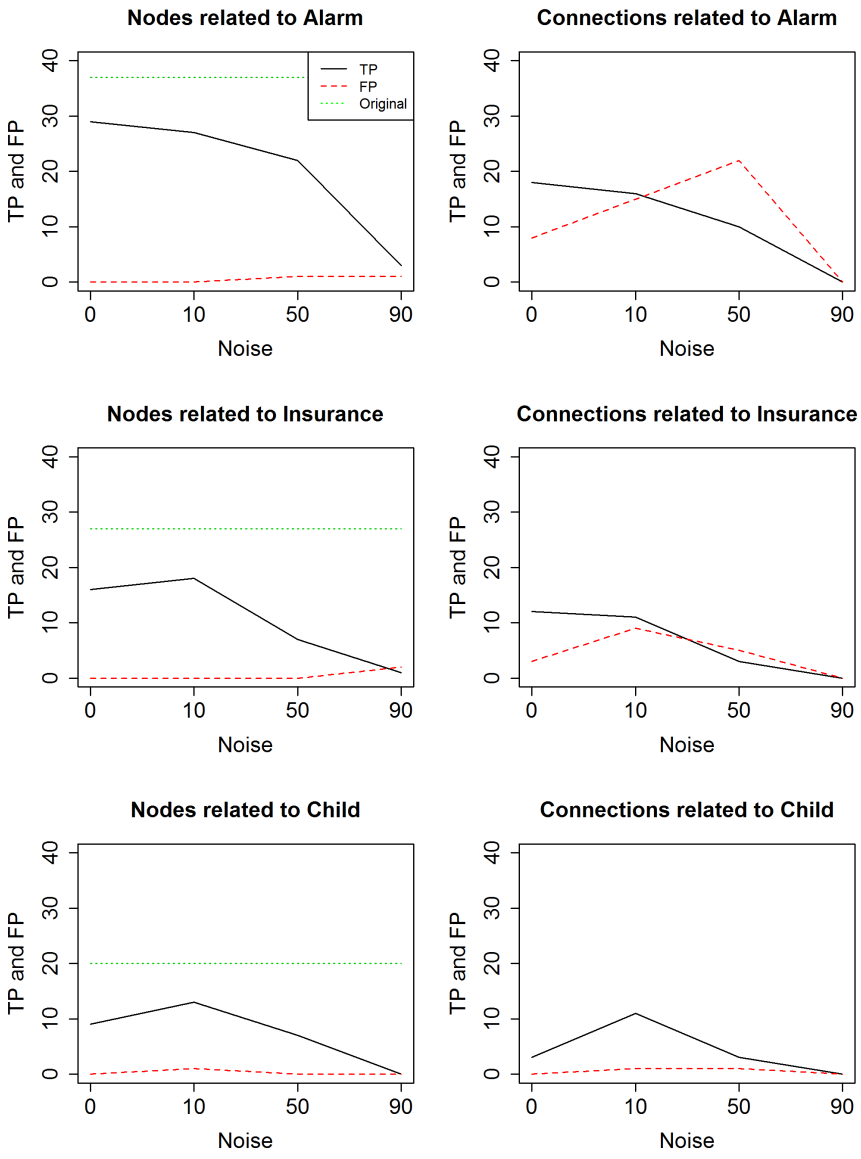



Figure 4.11: TPs and FPs vs noise **after** calculating correct-prediction. The graphs show the number of TPs and FPs nodes and connections detected at different levels of noise. Threshold set to 0.6. The dotted lines at the top of the graphs indicates the number of nodes in the relative original network.

## 4.5 Comparison with Biclustering

We now compare our pipeline with a biclustering method called *Questmotif* which is based on the framework described in Murali & Kasif (2003). Biclustering identifies both genes and samples simultaneously so whilst subnetworks are not discovered (which our approach focuses on), it should at least identify variables that are clustered for specific studies. We apply biclustering to the same *big matrix* dataset of 600 samples and 84 variables, and exploit the results. *Questmotif* detects 9 biclusters. Cluster 1 groups 124 samples out of which 122 belongs to network alarm, and 8 variables all involved in the alarm mechanism. Cluster two groups 261 samples of which 190 belongs to network insurance and only two genes both belonging to the insurance network. Cluster 3 groups 93 samples, 88 of which belong to the child network along with 4 variables from the child network. Bicluster 4 groups 20 samples and 10 variables from the alarm network. Bicluster 5 still groups a majority of samples belonging to alarm. The remaining clusters groups have mixed samples and mixed variables in a very low number. The results are shown in Figures 4.12a and 4.12b

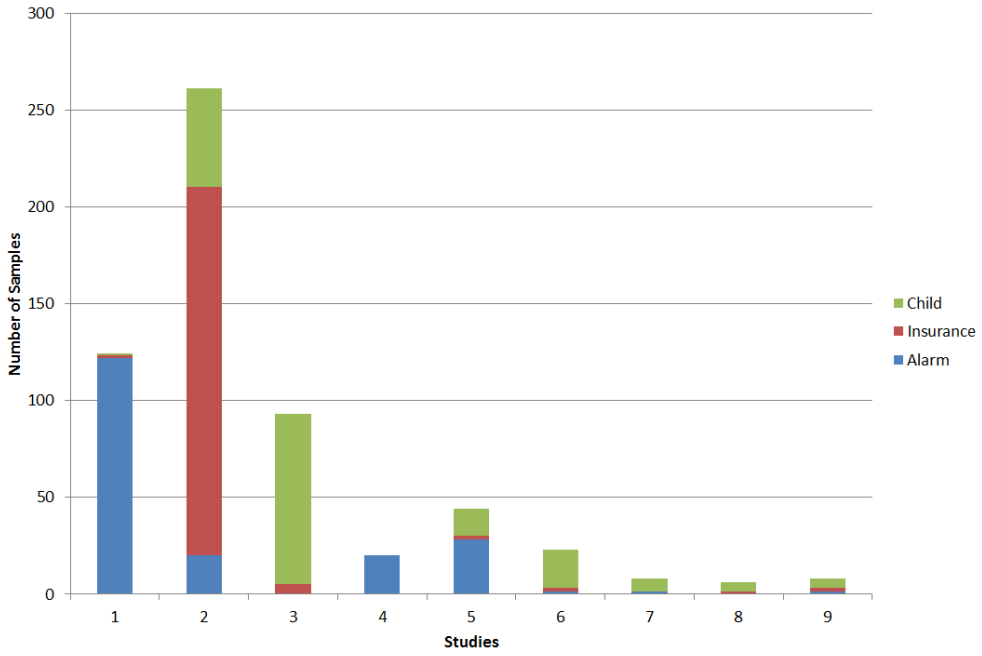
Overall, bicluster does not perform as well as our pipeline. It manages to identify a respectable number of correct samples, but fails at detecting as many corresponding true variables as our pipeline (and no connections are discovered as it is not a network-based approach).

## 4.6 Discussion

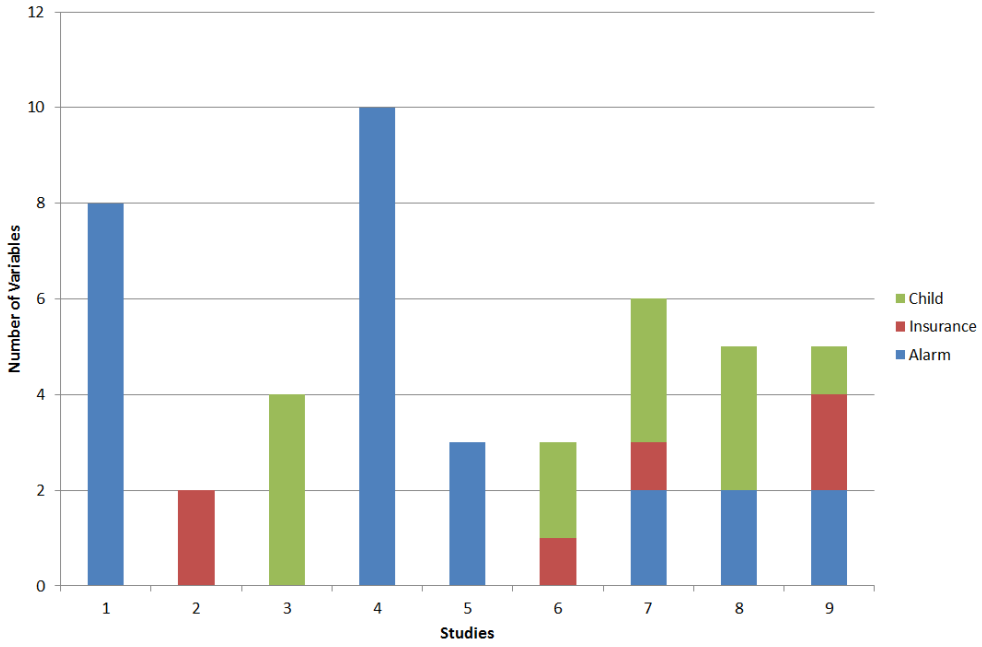
In this chapter we have explained our aims with its challenges and proposed a combination of steps to overcome them and achieve our goal. The pipeline developed is called UNIP (Unique Network Identification Pipeline) and consists of a list of steps to deal with certain characteristics of microarray data. To verify that UNIP robustly and reliably generates unique networks we test it on multiple independent synthetic datasets downloaded from a publicly available repository database.

We selected three networks with comparable numbers of nodes in a way that when the datasets are integrated the total number of variables stays below 100. This allows Bayesian networks to work with a computationally reasonable input dataset.

To simulate different conditions and the noise typical of real microarrays, we merge the data together adding random values. We also perturb the original data to simulate increasing level of noise from no-noise (0%) to 10% until 90%. For each level of noise a GRN for each study is built using Bayesian networks. Given the graphical structure obtained, the similarity measure is calculated and the studies are grouped in study-clusters. Finally, for each study-cluster a




(a)




(b)

Figure 4.12: The figures show the group of **samples** and **variables** respectively obtained using the bicluster method QuestMotif (Murali & Kasif 2003). Each bar represents a sample-group indicated with a number on the x-axis. The different colours indicate to which original network the samples in the sample-group truly belong to. The y-axis indicates the number of samples in Figure a and the number of variables in Figure b.

consensus network first and a unique-network afterwards is built and the prediction-accuracy intra and inter clusters is measured.

The simulated data study indicates that our pipeline works almost perfectly when the input data presents no-noise (0%). The same behaviour is followed when the noise level only slightly increases to 10%. Furthermore, it proved to be reasonably resilient to noise until 50% of the data is affected. While as expected much of the power is lost when the data is 90% or more random and therefore contains little information.
Both the network clustering process and the detection of variables that truly belong to the original networks seem robust and only fail at higher level of noise.
In conclusion we can state that our pipeline appears robust and reliable enough to explore real microarray data.

In the following chapter we will use our method with two sets of real microarray data studies: Wheat and *Fusarium*. Unlike the case of synthetic datasets, real data requires a pre-processing step which may affect the following results. In addition to the prediction-accuracy two different tools Mapman (Thimm et al. 2004) and AIC-MICA (Lysenko et al. 2011) are used as support to the biological validation. We will show that wheat datasets behave similarly to the case of zero or very low noise, while *Fusarium* appears to be associated with noisier data as a result of more clearly defined conditions for wheat.

# Chapter 5

# Analysis of Real Data

## 5.1 Introduction

In the previous chapter we developed a pipeline called UNIP to semi automatically identify subnetworks that are specific to a set of conditions. The pipeline takes as input a set of raw independent microarray datasets (studies) obtained using the same platform to avoid bias and extra pre-processing. The data is downloaded from public databases such as Array Express (Rustici et al. 2013, Parkinson et al. 2007) and NCBI GEO (Edgar et al. 2002). For each study it builds a GRN and uses a network similarity measure to group the studies into study-clusters using clustering which aims to cluster studies which belong to similar generic conditions. For example 'salt stress' and 'drought stress' both belong to the generic category of 'stress-enriched' and are therefore clustered together. After a consensus network for each study-cluster is calculated the unique-networks (study-specific subnetworks) are derived. Finally intra and inter clusters prediction accuracy are calculated to refine the results.

The first step developing this pipeline was to test it using synthetic data with characteristics that are already well known in order to evaluate the results. The findings proved the pipeline able to reliably identify sub-networks specific to a set of studies and to be robust for quite high levels of noise. Microarray data generated from organisms subjected to different conditions (even under well standardised experimental conditions) involves a lot of bias and noise.

We now apply UNIP to real datasets, explore the findings and statistically evaluate the results. When analysed we have to keep into consideration experimental variation, bias and both human and machine errors without forgetting issues with the structure of microarray, involving thousands of genes but only few tens of samples. In this Chapter we focus our attention on two different organisms: wheat and on Fusarium.

The following Chapter is organized as following. Section 5.2 describes the adaptations made to the UNIP pipeline to work with real datasets. Section 5.3 explains the real dataset structure and the results obtained from our pipeline. Section 5.4 compares the findings with other popular techniques. Section 5.5 shows what we found in the literature in support of our findings. Section 5.6 explores the results on Fusarium. Finally, section 5.7 discusses the findings.

## 5.2 Pipeline adaptation to real datasets

As previously explained, the number of variables in microarray data and connections between them can be reduced by several order of magnitude without any loss of information. Therefore, a preprocessing step is necessary, first, to make UNIP applicable to real data.

### 5.2.1 Variables selection

The first real data set we focus on is wheat. Wheat is an hexaploid organism and consequently presents a highly developed genome with 61290 genes (about three times the human genome) and therefore requires the identification of informative genes. To prevent noise and bias increasing we choose not to cluster but to discard all non informative genes. Knowledge of wheat is still young, so we decide to focus on those genes that researchers have already explored and assigned a (preliminary) function to.

The Gene Ontology database (Ashburner et al. 2000), as the name says is an up-to-date tree-structured ontology database which describes gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner. We first discard all the genes that are not yet present in GO to focus on genes that we can validate biologically. This filtering step reduce the variables number to about a third (21487).

All the studies downloaded for wheat have the characteristic of including one or few control samples. If a gene is particularly informative we expected it to behave very differently in the treated samples compared to the controls. Consequently its variance within the study under consideration will be high. Therefore we select the most informative genes selecting those with a variance higher than a threshold $thr$. As is commonly performed in gene expression analysis (see Section 2.4) we preserve only those genes that passed the $thr$ threshold in at least $s$ studies. Both $thr$ and $s$ are set by the user depending on the computational needs.

### 5.2.2 Consensus and unique networks

Once the most informative variables/genes have been selected, we simply apply each step of the pipeline UNIP described in Chapter 4.2 an overview of which is showed in Figure 4.1. Given $n$ independent studies we firstly need to derive $n$ GRNs, one for each study. Because of the continuous nature of microarray data and the higher number of genes (compared to synthetic data) we opt for glasso rather than BN learning. Glasso (Friedman et al. 2008, 2014) is a highly scalable approach which calculate the inverse covariance matrix using the lasso penalty (Tibshirani 1996) to define the dependencies between the nodes. To make the network as sparse as possible glasso uses the regularization parameter $\rho$. If $\rho = 0$ there is no penalization and consequently the matrix is not made more sparse. The higher $\rho$ is set to be the more sparse the matrix become meaning that the weaker connections are discarded and only the strongest and more reliable are revealed.

Given the $n$ glasso derived networks we calculate the sensitivity measure to quantify the graph similarity across all the studies and use it to cluster them using k-means, as it proved to be the most accurate. For each study-cluster we derive the consensus network which consists of the connections in common among all the studies (or a percentage of them) within the study-cluster under consideration.
Next, we identify the connections in each consensus-network that are present in the network under consideration but not in the others (unique-connections) and consider the genes that the unique-connections link together. Now from the original microarray dataset and for each study-cluster, we select the sub-dataset containing the genes involved in the unique-networks and the samples of the studies in the study-cluster under consideration. We discretize the value of the sub-dataset and build the *unique-networks* applying the Bayesian networks (Heckerman et al. 1995, Friedman et al. 2000) through the bnlearn package (Scutari 2009) using hill climbing coupled with the Bayesian Information Criterion (BIC) (Schwarz et al. 1978).

Finally to statistically evaluate our findings we calculate inter and intra cluster prediction accuracy. Across all study-clusters we derive a training and a test set using the leave-one-out cross validation (refer to Section 3.6.4). If the gene value is predicted correctly a 1 is assigned, zero otherwise. The average prediction is calculated across all genes and for all training and test combinations, to obtain the prediction accuracy.

### 5.2.3 Biological support

Having identified the study-clusters and, in turn, the study-specific mechanisms within the *unique-networks*, we explore the biological meaning behind them. To do this, we exploit two

pieces of software:

1. *Mapman* (Thimm et al. 2004) which explores gene-by-gene the functions related to it and returns a list of functions and a graph of connections

2. The AIC-MICA method (Lysenko et al. 2011). The method identifies functions in the biological process aspects of the Gene Ontology that best characterise particular groups of genes. It uses both the structure of the ontology and a term specificity measure (information content, IC) to find terms that are both biologically specific (e.g. not too high-level) and applicable to the largest possible subset of each group. Therefore, unlike the over-representation measures, it gives a general idea about the role of the cluster as a whole and a level of ontology at which such commonality could be found (e.g. average IC of the found terms).

The combination of these tools allows us to identify gene functions that are characteristic of the study-cluster in consideration, adding credence to our findings.
Finally, in the case of the wheat dataset, to prove that the results are robust and consistent, we conduct a search in the literature for every gene involved in the *unique-networks* and its connections. The results of this research are explained in Section 5.5.

## 5.3 Wheat results

We now focus on the analysis of various wheat transcriptome datasets derived from multiple experiments of plants subjected to a range of treatments: stress, development, etc. Unprocessed wheat microarray expression data for this work was downloaded from ArrayExpress database (Parkinson et al. 2007). Only studies using A-AFFY-57 GeneChip Affymetrix Wheat Genome Array technology which profiled wheat species were included. The combined dataset was pre-processed using Robust Multichip Average method (Irizarry et al. 2003) and redundancy-adjusted Pearson correlation coefficient was calculated according to the method described in Obayashi et al. (2011).
The combined microarray dataset contains 61290 and 16 independent studies for a total of 523 samples. Each study represents a different treatment the plant has been subjected to, as shown in Table 5.1. Studies 1-6, 12, and 13 are considered stress-enriched, and the remaining as non-stressed treatments based on the labels taken from Array Express (Parkinson et al. 2007, Rustici et al. 2013).

Wheat Studies

| Study | Label | Number samples | Description |
|---|---|---|---|
| 1 | E-MEXP-971 | 60 | Salt stress |
| 2 | E-MEXP-1415 | 36 | S and N deficient conditions |
| 3 | E-MEXP-1193 | 32 | Heat and Drought Stress |
| 4 | E-MEXP-1694 | 6 | Re-supply of sulfate |
| 5 | E-MEXP-1523 | 30 | Heat stress |
| 6 | E-MEXP-1669 | 72 | Different nitrogen fertiliser levels |
| 7 | E-GEOD-4929 | 4 | Study parental genotypes 2 |
| 8 | E-GEOD-4935 | 78 | Study 39 genotypes 2 |
| 9 | E-GEOD-6027 | 21 | Meiosis and microsporogenesis in hexaploid bread wheat |
| 10 | E-GEOD-9767 | 16 | Genotypic differences in water soluble carbohydrate metabolism |
| 11 | E-GEOD-12508 | 39 | Wheat development |
| 12 | E-GEOD-12936 | 12 | Effect of silicon |
| 13 | E-GEOD-11774 | 42 | Cold treatment |
| 14 | E-GEOD-5937 | 4 | Parental genotypes 2 biological replicates from SB location |
| 15 | E-GEOD-5939 | 72 | 36 genotypes 2 biological replicates from SB location |
| 16 | E-GEOD-5942 | 76 | Parental and progenies from SB location |

Table 5.1: Study numbers, labels, number of samples and descriptions of the wheat microarray dataset.

Each study contains a variable number of samples, the majority derived from the treatment and only few as controls. Because of this after filtering the genes existing in the GO (Ashburner et al. 2000) database (reduced to 21487 genes) we further reduce the number of variables calculating the standard deviation across each study and select only the genes presenting a sd higher than a user-defined threshold set to 2. This leaves each study with a different set and number of genes. To standardise and allow a comparison we globally select the genes that exceed the sd threshold in at least 4 (25%) of the 16 studies to obtain a final number of informative genes equal to 67. Again this percentage can be customized based on the user needs.
Once the relevant genes are selected, following the step of UNIP, we apply *glasso* to build a net-

work for each study and then, calculate the sensitivity measure in order to cluster the studies based on graphical similarities. As for the simulated data, we explored k-means (see Section 3.2 ) which generated the most convincing study-clusters. We evaluated different values of k but found that 3 clusters were the most revealing. Table 5.1 demonstrates that the studies can be grouped in two generic conditions: stress-enriched and non-stress. The clusters resulting from k-means are: $\{2, 5, 6, 10, 12\}$, $\{1, 3, 4, 9, 11, 13\}$ and $\{7, 8, 14, 15, 16\}$ based upon the studies numbering from Table 5.1. While the third cluster clearly groups together all the non-stress studies, the other two reflect studies that are stress enriched. In the Figures 5.1, 5.2 and 5.3 we show the unique-networks, learnt with *bnlearn*, for wheat in the two study-clusters of stress-enriched conditions (Figures 5.1 and 5.2) and the unique-network for the non-stress conditions cluster (Figure 5.3). Once the unique-networks for each cluster are derived, we calculate the prediction-accuracy using the leave one out cross validation (LOOCV) technique as explained in details in Section 4.2.6. For each combination of training and test set we build the corresponding unique-network and use this structure combined with the data to derive the conditional probability tables associated with it and consequently calculate the prediction accuracy for each gene involved. A clear and visible consequence of this is the existence, in the unique-networks derived from each cluster, of highly predicted isolated nodes (prediction-accuracy $\geq 0.6$). Although these nodes are isolated, in the networks in these figures, they clearly have one or multiple parents in at least one of the unique-networks derived during the LOOCV.

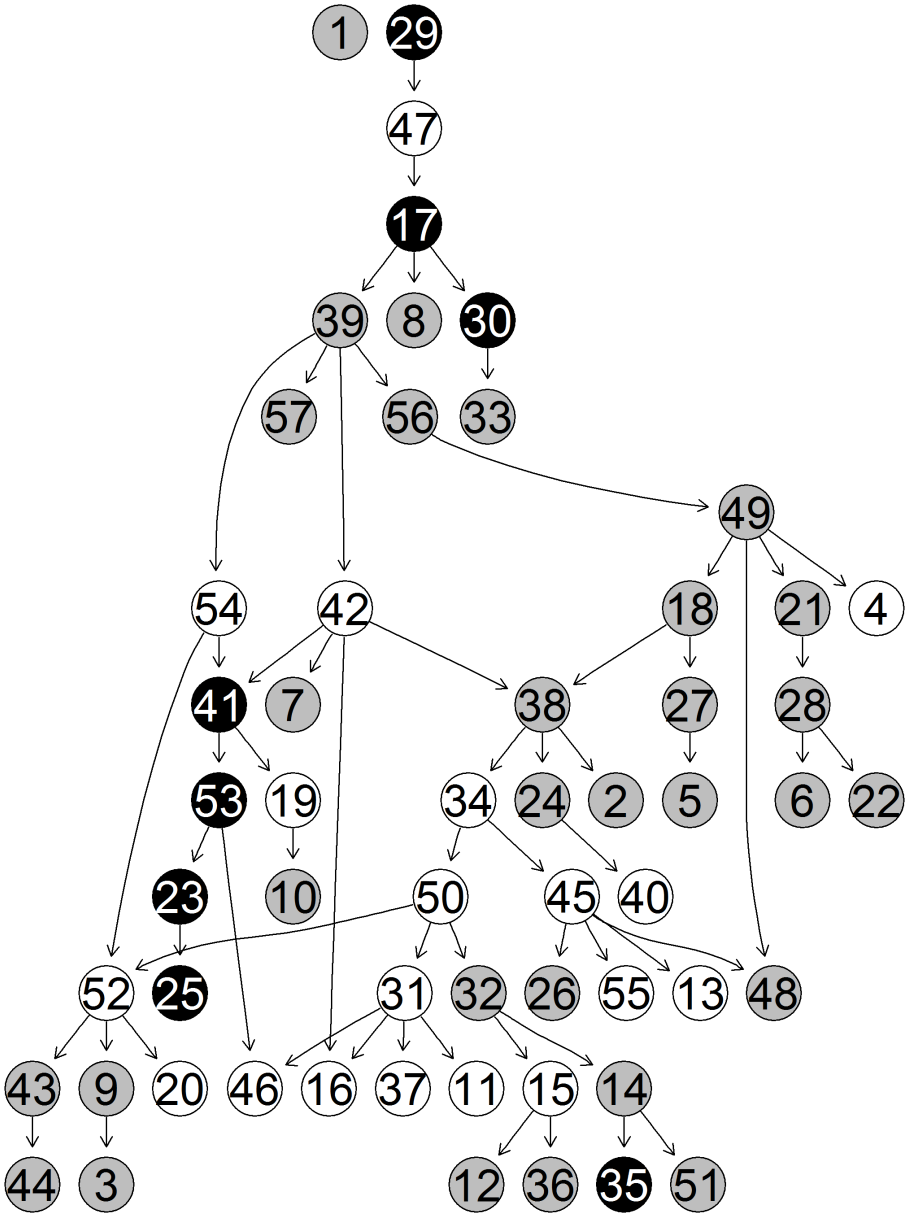



Figure 5.1: Network 1. Unique-Network for wheat under stress-enriched conditions in cluster 1. The highlighted genes (black and grey) have a *intra* prediction accuracy higher than 0.6, meaning that they have been predicted correctly at least 60% of the times by the remaining genes inside the same study-cluster. The network shows one big path starting with a highly predicted stress related gene (black genes number 29) connected through one gene (47) to a two more stress related genes all directly connected to highly predictive genes and another stress-related path involving the genes 41-53-23-25.

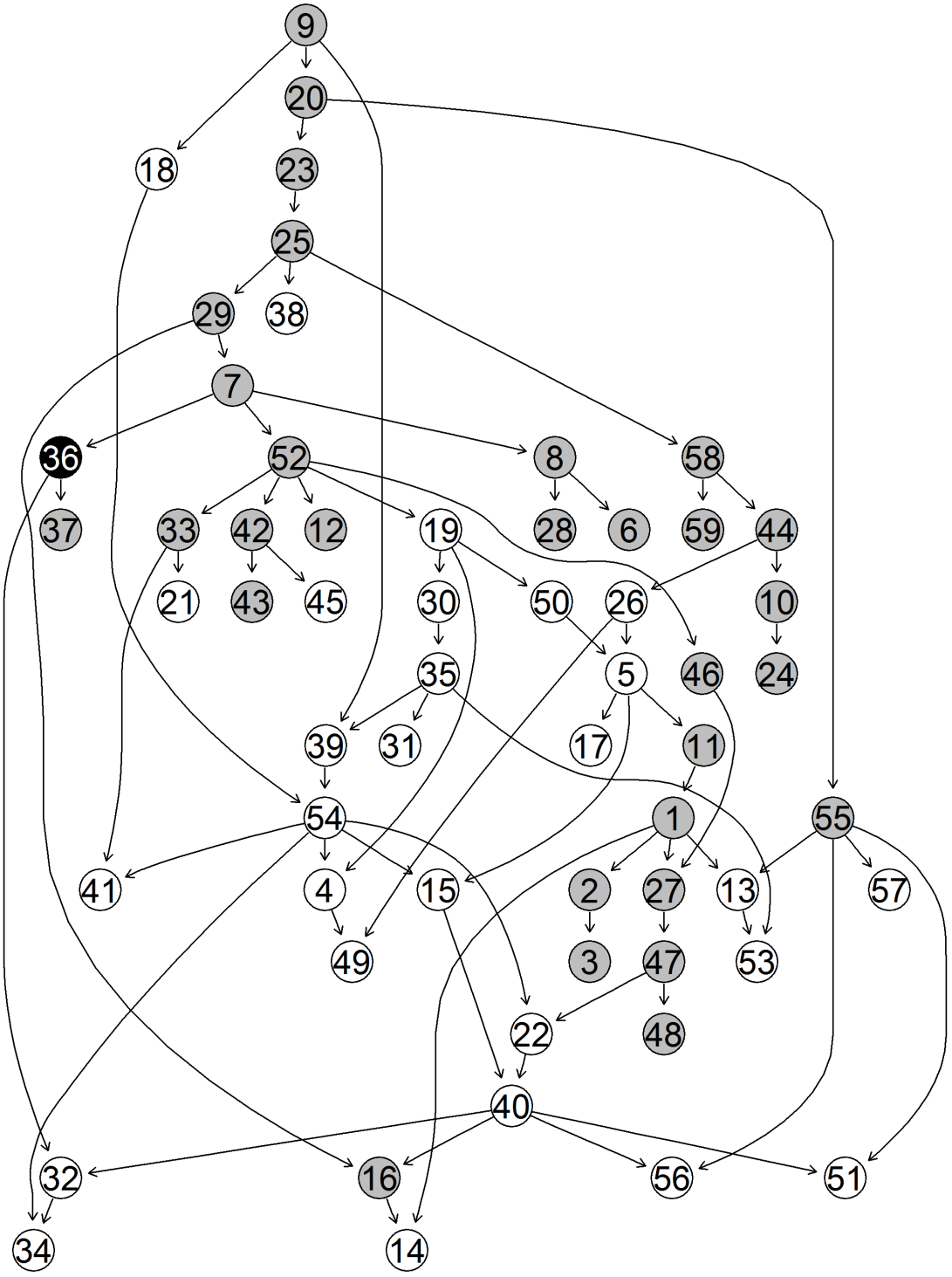



Figure 5.2: Network 2. Unique-Network for wheat under stress-enriched conditions in cluster 2. Grey nodes indicate highly predictive (average correct-prediction level higher or equal to 0.6) genes. Black nodes highlight highly predictive and stress related genes. This network presents a high number of highly predicted genes, but only one that is highly predicted and stress-related.

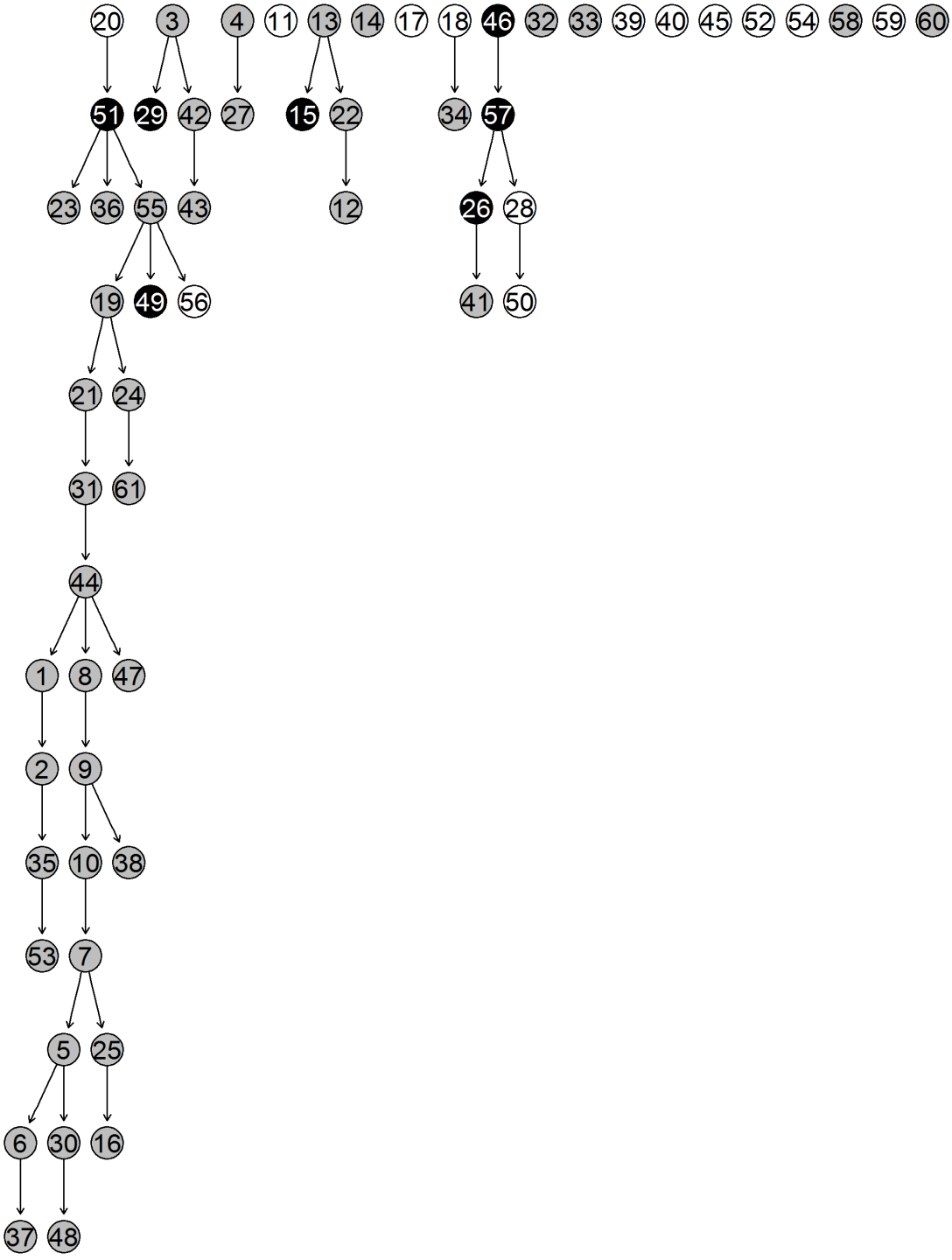



Figure 5.3: Network 3. Unique-Network for wheat under non-stress conditions in cluster 3. This network is composed of multiple smaller sub-networks not immediately related to each other. Few genes of those involved are still stress-related (black nodes) and almost the total of the them present high prediction (grey nodes).

For visualization purposes the numbers identify the genes (the corresponding gene number and boxplot of each gene's internal prediction are found on appendix table) and the black circles represent in both the highly predictive genes that are involved in biotic (caused by living organisms) and abiotic (caused by non-alive factors) stress response. In both networks we clearly see specific paths and groups of genes that are highly connected. Using Mapman (Thimm et al. 2004) we were able to associate a function to each gene.

Focusing on the stress-enriched conditions network, the procedure has managed to identify a relatively small number (58) of well-connected nodes which form a distinctive path. Isolated points are not shown because uninformative. We see that genes involved in both kinds of stress response (biotic and abiotic stress) are involved in the network. Specifically the first four genes that start the network pathway in Figure 5.1 (29 - 47 - 17 - 30) are all involved in biotic stress. The remaining highlighted genes instead are mostly involved in heat stress. A good number of photosynthesis related genes are also involved, in particular (18 - 27 - 21 - 28 - 6 - 22). On the non-stress related network in Figure 5.3, we have again identified a reasonable number of genes though these are less connected. However, one very well defined pathway exists that consists mainly of photosynthesis-related genes (not highlighted).

In the same network in Figure 5.3, less genes are found that are related to stress response and those that do appear are much less connected, except for the path formed by (46 - 57 - 26 - 50) nodes. The software described in Lysenko et al. (2011) returns the following (see Table 5.2) highlighted biological functions which go to reinforce the results from Mapman. Higher values of Information Content (IC) are associated with more informative terms. Values greater than 3 are generally considered to be biologically informative.

In the Figure 5.4 we show the intra predictive accuracy boxplot for each study-cluster and a line which indicate the average inter clusters prediction-accuracy. What we expect is a better correct-prediction within the study-clusters and a weaker one outside the clusters. Each boxplot represents the percentage of how many times the gene has been predicted correctly among all the different given samples.

The chance of correctly predicting the genes randomly is one in three (there are three possible states for each gene: *under-regulated*, *normal*, *over-regulated* ). Values above this can be considered better than random. In the figures we clearly see that the *intra cluster predictions* (calculated by cross validating within a study-cluster) are quite high for most of the genes with little variations. For the *inter cluster predictions* (predictions on data outside of the study-cluster), however, the mean correct-prediction values are mostly not better than chance as one would expect, and the standard deviations are very high making them not reliable. In the majority of the cases, in fact, when a gene has an extremely high intra cluster correct-prediction it

also shows a very low or a wide standard deviation in the inter clusters correct-prediction graph. This implies that the identified subnetworks are indeed specific to their study cluster, making them easier to characterise.

Figure 5.5 summarizes the results shown in Figure 5.4. It shows a comparison of the average mean and average variance across all genes between intra-cluster and inter-clusters prediction-accuracy. As expected it shows a very clear distinction in Network 1 and Network 3 study-clusters with intra-cluster prediction always higher than the inter-clusters. The poor performance of Network 2 is easily explained by the fact that study-cluster 2 is not a good cluster containing half stress-enriched studies and half non-stress.

| U-N | GO Id | GO Name | IC |
|---|---|---|---|
| 1 | GO:0019538 | protein metabolic process | 3.19 |
| 1 | GO:0006950 | response to stress | 3.96 |
| 1 | GO:0071840 | cellular component organization or biogenesis | 3.98 |
| 2 | GO:0006950 | response to stress | 3.96 |
| 2 | GO:0071840 | cellular component organization or biogenesis | 3.98 |
| 2 | GO:0019684 | photosynthesis, light reaction | 8.32 |
| 2 | GO:0044267 | cellular protein metabolic process | 3.45 |
| 3 | GO:0006950 | response to stress | 3.96 |
| 3 | GO:0015979 | photosynthesis | 7.13 |
| 3 | GO:0071840 | cellular component organization or biogenesis | 3.98 |
| 3 | GO:0009628 | response to abiotic stimulus | 4.97 |
| 3 | GO:0042221 | response to chemical stimulus | 4.12 |
| 3 | GO:0006091 | generation of precursor metabolites and energy | 5.14 |
| 3 | GO:0044267 | cellular protein metabolic process | 3.45 |

Table 5.2: Wheat Unique-Networks(U-N) biological process functions from Gene Ontology as described in Lysenko et al. (2011). IC values greater than 3 are considered to be biologically informative.

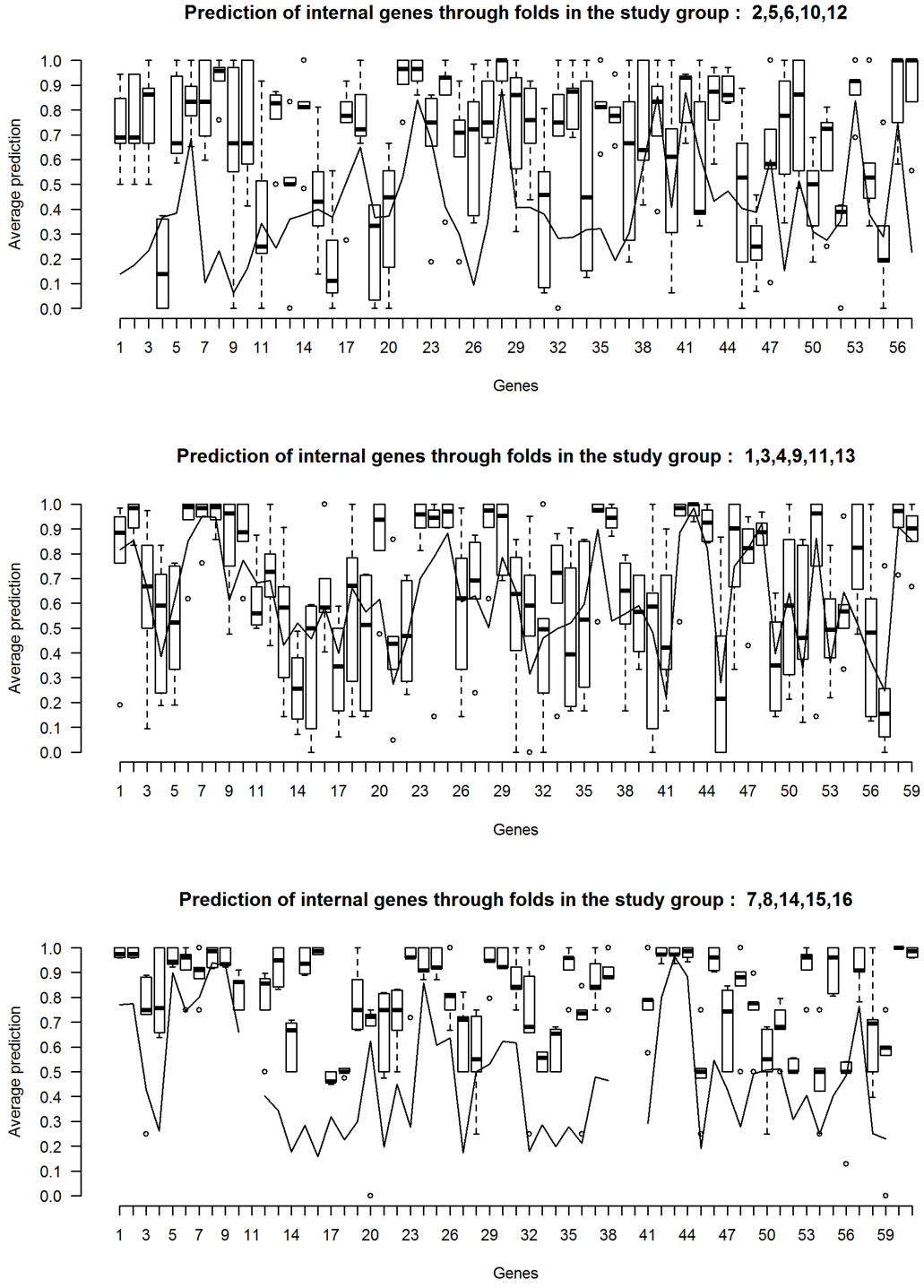



Figure 5.4: Boxplot intra clusters prediction. The boxplots in each figure represents the *intra* (internal) cluster prediction-accuracy for each gene where the line indicates the average *inter*-clusters (external) prediction-accuracy.

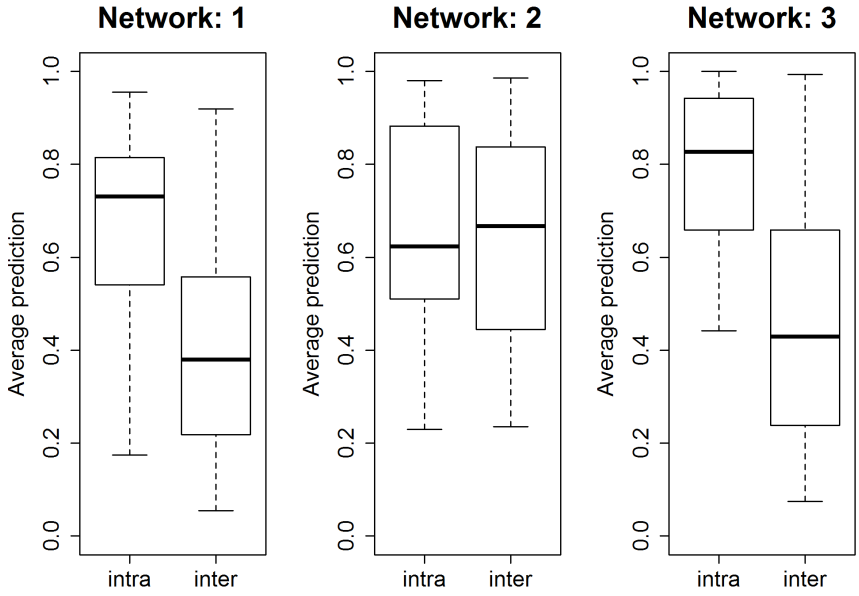



Figure 5.5: Boxplot intra vs inter clusters correct-prediction.

## 5.4 Wheat comparison

### 5.4.1 Comparison with Bicluster

Finally, we compare the results obtained with our algorithm in wheat with the one obtained using the Spectral Bicluster algorithm (Kluger et al. 2003). The method, after appropriately tuning the parameters, identifies 17 biclusters. On the wheat data each resulting bicluster highlights a different set of samples but the same set of six genes, 5 of which are related to abiotic heat stress. The genes highlighted by biclustering are also in the list of genes detected by the algorithm described in this paper, specifically we can see five of these genes also highlighted in Figure 5.1 (23 -25 - 41 - 46 - 53). This discovery points out the importance of these 5 stress-related and 1 protein-degradation-related genes but unfortunately biclustering fails at identifying other equally important stress-related genes identified by our algorithm. In addition the six genes that are identified do not seem to be associated with a specific subset of samples. Rather each of them have been detected in all of the biclusters. Regarding the samples, about half of the biclusters manage to group together samples of stress-enriched studies but split samples from the same study. Unfortunately, none of the biclusters group the non-stress studies accurately enough to identify specific non-stress clusters. Furthermore, considering that each

study consists of both actual treatment samples and a small number of controls it might be that biclustering merges together the control samples of the stress-conditions with non-stress samples but this union occurs too often and with too many samples for this to be considered the case. In conclusion, we have found that the resulting biclusters do not properly cluster the samples together, even ones belonging to the same study. Every bicluster highlights the same group of genes preventing any discovery of differences between treatments. It still discovers some important genes but much less than the ones we are able to find with the method proposed in this thesis.

### 5.4.2 Comparison with WGCNA

As previously pointed out the *glasso* technique goes beyond simple pairwise relationships estimating a sparse inverse covariance matrix using the lasso ($L_1$) penalty. We compare it with the WGCNA (Weighted Gene Co-expression Network Analysis) technique as explained in section 3.4 of Chapter 3. We applied both the scale free criterion for each study obtaining an array of different values of beta and then with only one value of beta set to 6 which is suggested to be the most appropriate value (Horvath 2005). In both cases the results are extremely similar. Of the three clusters obtained with k-means only one of the stress clusters is quite reliable while the other two are quite mixed or meaningless (only two elements). Furthermore the unique networks reveals very small size graphs with much less nodes (less than 10) involved and very few connections. The small number of nodes detected in WGCNA have also been previously detected in *glasso*. As expected, the intra cluster correct-prediction is extremely good for the genes involved in each study-cluster, but, in this case, the number is so little that these results leave some strong doubts on the WGCNA algorithm usability on this dataset. Next, we show another case study with Fusarium microarray data.

## 5.5 Biological validation - literature

A key focus of this work is the exploration of wheat of which there is still much uncertainty. We now explore in some detail the biological feedback based on the discovered unique networks. The three networks in Figures 5.1, 5.2 and 5.3 are indicative for different sample sets e.g different stress conditions. They represent increase in the gene transcription for certain genes and the links between them. Eighty percent of Networks 1, 2 and 3 are consistent with the literature. The remaining twenty percent did not present direct correlation though there is evidence for some correlation in database sources such as *The Arabidopsis Information Resource* (Lamesch et al. 2012), *NCBI - The National Center for Biotechnology Information* (Edgar et al. 2002)

and *Plant Transcription Factor Database* (Pérez-Rodríguez et al. 2009).
First, the main genes correlated to biotic stress were basic chitinase. Basic chitinases are antimicrobial proteins that are capable of degrading fungal cell wall chitin. They are two classes either basic or acidic isoelectric points (Samac et al. 1990). Gene 19 (PR3 (Basic chitinase)) in network 2 (NW2) in Figure 5.2 (30 in NW1, Figure 5.1; 15 in NW3, Figure 5.3) is correlated to gene 30 (allergen V5/Tpx-1-related family protein) in NW2, followed by 35 (BMY1, (BETA-AMYLASE)) in NW2 and 31 (PR3, (Basic chitinase)) in NW2. Basic chitinase (19 in NW2) also affects 49 (CK215257 Dirigent-like superfamily) via gene 4 (cysteine proteinase, putative). Allergen V5, pathogenesis related 4 and basic chitinase (29, 47, 17 in NW1; 30, 19 and 50 in NW2, respectively) are represented in both networks with different links between the gene expressions. Differently in network three (NW3) begins with gene 20 (PR3, (Basic chitinase)) followed by 51 (HEL, PR-4, (Pathogenesis-related 4)) and 36 (BMY1, (Beta-amylase)), where allergen V5 is completely missing. Therefore we conclude that gene expression of allergen V5 may be only visible under certain stress conditions.

Glycine decarboxylase complex H (gene 39, NW1) was correlated to transcription of Rubisco gene (56, NW1) that regulated genome uncouples 5 (GUN5). GUN5 is a plastid derived signal that plays an important role in the coordinated expression of both nuclear and chloroplast localised genes that encode photosynthetic-related proteins (Mochizuki et al. 2001). It regulated genes 21 (LHCA1), 28 (PSAK (Photosystem subunit K)), 6 (LHCB5 (Light harvesting complex of photosystem II 5), 22 (PSAD-1 (photosystem I subunit D-1)) and 4 (cysteine proteinase, putative) and gene 18 (LHCB1.5, Photosystem II light harvesting complex gene 1.5). Followed by gene 27 (LHCB3*1, Light-harvesting chlorophyll binding protein 3) and 5 (RNS1 (Ribonuclease 1); endoribonuclease) confirming its functional properties. In NW2 the relationship between Rubisco (gene 58, NW2) and glycine decarboxylase complex H (44, NW2) seems to be in the opposite direction. The previously published data suggest that the expression of both genes is light dependent and tissue specific, which is due to 259-bp upstream region of the promoter region (Srinivasan & Oliver 1995). In both NWs ferredoxin gene (59, NW2) and (57, NW1)) was linked to Rubisco and glycine decarboxilase complex. Due to physiological importance of these genes in both networks the two relationships could be correct. In NW3 the photosynthetic reaction is regulated by MYB  like transcription factor (19, NW3) and glycine decarboxylase complex (44, NW3) while the transcription of Rubisco gene is below the level of significance (Kwon et al. 2013).

Photosystem I was represented by genes 22 and 28 in NW1; 24, 29 and 22 in NW2; and 24, 31 34 in NW3. The photosystem I composed of four complex (Lhc (light harvest complex) proteins and a1-Lhca4 belonging to the light harvesting protein family (Wientjes & Croce 2011).

Also the light harvesting complex II (LHCII) is implicated by the regulation of excitation energy distribution between Photosystem I (PSI) (21, NW 1) and Photosystem II (PSII) (6, NW 1) during the state transition and also light-harvesting complex II binds to several small subunits of photosystem I (Zhang & Scheller 2004). PSI-K subunit of photosystem I (28, NW1; 29, NW2 and 31, NW3), is involved in the interaction between light harvesting complex I and the photosystem reaction centre core (Ihalainen et al. 2002, Jensen et al. 2000).

The main trimeric light-harvesting complex of higher plants (LHCII) consists of three different Lhcb proteins (Lhcb 1-3) in Arabidopsis thaliana. In NW1 these genes are 27 (LHCB3*1, (Light-harvesting chlorophyll binding protein 3) and gene 18 (LHCB1.5, (Photosystem II light harvesting complex gene 1.5)) (Damkjær et al. 2009). Gene 6 or LHCB5, (Light harvesting complex of photosystem II 5), this gene is significant because is affected by different light regimes in rye plants. It may be also indicative for wheat function due to the high similarity in the gene sequences between wheat and rye. In NW2, the genes 7, 8 were the same as in the NW1. Also gene 33 (PSAN (photosystem I reaction centre subunit PSI-N); calmodulin binding), 42 (APX4 (Ascorbate peroxidase 4); peroxidase) are related due to their function in photosynthesis (Bang et al. 2008).

Other fundametal parts of the network are the group of heat shock proteins. The major groups are HSP100, HSP90, HSP70 and they are also confirmed in wheat (Grigorova et al. 2011). The novel finding in NW1 is that the genes indicated by 41 (HSP70), 23 (HSP101 (Heat Shock Protein 101)), 53 (HSP70), 25 (HSP21) and 46 (ATHP22.0) are related to a protein degradation gene 54 (CLPP_wheat.gb/CA607537) which is 98% similar to AB042240 Triticum aestivum chloroplast (http://www.ncbi.nlm.nih.gov/nucleotide/13928184). This finding provides new insights into relationships between heat shock proteins and this particular chloroplast gene that seems to have a regulatory function over the sequence in Figure 5.1. In NW2 transcripts for heat shock proteins were not present.

In NW2 the main effects were indicated with the gens MLP-like protein (39, NW2 and 35, NW1), beta amylase (35 in NW2 and 33 in NW1) and rare-cold inducible (RCI) 54, NW2 and 51, NW1). The MLP-like protein is related to beta amylase but there was no explanation exactly how (Ando & Grumet 2010). The link with rare-cold inducible protein and one helix protein seems impossible because rare cold inducible protein is expressed in the roots and is mainly restricted to endodermis (Llorente et al. 2002), one helix protein belong to one of the light-harvesting chlorophyll a/b-binding (Lhc) proteins (Andersson et al. 2003). More research would be required to prove or disprove the relationship between them. Transcript for MLP-like protein in NW3 was not detected to be involved in the network (Figure 5.3; NW3).

ATPRX Q; antioxidant gene (42, NW1 and 46, NW2 and 47, NW3) is central for NW1 and

NW2 but peripheral for NW3. It is highly expressed in leaves and low expressed in the stem. Its expression patterns indicated that is induced by ultraviolet irradiation, low temperature and salt stress. The induction of Prx in response to abiotic stimuli may suggest that Prx may protect the host against environmental stresses (Kim et al. 2010). It looks like gene 42 affects gene 41 (HSP70T-2; ATP binding) and gene 7 (PSBS, (Nonphotochemical quenching), 16 (lipase, putative) and 38 (APX4 (Ascorbate peroxidase 4); peroxidase) and it is itself affected by 39 (GDCH (Glycine decarboxylase complex H)).

The transcript of the chloroplast glyceraldehyde-3-phopshate dehydrogenase (phosphorylating, E.C 1.2.1.14) (GADPH) (38 (GAPA-2—GAPA-2) was only found in NW2. In higher plants exists as heterotetrameter that catalyses the reductive step of the Calvin cycle (Baalmann et al. 1996). GAPA-A subunit was also identified chloroplast localized proteins (Infanger et al. 2011). GAPDH is a classical glycolytic enzyme that is involved in cellular energy production and has suppressed heat shock-induced peroxide production and cell death (Baek et al. 2008). It is also involved in spontaneous assembly of photosynthetic supramolecular complex with CP12 protein that contributes to Calvin cycle regulation and phosphoribulokinase (PRK) in photosynthetic organisms (Marri et al. 2008). It is surprising that the tree proteins GAPDH, CP12 and PRK are not expressed together (Marri et al. 2005). The importance of this gene is its involvement in photosynthesis and Calvin cycle regulation at the same time. Its strategic place in our NW2 points that this gene could be a potential target for further investigation to establish the relationships and regulatory function in both processes.

As described in these biological findings the networks principally highlight stress (as expected), photosynthesis and the Calvin cycle mechanisms. The majority of the links identified by our method have been observed in the literature validating the reliability of our pipeline. The remaining relationships, on the other hand, may be a starting point for further analysis.

## 5.6 Fusarium results

Together with wheat, we also analyse a *Fusarium* graminearum dataset. The microarrays related to this organism (downloaded from Dash et al. (2012)) include 18069 genes and 158 samples gathered in 13 treatments as shown in Table 5.3. We apply the variable selection, as described in Section 5.2.1, and we reduce the number of variables from 18069 to 98. For computational reason we aim to keep the number of genes under 100, though this can be altered at the discretion of the analyst.

Fusarium Studies

| Study | Label | Samples | Description |
|---|---|---|---|
| 1 | FG11-CEL | 9 | Gene Regulation by Fusarium TFs Tri6 and Tri10 |
| 2 | FG13-CEL | 18 | The TF FgStuAp influences spore development, pathogenicity and secondary metabolism |
| 3 | FG14-CEL | 8 | DON induction media |
| 4 | FG2-CEL | 9 | Expression Profiles in Carbon and Nitrogen Starvation Conditions |
| 5 | FG3-CEL | 14 | Cross-species hybridization |
| 6 | FG1-CEL | 18 | Transcript detection on Morex barley spikes |
| 7 | FG12-CEL | 15 | Gene expression during crown rot of wheat |
| 8 | FG6-CEL | 9 | Transcript detection during in vitro sexual development of Fusarium Cch1 calcium channel deletion mutant |
| 9 | FG10-CEL | 6 | Response to trichodiene treatment |
| 10 | FG7-CEL | 12 | Gene expression profiles during conidia germination stages |
| 11 | FG16-CEL | 12 | Fusarium gene expression in wheat stems during infection |
| 12 | FG4-CEL | 5 | Fusarium/Barley RNA dilution |
| 13 | FG5-CEL | 23 | Transcript detection during in vitro sexual development |

Table 5.3: Study numbers, labels, number of samples and descriptions of the *Fusarium* microarray dataset.

Unlike in the wheat dataset, *Fusarium* studies are not easy to group at a first sight. As a result we decided to apply the *glasso* algorithm and calculate the sensitivity measure as it has been done before and then we apply k-means with different values of k and verify if there is any constant pattern. We repeatedly change the value of k in a range from 2 to 10 and we find

that two groups of studies (of 5 and 2 studies respectively) always group together. This allows us to identify two study groups: cluster 1: 8,11 and cluster 2: 2,5,6,7,13. These studies do not belong to any stress condition, but they are recognized to have a similar underlying mechanism through the sensitivity measure.

After the cluster detection we build the Bayesian unique networks for these two groups. Because of their similarity here we show only the unique network for the second group in Figure 5.6. All 98 variables selected appear to be involved in both study-cluster unique networks (except number 45 in the unique for cluster 1). This is because there are no major theoretical differences between the two study-cluster which means that the underlying mechanism might have only slight differences. Again we calculate the prediction accuracy for each gene using the leave one out cross validation as described in Section 4.2.6. The intra cluster prediction shows for both clusters a very good prediction accuracy. For the first cluster, because of its size (only 2 studies) we need to consider only genes with a very high accuracy average and a limited standard deviation range. Only few genes respect these criteria in both clusters. But a very limited number of genes results being very predictive in cluster one and not in cluster two and vice versa. Figure 5.7 compares the average intra-cluster prediction with the average inter-cluster prediction for both cluster. Because only some of the genes are better predicted internally than externally, as expected no difference appears in the average behaviour between internal and external prediction in either groups.

We now apply the AIC-MICA algorithm developed in Lysenko et al. (2011). Since both networks involve the same genes they both have the same main functions. In Table 5.4 we show the main functions. Mapman was not applicable because it does not contain *Fusarium* data.

These results show us that even if the clusters have a similar underlying mechanism we still can identify genes that are highly predictive and therefore characteristic of the clusters. These results can be compared to the one found for the simulated data with a higher level of noise.

### 5.6.1 Comparison with WGCNA:

At this point we explore the WGCNA technique and compare it with *glasso*. As explained in Chapter 3 we first calculate the co-expression similarity matrix and convert it into the adjacency matrix using the scale-free topology criterion. Here again the clusters are organized differently and are not as significant as the ones obtained with *glasso*. The unique networks include far fewer genes and the internal correct-prediction also shows less highly predictive genes compared to the ones we found using *glasso*. Based on the poor results previously obtained from applying biclustering, we decide not to apply this technique on this dataset.

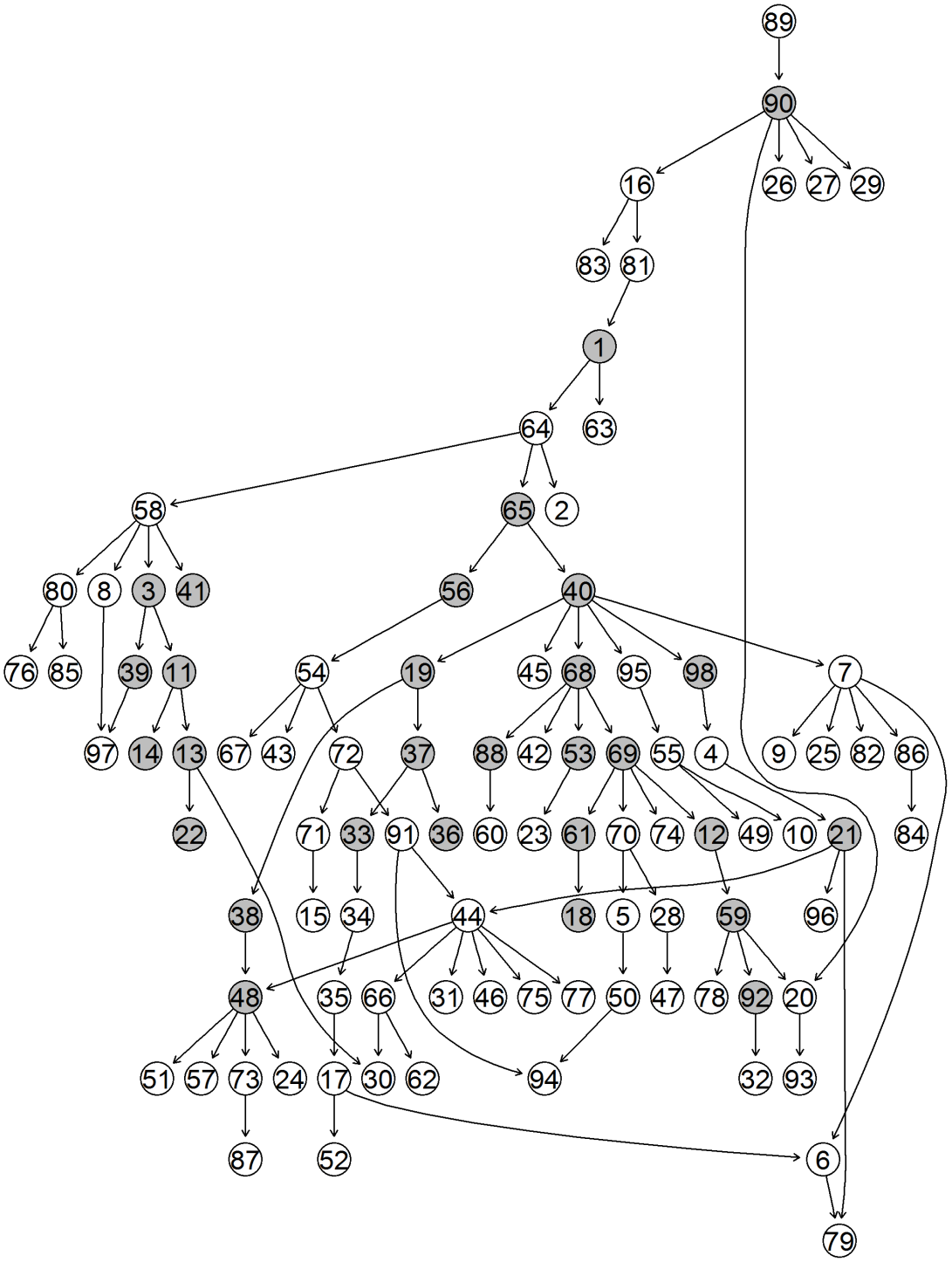



Figure 5.6: Unique-Network for *Fusarium* cluster 2,5,6,7,13. In this figure grey background indicates highly predictive genes (average correct-prediction equal or higher than 0.6). Despite the lack of different conditions in the dataset, as explained in the text, still about a 1/3 of the genes selected are highly predictive.

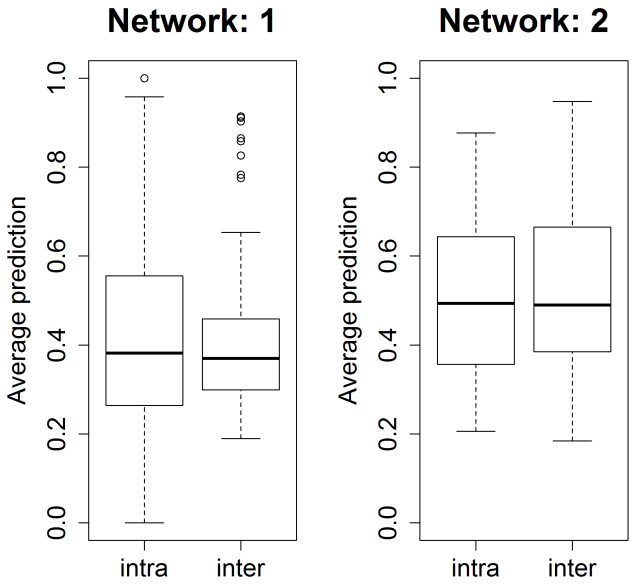



Figure 5.7: Intra vs inter clusters prediction for *Fusarium*.

| GO Id | GO Name | IC Term |
|---|---|---|
| GO:0004175 | endopeptidase activity | 6.93 |
| GO:0015179 | L-amino acid transmembrane transporter activity | 6.90 |
| GO:0004497 | monooxygenase activity | 6.23 |
| GO:0008324 | cation transmembrane transporter activity | 6.16 |
| GO:0005506 | iron ion binding | 6.12 |
| GO:0022804 | active transmembrane transporter activity | 5.29 |
| GO:0022891 | substrate-specific transmembrane transporter activity | 3.78 |
| GO:0046872 | metal ion binding | 3.50 |
| GO:0016491 | oxidoreductase activity | 3.15 |

Table 5.4: *Fusarium* unique networks biological process functions from Gene Ontology as described in Lysenko et al. (2011). IC values greater than 3 are considered to be biologically informative.

## 5.7 Discussion

In this chapter we apply the UNIP pipeline thoroughly described step by step in Section 4.2 and schematically explained in Figure 4.1 to two different set of microarray data: wheat and Fusarium to identify sub-networks specific for a set of conditions. Compared to synthetic data, real studies are noisier and often affected from high level of bias. We applied a pre-processing step involving Robust Multichip Average method (Irizarry et al. 2003) and redundancy-adjusted Pearson correlation coefficient according to the method described in Obayashi et al. (2011). Given the high dimensionality of the wheat dataset containing more than 60000 genes, two variable-selection techniques are combined and the informative genes reduced of three orders of magnitude. GRNs for each study are calculated and through a combination of graph similarity and clustering the consensus network are calculated for each study-cluster. Once the data are discretized we finally derived the unique-networks using Bayesian networks. We then exploit the inference ability of BNs to calculate intra and inter cluster prediction accuracy for each gene across all study-clusters. To biologically support our findings we explore two well-developed tools: Mapman (Thimm et al. 2004) and AIC-MICA (Lysenko et al. 2011) knowing that specific mechanisms must be carried on by paths of genes already known to be involved in that function.

For the wheat dataset the pipeline managed to distinguish three clusters of studies two belonging to stress-enriched conditions and one to non stress. The related unique-networks found are of small size and show clear gene paths. Throughout the whole process the results appear to be robust. The clustering technique, even if using the simple k-means algorithm, combined with the sensitivity measure returns study-clusters that reflect the study description in Table 5.1. The size of the final unique-network indicates that the unique networks pipeline can discriminate important gene-paths and avoid uninformative connections. Thanks to the high value of intra and inter cluster prediction accuracy and the results extrapolated from Mapman, the AIC-MICA algorithm and the literature research conducted (Section 5.5) means that we are confident that the genes, and consequently the links between them in these subnetworks are truly involved in the underlying mechanisms.

The Fusarium dataset, is not as straightforward. First of all the studies collected do not show as clear a distinction of generic conditions, but k-means still identifies one set of conditions that are graphically similar and therefore clusters them together. Consequently the unique-networks identified are not specific for one particular generic condition but still identify highly predictive genes that are specific for the group of studies in the study-cluster. These results show similarity to the high level of noise in the simulated data.

Finally, based on these biological findings we can conclude that our pipeline is a robust and

reliable method to analyse large sets of transcriptomic data. It easily detects the main complex relationships between transcriptional expression of genes specific for different conditions and also highlights structures and nodes that could be potential targets for further research.

# Chapter 6

# Cancer data and logic application

## 6.1 Introduction

In the previous chapter we applied our method (UNIP) to two real datasets: wheat and Fusarium. In addition, to qualify the quality of our findings we first explore genes' functions using Mapman (Thimm et al. 2004) and the AIC-MICA algorithm (Lysenko et al. 2011) and then explore the literature for further information.
Wheat and its combination of studies resembled the performances obtained when the pipeline was applied to a dataset with low level of noise (see Chapter 4.4). Fusarium on the other hand showed a different set of studies where no clear division in clusters was visible. This results showed similarities with the case of high level of noise in the simulated dataset but still identified several highly predicted and predictive genes.

In this chapter, in order to demonstrate the general applicability of our pipeline, we apply UNIP to a new set of real studies, but all relating to the same generic condition: human cancer. We select four independent datasets of different kinds of cancer: breast, ovarian, medullary-breast and lung.
An organism affected by cancer is characterised by the ability to sustain chronic proliferation, evade growth suppressors and avoid apoptosis, all of which are caused by genome instability (Hanahan & Weinberg 2011) which defines changes in gene expression which in turn clearly affect the underlying mechanisms among the genes involved.

Zhang J. et al. (2012) select several cancer studies together with several control ones. For each dataset gene-pair expression correlation is computed and then used to build a frequency table whose values are used to build a weighted gene co-expression frequency network. Sub-networks with similar members are identified and iteratively merged together to generate two

final networks one representing the underlying mechanism in cancer and the other in the healthy tissues. Compared to Zhang's work we go beyond simple pairwise correlation analysis and explore the *differences* between similar studies rather than the similarities.
We exploit Genecards encyclopaedia (Safran et al. 2010) and its tools to identify the genes uniquely involved in each cancer type and measure the significance of these findings using the probability score used in Swift et al. (2004).
Finally, we develop a user-friendly application to detect both unique connections and genes using AND and OR logic operators in order to select different types of conditions (here, cancer types).

The remainder of this Chapter is organized as follows. Section 6.2 describes how the pipeline has been adapted for the analysis of the cancer datasets. Section 6.3 shows the results obtained with the cancer datasets. Section 6.4 explains how the logic interface was developed and how does it work with the help of some examples. Finally, Section 6.5 discusses the results and derives the conclusions of the chapter.

## 6.2 Method description

We now focus on the analysis of human microarray datasets based on different kinds of cancer. We downloaded 4 independent cancer data from the NCBI GEO database (Edgar et al. 2002). To avoid inter-platform bias we selected studies obtained using the Affymetrix HU133 Plus 2.0 Genechip platform. We included only studies with a substantial number of samples. No controls were available. The four chosen studies are listed in Table 6.1 with a description summary and the corresponding size (number of samples) of each study.
Each downloaded raw series of data contains a total of 54675 genes and a different number of samples, specified in Table 6.1. Firstly, the *rma* (Robust Multi-Array Average) (Irizarry et al. 2003) expression measure is applied as a pre-processing step to convert each AffyBatch object (class representation for Affymetrix GeneChip probe level data) into a normalized numeric matrix.

| Study number | Study ID | Study title | Samples |
|---|---|---|---|
| 1 | GSE18864 | Triple Negative Breast Cancer | 84 |
| 2 | GSE9891 | Ovarian Tumour | 285 |
| 3 | GSE21653 | Medullary Breast Cancer | 266 |
| 4 | GSE10445 | Adenocarcinoma and large cell Lung Carcinoma | 72 |

Table 6.1: Cancer datasets identification code, description and samples number.

### 6.2.1 Variable selection

The high discrepancy between the number of genes (54675) and the samples (refer to Table 6.1) measured simultaneously in microarray data leads to the necessity of reducing the number of variables (genes) involved in the analysis. Unlike the previously analysed datasets (wheat and Fusarium) these studies do not contain control samples. As a result, standard deviation thresholding is not immediately applicable as a first step and therefore it is necessary to find a valid alternative to it.

R statistics provides the *pvac* package (Lu & Bushel 2010) which applies the PCA (Principal Component Analysis) (Pearson 1901) and returns a subset of the original variables: the closest to the principal components identified.

To further refine the variable reduction, the standard deviation of each gene across all the samples in each separate study is calculated and only genes with $sd \geq 1.5$ in at least one of the 4 studies are selected. In fact, the genes that variate the most among patients are probably the ones that are active the most. The reduced datasets are used as input to the following steps of the analysis.

At this point we apply glasso with the penalization parameter $\rho = 0.05$, to build a GRN for each study dataset.

In addition, to further improve the sparsity and reduce the nodes involved, we maintain only the connections with an inverse covariance value greater or equal to 0.8.

In this study we only have four cancer datasets and we are interested in identifying the unique mechanism for each of them, so we don't apply the consensus network algorithm but we consider each of the four studies as a study-cluster of one element and the related glasso-network (built earlier) as the consensus network for that study-cluster. Given each GRN (4 GRNs - one per each cancer study), each *unique-network* consists of the same set of edges in the network under consideration except those that also exist in the remaining ones. We choose to measure the reliability of the unique-networks through prediction using Bayesian

Networks (BNs) (Heckerman et al. 1995, Friedman et al. 2000) which naturally perform this using inference, given the graphical structure obtained using the genes involved in the unique-networks provided by *glasso*.

Given the unique edges in the *glasso*-derived networks we first build one BN for each of the study-clusters and then identify the most predictive (how well it predicts other expression level values) and predictable (how well its expression level values are predicted) genes within (intra) and outside (inter) the study using the leave one out cross validation technique as described in details in section 4.2.6. The idea is that genes that are predictive or predicted better within the selected study than on other studies are more likely to be relevant to the unique-network.

### 6.2.2 Genecards validation and probability score

As we detect study-specific sub-networks we also want to verify that our method captures study-specific genes. We query GeneCards encyclopaedia (Safran et al. 2010) selecting each cancer type to obtain the list of genes that are known to be involved in each of them. We compare the list for each study to the others and select the genes that appear *only in the study under consideration*. To compare the unique-gene list for each type of cancer with the genes found in the corresponding unique-network, we apply the NBH (normal approximation of the binomial approximation of the hypergeometric approximation) probability score developed in Swift et al. (2004) used to test the significance of observing multiple genes with known function in a given cluster against the null hypothesis of this happening by chance. This score is based on the hypothesis that, if a given cluster, $i$ of size $s_i$, contains x genes from a defined functional group of size $k_j$, then the chance of this occurring by chance follows a binomial distribution and is defined by: Pr(Observing $x$ from group $j$) $= \binom{k_j}{x} p^x q^{k_j - x}$ where $p = \frac{s_j}{n}, q = 1 - p$ and $n$ is the number of genes in the dataset. As in this paper, when $k_j$ and $x$ are very large Pr cannot be evaluated. Therefore we use the normal approximation of the binomial distribution where: $z = \frac{x-\mu}{\sigma}$, $\mu = k_j p$ and $\sigma = \sqrt{k_j pq}$. Values of $z$ above zero mean that the probability of observing $x$ elements from functional group $j$ in cluster $i$ by chance is very small (values of $z \geq 2.326$ correspond to a probability less than 1%). The test performed is the one tailed test.

### 6.2.3 Logic and GUI

Finally a user interface has been developed using the R package *shiny* (RStudio & Inc. 2014). This interface allows the user to input the networks obtained with glasso and let the user choose which combination of unique networks to identify, using the logic operators AND and NOT. For example setting **1 AND 2 - NOT 3** will identify the sub-networks that study 1 and 2 have

in common but do not appear in study 3. The unique sub-networks for that rule/pattern are identified and plotted on the interface together with the list of genes involved. The user has the possibility to save the network in a .tiff file and the list of genes involved in .csv format.

## 6.3 Results and applications

In this study four cancer datasets are explored: breast, ovarian, medullary breast (a subtype of breast cancer) and lung, in human patients. Each dataset contains a different number of samples (see Table 6.1). The variable selection approach reduces the number of variables/genes to analyse from 54675 to 1629. Variable reduction is followed by the implementation of *glasso* with the parameter $\rho = 0.05$ and the application of a threshold on the inverse covariance values set to 0.8. Given the glasso networks for each study we consider only the edges that are present in the network under consideration but not in the others. Once the unique-connections are detected, the genes involved are used to build a BN for each study called unique-networks (U-Ns). The unique-networks obtained for all the studies are shown in Figure 6.1, 6.2, 6.3 and 6.4. Nodes with grey background indicate a prediction accuracy for the nodes greater than 0.6. Isolated nodes do not have connections due to the structure differences between glasso U-Ns and Bayesian U-Ns. Nodes are labelled with numbers, directly corresponding to the gene ID (see Appendix), for visualization purposes.

Because of the study description in Table 6.1, we would expect breast cancer to be very similar (involving almost the same genes) to medullary breast cancer and slightly less similar to ovarian, but very different from lung cancer. This implies that the average internal prediction for each study will not differ much from the external prediction. The internal vs external prediction for each study shown in Figure 6.5 reveals, as expected a very clear difference only in Network 2 and 4, ovarian and lung cancer respectively, with a small difference in 1 and 3. This deduction is supported by the p-values obtained from the applied t-test as shown in Table 6.2.

| Study number | Study ID | Study title | P-value |
|---|---|---|---|
| 1 | GSE18864 | Triple Negative Breast Cancer | 0.55 |
| 2 | GSE9891 | Ovarian Tumour | 0.00 |
| 3 | GSE21653 | Medullary Breast Cancer | 0.02 |
| 4 | GSE10445 | Adenocarcinoma and large cell Lung Carcinoma | 0.00 |

Table 6.2: Cancer datasets identification code, description and the p-values obtained from the t-test.

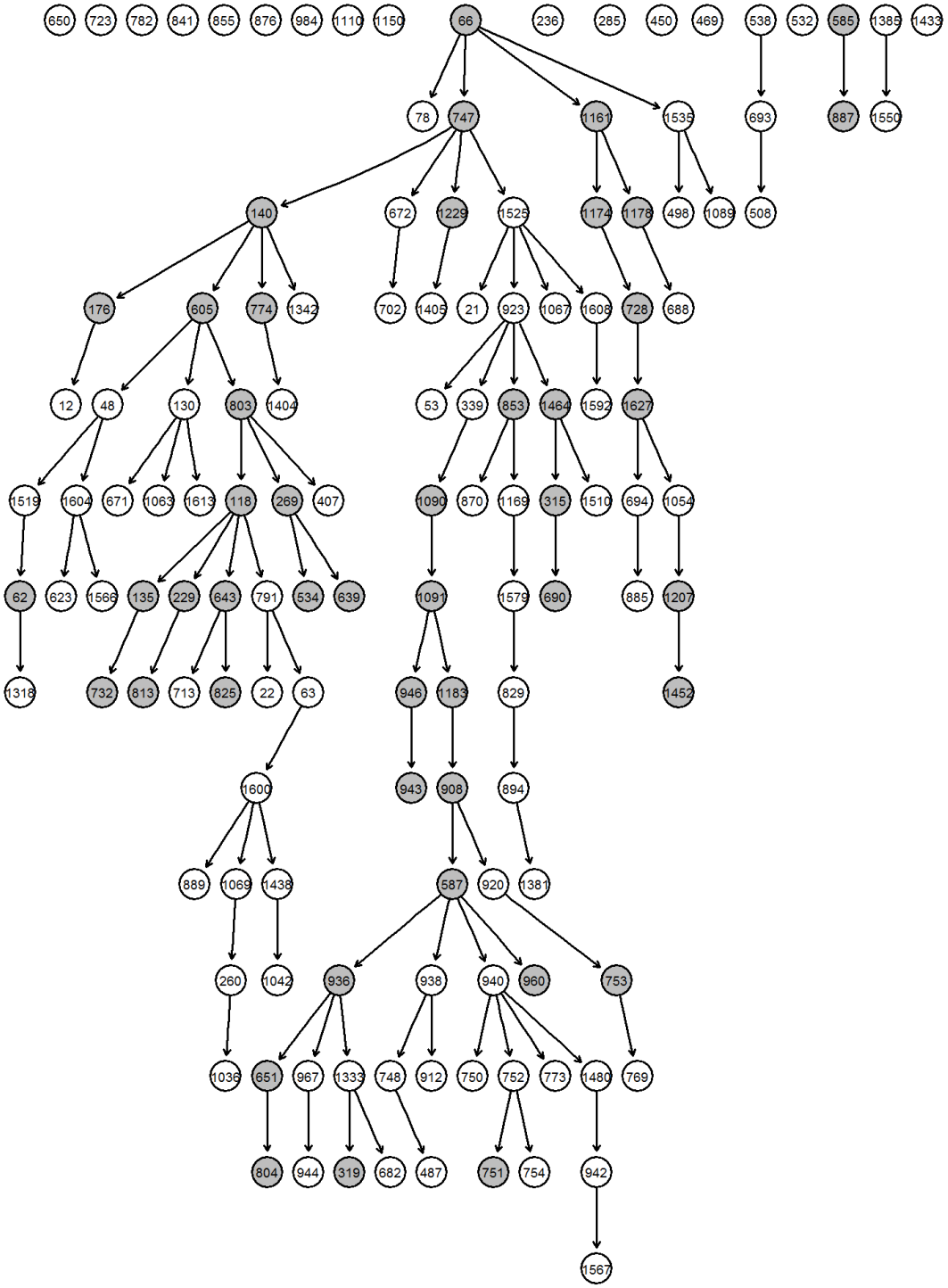



Figure 6.1: Bayesian unique-network for breast cancer.

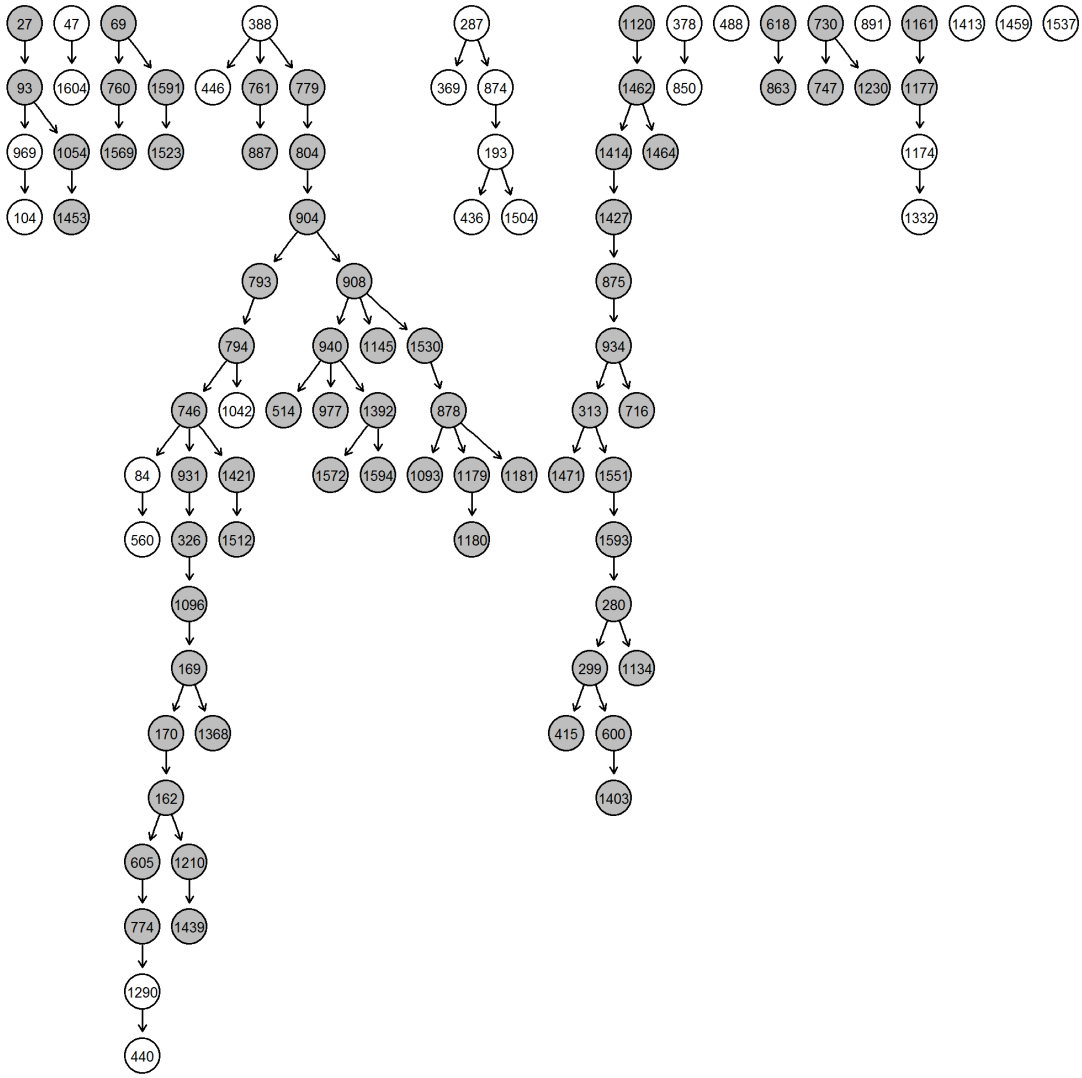



Figure 6.2: Bayesian unique-network for ovarian cancer.

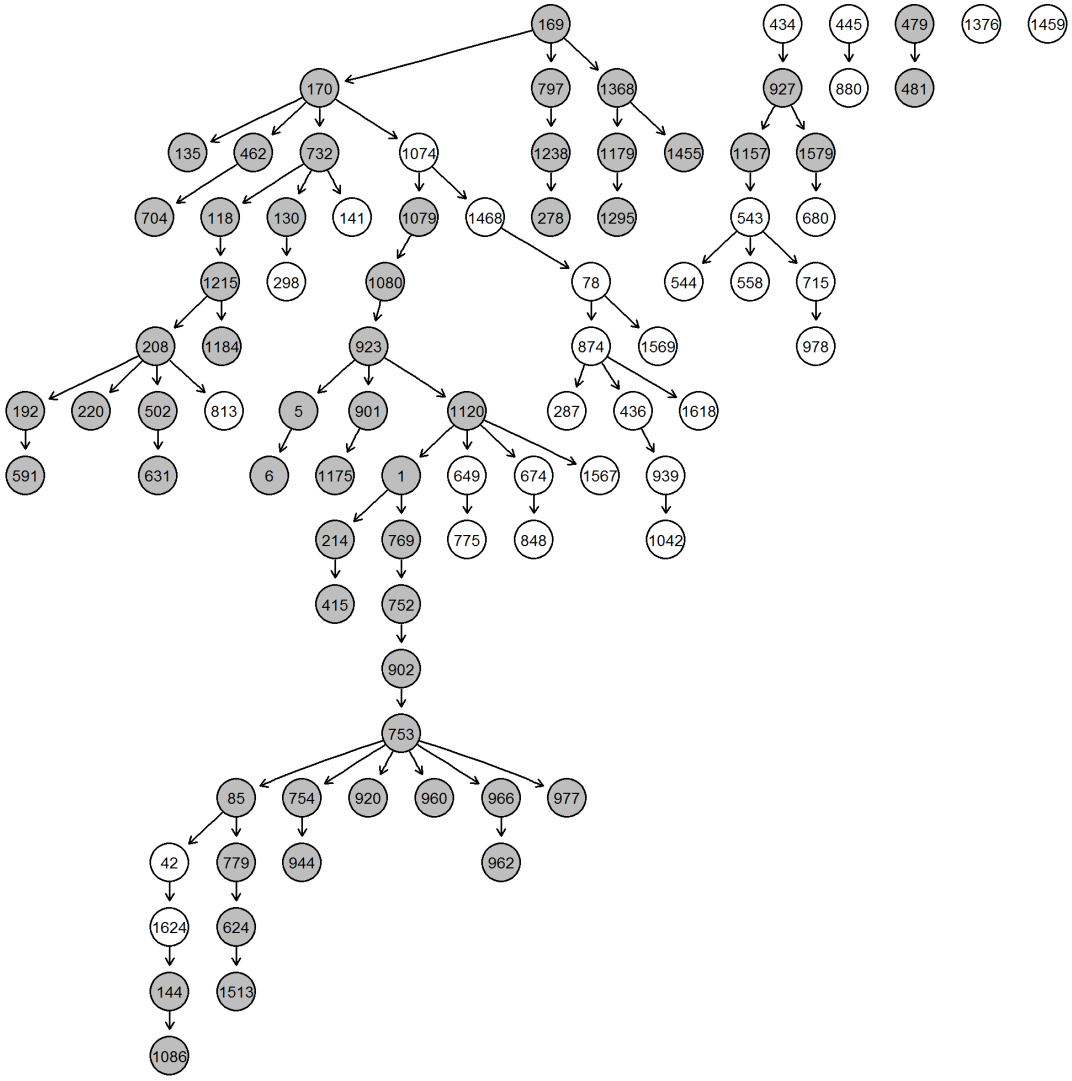



Figure 6.3: Bayesian unique-network for medullary-breast cancer.

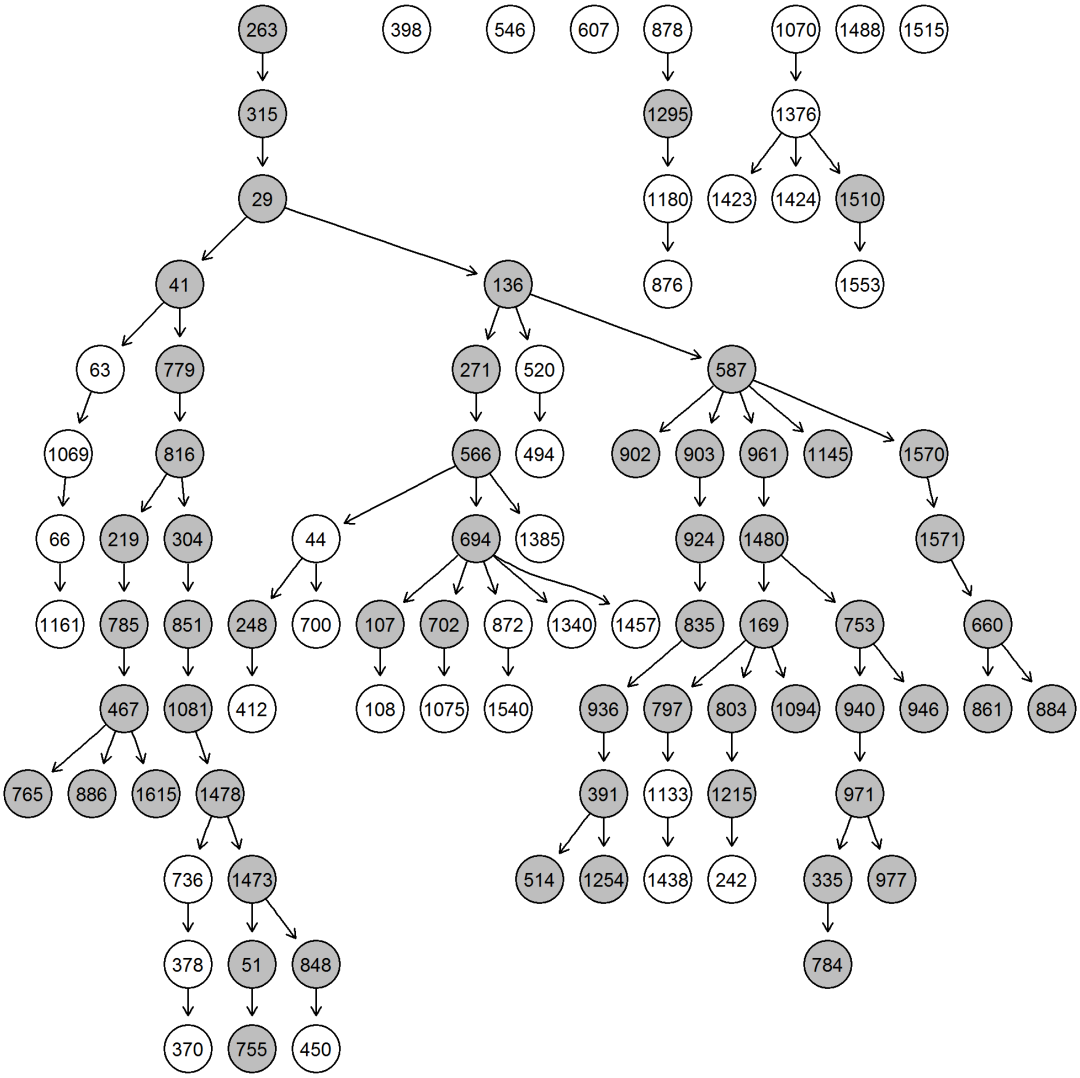



Figure 6.4: Bayesian unique-network for lung cancer.

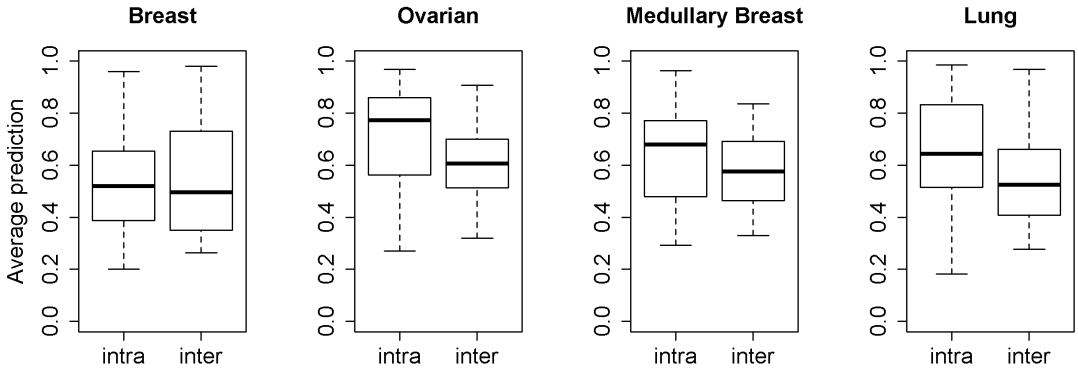



Figure 6.5: Internal (intra) vs External (inter) prediction accuracy for each study averaged among all genes involved in the related unique-network.

### 6.3.1 Identification of unique-genes through GeneCards

We now evaluate the significance of detecting the identified unique-genes by calculating the NBH probability score using the normal approximation. For this paper $s_i$ is the size of each unique network, $k_j$ the number of genes in the unique gene-list obtained for each cancer type comparing the GeneCards gene lists, $x$ the number of genes that are present on both the unique network and the corresponding unique gene-list and $n$ is the number of genes in the original unprocessed dataset. The results in Table 6.3 show the $z$-score and the corresponding $p$-value indicating that the probability of observing $x$ elements from functional group $j$ in cluster $i$ by chance is in all four cases very small. This implies that the unique genes identified by our pipeline are highly significant in all studies.

Parameters values for each study

| Study number | Study ID | $s_i$ | $k_j$ | $x$ | $n$ | $z$-score | p-value |
|---|---|---|---|---|---|---|---|
| 1 | GSE18864 | 117 | 2982 | 11 | 54675 | 1.83 | $\leq 3.4\%$ |
| 2 | GSE9891 | 61 | 692 | 4 | 54675 | 3.68 | $\leq 1\%$ |
| 3 | GSE21653 | 89 | 0 | 0 | 54675 | N/A | $\leq 1\%$ |
| 4 | GSE10445 | 80 | 240 | 3 | 54675 | 4.47 | $\leq 1\%$ |

Table 6.3: Parameters values, z-score and p-value for each study.

### 6.3.2 Gene-a-la-carte and source selection

Gene cards research domain includes several sources among which: Kegg (Kanehisa et al. 2000), UniProtKB (Magrane et al. 2011), GO (Ashburner et al. 2000) and so on. When we obtain the list of genes involved in each disease we don't automatically see where the information regarding each gene involvement come from. Assuming that for the most part the selection of the genes is done made on the content of the articles in Pubmed where the genes are cited, we now explore the genes' information from some of the available selected based on which are considered relevant to us.

*Gene a la carte* is a tool of Genecards where the user input the list of genes of interest and the sources (one or multiple) he/she is interested in retrieving. In this work we input each list of unique-genes and select the following sources: UniProtKB (Magrane et al. 2011) for the functions, Biosystems (Geer et al. 2009), KEGG (Kanehisa et al. 2000) and UniProt/UniPathway (Morgat et al. 2011) for the genes pathways, Novoseek (a tool to extract knowledge from biological databases and text repositories), Malacards (Rappaport et al. 2013) and UniProt for the diseases and PubMed for the publications IDs.

Inputting the list of unique-genes involved in breast cancer (see Table 6.4) we find that 7 of the 11 genes have been assigned a function in UniProt, 5 are involved in pathways in both BioSystems and KEGG while only 2 in UniPathway. 8 are indicated in Malacards, 6 in Novoseek and only 3 in UniProt-disorders. As expected all 11 genes are mentioned in several PubMed articles which makes it the most important source. Only one gene HBA2 (involved in oxygen transport from the lung to the various peripheral tissues and in the Selenium pathway) appears in all the sources selected, immediately followed by FGG which is missing only in UniPathway. TMEM45B and FSIP1 instead appear only in 1 of the 8 sources - PubMed. MAGEA12 is the only one that is directly associated with tumour transformation and progression. Following PubMed, Uniprot Function and Malacards contain information on 7 and 8 genes out of 11 respectively and therefore they are the sources from where most of the information come from.

For the case of ovarian cancer FSTL1 (involved in cell proliferation and differentiation) is registered in Uniprot both functions and Pathways, in MalaCards and Novoseek diseases and of course, in Pubmed. RAD51AP1 results to be involved in DNA damage response pathway where DNA damage is one of the hallmarks of tumours while the others seems to be involved in arthritis and muscular dystrophy. Here again Uniprot Function and Malacards contain entries for most of the genes: 4 and 3 out of 5 respectively.

While medullary breast show no unique-genes, probably due to the high similarity to breast cancer, lung instead reported 3 unique-genes - refer to Table 6.4. MALAT1 is recognized to be

involved in lung carcinoma (Malacards) SFTPC and MZB1 in pulmonary surfactant metabolism dysfunction (Malacards and Novoseek).

As expected, Medullary breast cancer analysis did not return any unique-genes due to the high similarity to breast cancer. On the other hand, breast cancer indicate several genes uniquely involved in it. Out of 11 genes only 2 (FSIP1 and TMEM45B) are recognised in less than half of the sources we explored. For the ovarian cancer only 1 gene out of 5 (SPRR1A) is recognised in 4 of the 10 sources available. Finally for the lung cancer 2 genes (MALAT1 and MZB1) have entries in 4 sources and the remaining gene (SFTPC) in 7 sources out of 10.

Based on these findings we conclude that although Pubmed and its articles are the most important source of information, also other sources, easier and quicker to explore, are also extremely useful. In particular, UniProt followed by Malacards resulted to be the most informative.

Unique-genes in each cancer study

| Breast | Ovarian | Medullary-Breast | Lung |
|---|---|---|---|
| FSIP1 | RAD51API | | MALT1 |
| PCSK1 | FSTL1 | | SFTPC |
| ADH1B | SPRR1A | | MZB1 |
| TFPI | COL12A1 | | |
| MAGEA12 | | | |
| RPS11 | | | |
| HBA2 | | | |
| FGG | | | |
| ODAM | | | |
| THEM45B | | | |
| RERG | | | |

Table 6.4: List of the identified unique-genes in each study.

## 6.4 Interface description - Logic

As a final part of this work we developed a user interface to derive unique-connections and unique-genes of a given set of studies. The application is composed of two main panels one intended for the selection of the set of parameters and the other for the visualization of results. The first panel is situated on the left hand-side and includes three browse buttons:

- **Load continuous data:** to allow loading of datasets of studies in the format obtained after the rma pre-process step.
- **Load adjacency matrix:** to browse the corresponding adjacency matrix of each study network.
- **Studies description:** to load a table with the corresponding study-number and description.

Once all files have been uploaded a new part is automatically generated to allow the combination of AND and NOT logic operators among the studies under consideration. The implementation of two boxes help the user to easily select which studies to include in the AND box and which ones in the NOT box. The decision is guided through the table that indicates each study description. Figures 6.6 and 6.7 show the Logic Application interface. The example represented in Figures 6.6 shows the case where the user wants to visualize the unique-connections and the list of related genes that study 1 AND 4 have in common but do not appear in study 2. Following the same example, if the NOT box is left empty by the user, the algorithm will automatically assume that the resulting unique-connections network needs to include all the connections shared by the studies 1 and 4 but that do no appear in any of the other studies in the set. This was implemented for practical reason based on what the user is most probably interested in finding. Finally the user can use the buttons at the end of each tab to save the results.

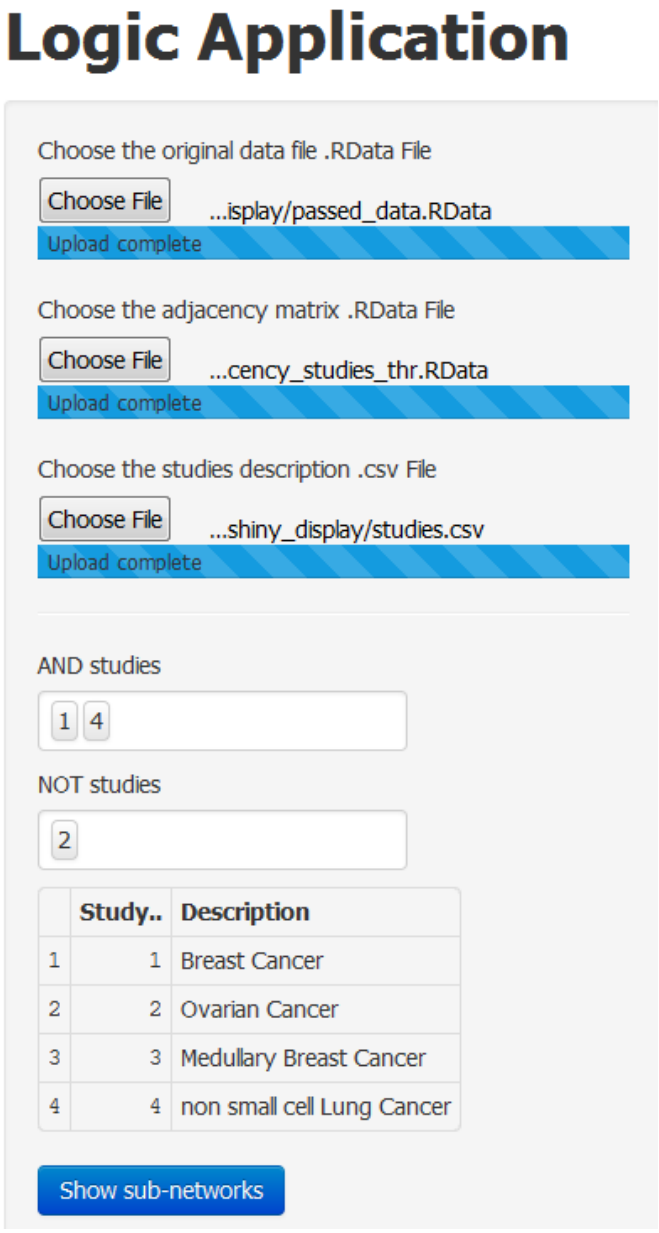



Figure 6.6: Left hand-side panel of the Logic Application interface. The figure shows the three loading buttons and the AND and NOT boxes for the studies logic combination. This example shows the case where the user wants to visualize the unique-connections and the list of related genes that study 1 AND 4 have in common but do not appear in study 2.

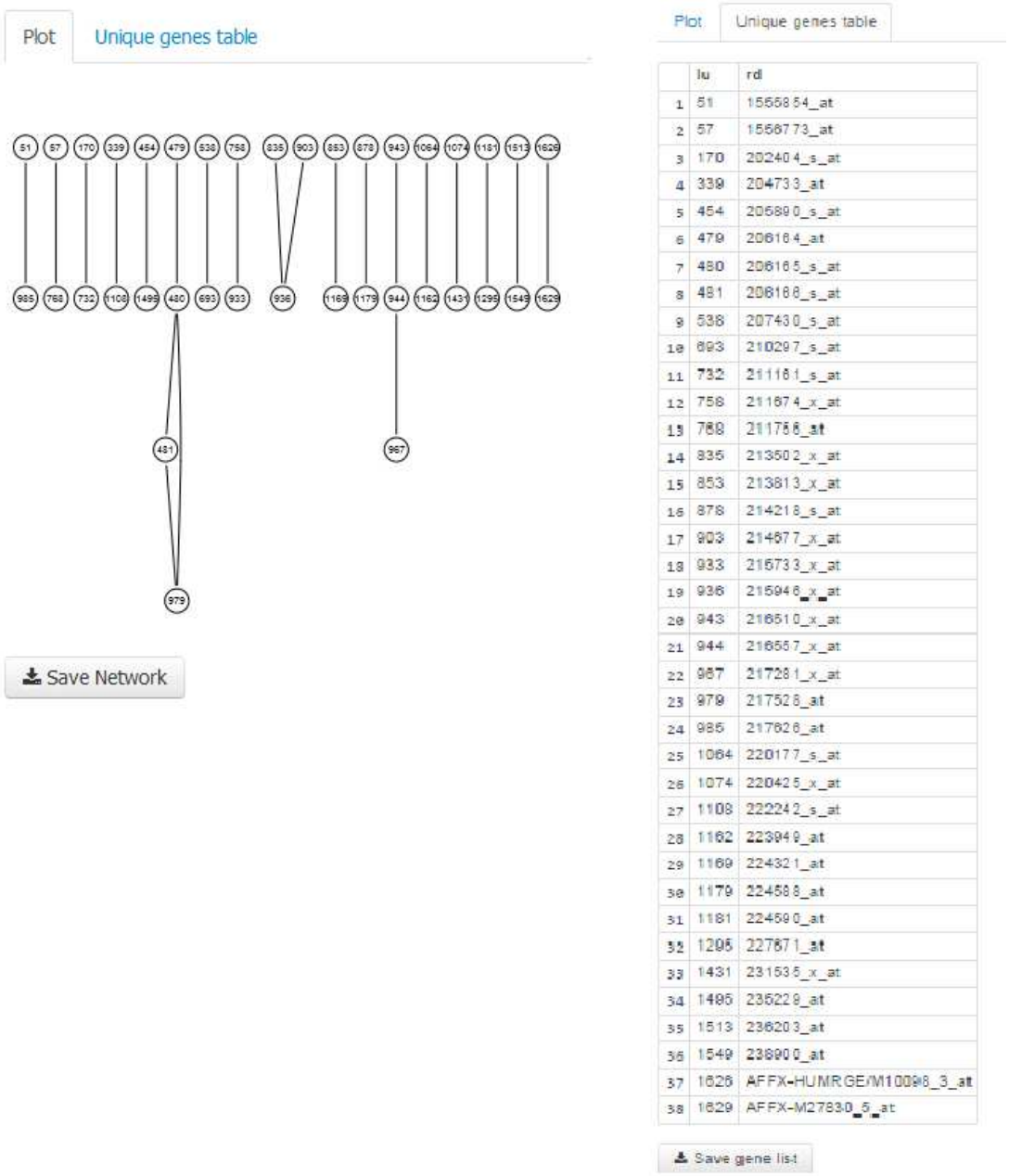



Figure 6.7: Right hand-side of the Logic Application interface. The figure shows both tabs of the results panel placed side by side. The first shows the unique-connections network and the other the table containing the correspondence between genes number in the network and real names.

## 6.5 Discussion

In this chapter we applied the UNIP pipeline to four different cancer datasets to show the general potential of UNIP on different microarray data. We focus on the discovery of unique-networks for each study skipping the identification of study-cluster and relative consensus-networks. In addition we explore the concept of unique-genes using GeneCards and its internal tools. We support our results using prediction accuracy and a score to test the significance of identifying a subset of unique genes. Furthermore, we developed a user interface to allow the user to combine studies under analysis using AND and NOT logic operators in order to derive the unique-connections and genes. Based on the results, our pipeline proved once again to be reliable and robust and identifies the mechanisms and the genes involved in them that characterize the studies under consideration.

The following chapter discusses the methods and the findings of this thesis, highlighting contributions of the research, limitations and suggests future works.

# Chapter 7

# Conclusions

This chapter discusses the conclusions reached, based on the research presented in this thesis. Firstly the research contributions are outlined. Followed by an analysis of the limitations. Finally, a list of potential future work addressing both research limitations and extending the applicability of the work.

## 7.1 Thesis contributions

### 7.1.1 Unique-networks

Literature analysis showed that researchers focus their attention to discover underlying genetic mechanisms *common* to a set of studies. In this research instead we introduced and fully define the concept of unique-networks. Unique-networks are gene regulatory networks and sub-networks that are found to be specific for one or a set of studies selected. While consensus mechanisms highlight what different conditions applied to the same organism have in common, unique mechanisms instead highlight what make them different.

### 7.1.2 Unique Network Discovery Pipeline

We developed a pipeline to semi-automatically identify unique-networks for one or a set of studies.

Deriving unique-networks from the data is not a straightforward process. Microarrays, despite their popularity, have several issues which include noise and bias in addition to a large discrepancy between the size of samples and the size of genes measured simultaneously.

In Chapter 4 we described how we implemented the Unique Network Discovery Pipeline (UNIP)

and quantified its performances based on a synthetic dataset. UNIP first integrates a set of studies on the same organism but subjected to different conditions. The larger number of samples allows us to build reliable gene regulatory networks for each group of similar conditions (study-clusters). Then, the networks are compared and those links that are present in the network under consideration but not in the others are considered together with the genes involved to apply Bayesian networks and derive the unique-networks that are specific for the study or set of studies under consideration. This pipeline take a set of studies as input and returns the corresponding unique networks. To allow the user to customize the results based on the information he/she needs to retrieve, several parameters can be tuned along the process.

### 7.1.3 Application to several datasets

The UNIP pipeline has been applied to a synthetic dataset first, to quantify its performance. Once we were confident enough on its reliability we applied it to several sets of real data to further test the pipeline and also derive new information.
The first set of data regarded wheat. Both stress and non stress studies were included. Based on the validation process the results were robust and reliable showing a clear difference between the two classes of conditions in terms of different networks that characterised each condition based on predictive power and biological support such as Mapman and literature analysis.
Fusarium, on the other hand, included a set of control studies with no clear separation between them. Despite the results being affected by the higher level of noise, still the pipeline was able to retrieve a relevant number of highly predictive genes.
Finally, a small set of cancer datasets (only four) was analysed. Unlike the other sets of real data this one was (intentionally) too small to apply either clustering or consensus analysis and instead we derive unique-networks and genes for each study from selected cancer types. Despite the lower number of samples in some studies UNIP still reported interesting and reliable results in terms of networks that were specific to each cancer type based on predictive power and biological validation obtained through the use of Genecards and the calculation of a probability score to quantify the importance of the findings.

### 7.1.4 Unique genes and probability score

Along with the concept of unique-networks we also derive the one of unique-genes. Unique networks are derived in a way that only the unique links (unique-connections) are considered, no matter which genes are involved. This means that unique networks of different conditions can still include genes that are present in both as long as they are connected in different ways. On

the other hand, there are genes whose expression is triggered only by the organism reacting to a specific condition it is subjected to and therefore even the detection of one single gene, specific for one disease, can be highly important and further simplify the diagnosis of the correct disease in shorter time.
Following the same line of reasoning as the one for the unique-networks we identify the unique genes by selecting those that only appear in the unique-network under consideration and not the others. In addition we also include the literature knowledge and further filter the unique-genes found including only those that are known (in the literature) to be involved.
Finally, we evaluate the significance of detecting the identified unique-genes by calculating the NBH probability score using the normal approximation.

### 7.1.5 Logic Application

The set of algorithms described in this work can be difficult to apply for non-technical users. Therefore, for visualization purposes, we created a Graphical User Interface (GUI) to allow the user to explore different combinations of any set of studies.
The application takes, as input, the adjacency matrices of the corresponding networks of the studies under consideration. Then, the user applies the AND and NOT logic operators to combine the studies based on the information needed, for example, to identify sub-networks and genes that are unique to a subset of cancer types. In addition a button is implemented to download the resulting unique sub-network(s) in figure format and the corresponding list of genes involved in a csv table.
The user is allowed to make several attempts of different logic combinations to explore new studies and new datasets and eventually discover the information needed or new information to use as a base for new studies.

## 7.2 Limitations

The UNIP pipeline coupled with several external tools for the results' refinement and the validation process implied to be a robust method for this type of data. Although we detect and highlight here its limitations.

- **Data Quality.** In this work we focus on microarray datasets and the pipeline is consequently adapted to the analysis of this type of data. Microarray are well known for being often biased and characterized with high levels of noise which may reflect on the results obtained with our pipeline.

- **Small samples.** Unlike consensus-network techniques our study-networks are based on smaller set of samples which if not combined together may obstruct the reliability of the results.
- **Undefined pre-process step.** Based on the set of data that the user intends to use a different gene filtering method is required. The user must specify this based on the data structure and the number of genes.
- **Scalability.** The high number of genes usually involved in biological data is a huge computational issue. We solved this problem only for a number of variables up to 2000 applying the glasso technique which proved to scale very well.
- **Dynamic Data.** The whole pipeline is shaped on the analysis of static microarray data. Although a large amount of information is held in static data, a great deal of information is also lost when the variable time is not taken into consideration.
- **Running Time.** While the majority of the steps involved in the pipeline manage to run in a reasonable amount of time, the step involving the use of the inference to calculate the intra and inter cluster prediction-accuracy is instead extremely long. The high amount of variables involved and the numerous combinations, in fact, may even require the algorithm to run overnight.
- **Shiny.** In this work we recognized the importance of developing a user friendly interface, but shiny has been remarkably slow and therefore, we could only develop a GUI for visualization purposes and were forced to leave the actual process hidden from the user.

## 7.3 Further work

The following sections bring to the attention potential future work, based on the limitations analysed above and the extension of the method presented in this thesis.

### 7.3.1 Next Generation Sequencing

Microarray techniques are still very popular thanks to their ability to collect thousands of individual gene sequences in parallel to study gene expression and gene variation in any given cell type, time, set of conditions or treatments. Although more recently a new set of techniques called Next Generation Sequencing has been developed, which appear to be more reliable and, in certain cases, more appropriate. Biologists still show a higher preference for microarrays

but also their maturity as a technology results in an enormous amount of data still waiting to be analysed. Laboratories are now making the switch and more and more data are becoming publicly available and so an important development that could be applied to our pipeline would be to introduce new pre-processing steps or modify the already existing ones in a way that NGS data can be examined.

### 7.3.2 Application to different kind of data

Several areas now make use of networks to explore real problems: biology, medicine, economics, etc. Although, the UNIP pipeline has been created and specifically studied for biological data, the concept of unique networks combined with unique variables can be easily applied to many more kind of real data such as social networks. Few adjustments in the preprocessing step will shape UNIP in a way to be applied to all kinds of data.

### 7.3.3 Static vs dynamic data

Here we only focused on static data. We analysed only studies that did not involve time. Although, several diseases and underlying mechanisms in general vary together with time. Static data carries a lot of knowledge but also misses a great deal of information that is usually revealed by taking time into consideration.
A further step would be to explore dynamic data and the potential of their use following the adaptation of all steps of the pipeline.

### 7.3.4 Improvement of the Graphical User Interface

The importance of implementing Graphical User Interfaces (GUI) resides in the fact that it renders complex code accessible to non-technical researchers and therefore extend their usability. The GUI developed in this research and described in Chapter 6 allows the user to set different combinations of the set of studies using the AND and NOT logic operators. Once the logic combination has been set the application visualizes the resulting unique-connections and the genes involved in them.
We have made several attempts to implement more than just the visualization process but the R package Shiny, despite all its advantages, is not able to elaborate the necessary query in a reasonable amount of time due to the complexity of the algorithms involved.
In the future it would be extremely useful to implement a GUI, even using a new language to load directly the raw data and allow the user to follow step by step the process and tuning the parameters directly in the GUI to customize the results.

# Appendix A

# Additional tables and results

This appendix contains additional tables and results relating to Chapter 5 and 6.

## A.1 Chapter 5 additional tables

The tables listed in here refer to the results found applying UNIP to the wheat dataset.They represent the correspondence of genes numbers in the networks and genes names together with the genes functions detected with Mapman (Thimm et al. 2004). Table A.1 details the genes in the unique-network for the first study-cluster stress-enriched using the wheat dataset. Table A.2 details the genes in the unique-network for the second study-cluster stress-enriched using the wheat dataset. Table A.3 details the genes in the unique-network for the third study-cluster, non-stress using the wheat dataset.

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 1 | Ta.10329.17.S1_at | DNA.synthesis/chromatin structure.histone |
| 2 | Ta.10329.17.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 3 | Ta.10329.3.S1_at | DNA.synthesis/chromatin structure.histone |
| 4 | Ta.10390.1.S1_at | protein.degradation.cysteine protease |
| 5 | Ta.10480.1.S1_a_at | RNA.processing.ribonucleases |
| 6 | Ta.1139.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 7 | Ta.1161.1.S1_at | PS.lightreaction.photosystem II.PSII polypeptide subunits |
| 8 | Ta.12118.1.S1_a_at | misc.gluco-, galacto- and mannosidases |
| 9 | Ta.14034.1.a1_at | misc.protease inhibitor/seed storage/lipid transfer protein family protein |
| 10 | Ta.14475.1.S1_at | protein.degradation.ubiquitin.ubiquitin |
| 11 | Ta.14543.2.a1_at | protein.glycosylation |
| 12 | Ta.1725.3.S1_at | misc.plastocyanin-like |
| 13 | Ta.1929.1.S1_at | stress.biotic |
| 14 | Ta.1953.1.S1_x_at | RNA.regulation of transcription.unclassified |
| 16 | Ta.20949.1.a1_at | amino acid metabolism.degradation.aspartate. family.threonine |
| 17 | Ta.21342.1.S1_x_at | stress.biotic |
| 18 | Ta.22984.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 19 | Ta.23366.2.S1_x_at | misc.peroxidases |
| 21 | Ta.2402.3.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 22 | Ta.24304.2.S1_a_at | PS.lightreaction.photosystem I.PSI.polypeptide subunits |
| 23 | Ta.261.1.S1_at | stress.abiotic.heat |
| 24 | Ta.26907.1.S1_at | RNA.processing.ribonucleases |
| 25 | Ta.2747.1.S1_at | stress.abiotic.heat |
| 26 | Ta.27657.11.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 27 | Ta.27751.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 28 | Ta.27761.1.S1_x_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 29 | Ta.278.1.S1_x_at | stress.biotic |
| 30 | Ta.2784.1.a1_at | stress.biotic |
| 31 | Ta.28123.1.S1_at | protein.synthesis.ribosomal protein.prokaryotic.chloroplast.50S subunit.L28 |
| 32 | Ta.28368.2.S1_at | lipid metabolism.lipid transfer proteins etc |

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 33 | Ta.3.1.S1_at | major CHO metabolism.degradation.starch.starch cleavage |
| 34 | Ta.30501.1.S1_at | stress.biotic |
| 35 | Ta.3361.1.S1_x_at | stress.abiotic.unspecified |
| 36 | Ta.3651.1.S1_at | misc.nitrilases, *nitrile lyases, berberine bridge enzymes, reticuline oxidases, troponine reductases |
| 37 | Ta.3987.1.S1_x_at | PS.lightreaction.photosystem II.PSII polypeptide subunits |
| 38 | Ta.488.1.S1_at | redox.ascorbate and glutathione.ascorbate |
| 39 | Ta.581.2.S1_a_at | PS.photorespiration.glycine cleavage |
| 40 | Ta.601.1.a1_at | lipid metabolism.lipid degradation.lipases |
| 41 | Ta.6123.1.a1_s_at | stress.abiotic.heat |
| 42 | Ta.6572.1.S1_a_at | redox.peroxiredoxin |
| 43 | Ta.7378.18.S1_at | DNA.synthesis/chromatin structure.histone |
| 44 | Ta.7378.18.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 45 | Ta.7963.2.S1_x_at | stress.biotic |
| 46 | Ta.8665.1.S1_at | stress.abiotic.heat |
| 47 | Ta.9226.1.S1_at | stress.biotic |
| 48 | Ta.9409.1.S1_at | RNA.regulation of transcription.General Transcription |
| 49 | Ta.9574.1.S1_at | tetrapyrrole synthesis.magnesium chelatase |
| 50 | Ta.9599.1.S1_a_at | misc.glutathione S transferases |
| 51 | Ta.9679.1.a1_at | misc.peroxidases |
| 52 | Ta.9718.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 53 | Taaffx.18332.1.S1_at | stress.abiotic.heat |
| 54 | Taaffx.3720.7.S1_at | protein.degradation |
| 55 | Taaffx.38476.1.S1_at | misc.UDP glucosyl and glucoronyl transferases |
| 56 | Taaffx.449.1.a1_at | PS.calvin cyle.rubisco small subunit |

Table A.1: Correspondence of genes numbers and affymetrix names together with the functions indicated by Mapman in unique-network 1 (stress-enriched) for the wheat dataset.

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 1 | Ta.10329.17.S1_at | DNA.synthesis/chromatin structure.histone |
| 2 | Ta.10329.17.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 3 | Ta.10329.3.S1_at | DNA.synthesis/chromatin structure.histone |
| 4 | Ta.10390.1.S1_at | protein.degradation.cysteine protease |
| 5 | Ta.10480.1.S1_a_at | RNA.processing.ribonucleases |
| 6 | Ta.1130.1.S1_a_at | PS.lightreaction.photosystem II.LHC-II |
| 7 | Ta.1130.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 8 | Ta.1130.3.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 9 | Ta.1139.1.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 10 | Ta.1161.1.S1_at | PS.lightreaction.photosystem II.PSII polypeptide subunits |
| 11 | Ta.12118.1.S1_a_at | misc.gluco-, galacto- and mannosidases |
| 12 | Ta.14034.1.A1_at | misc.protease inhibitor/seed storage/lipid transfer protein (LTP) family protein |
| 13 | Ta.14475.1.S1_at | protein.degradation.ubiquitin.ubiquitin |
| 14 | Ta.14543.2.A1_at | protein.glycosylation |
| 15 | Ta.1725.3.S1_at | misc.plastocyanin-like |
| 16 | Ta.1953.1.S1_x_at | RNA.regulation of transcription.unclassified |
| 19 | Ta.21342.1.S1_x_at | stress.biotic |
| 20 | Ta.22984.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 21 | Ta.23366.2.S1_x_at | misc.peroxidases |
| 23 | Ta.2402.3.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 24 | Ta.24304.2.S1_a_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 25 | Ta.25600.1.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 26 | Ta.26907.1.S1_at | RNA.processing.ribonucleases |
| 27 | Ta.27657.11.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 28 | Ta.27751.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 29 | Ta.27761.1.S1_x_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 30 | Ta.278.1.S1_x_at | stress.biotic |
| 31 | Ta.2784.1.A1_at | stress.biotic |
| 32 | Ta.28123.1.S1_at | protein.synthesis.ribosomal protein.prokaryotic. chloroplast.50S subunit.L28 |
| 33 | Ta.28363.3.S1_x_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 34 | Ta.28368.2.S1_at | lipid metabolism.lipid transfer proteins etc |
| 35 | Ta.3.1.S1_at | major CHO metabolism.degradation.starch.starch cleavage |
| 36 | Ta.30501.1.S1_at | stress.biotic |
| 37 | Ta.30727.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 38 | Ta.30808.1.S1_s_at | PS.calvin cyle.GAP |
| 38 | Ta.30808.1.S1_s_at | glycolysis.glyceraldehyde 3-phosphate dehydrogenase |
| 39 | Ta.3361.1.S1_x_at | stress.abiotic.unspecified |
| 40 | Ta.3651.1.S1_at | misc.nitrilases, *nitrile lyases, berberine bridge enzymes, reticuline oxidases, troponine reductases |
| 41 | Ta.3987.1.S1_x_at | PS.lightreaction.photosystem II.PSII polypeptide subunits |
| 42 | Ta.488.1.S1_at | redox.ascorbate and glutathione.ascorbate |
| 43 | Ta.488.1.S1_x_at | redox.ascorbate and glutathione.ascorbate |
| 44 | Ta.581.2.S1_a_at | PS.photorespiration.glycine cleavage |
| 45 | Ta.601.1.A1_at | lipid metabolism.lipid degradation.lipases |
| 46 | Ta.6572.1.S1_a_at | redox.peroxiredoxin |
| 47 | Ta.7378.18.S1_at | DNA.synthesis/chromatin structure.histone |
| 48 | Ta.7378.18.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 49 | Ta.7963.2.S1_x_at | stress.biotic |
| 50 | Ta.9226.1.S1_at | stress.biotic |
| 51 | Ta.9409.1.S1_at | RNA.regulation of transcription.General Transcription |
| 52 | Ta.9574.1.S1_at | tetrapyrrole synthesis.magnesium chelatase |
| 53 | Ta.9599.1.S1_a_at | misc.glutathione S transferases |
| 54 | Ta.9679.1.A1_at | misc.peroxidases |
| 55 | Ta.9718.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 56 | Taaffx.3720.7.S1_at | protein.degradation |
| 57 | Taaffx.38476.1.S1_at | misc.UDP glucosyl and glucoronyl transferases |
| 58 | Taaffx.449.1.A1_at | PS.calvin cyle.rubisco small subunit |

Table A.2: Correspondence of genes number sand affymetrix names together with the functions indicated by Mapman in unique-network 2 (stress-enriched) for the wheat dataset.

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 1 | Ta.10329.17.S1_at | DNA.synthesis/chromatin structure.histone |
| 2 | Ta.10329.17.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 3 | Ta.10329.3.S1_at | DNA.synthesis/chromatin structure.histone |
| 4 | Ta.10480.1.S1_a_at | RNA.processing.ribonucleases |
| 5 | Ta.1130.1.S1_a_at | PS.lightreaction.photosystem II.LHC-II |
| 6 | Ta.1130.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 7 | Ta.1130.3.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 8 | Ta.1139.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 9 | Ta.1139.1.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 10 | Ta.1161.1.S1_at | PS.lightreaction.photosystem II.PSII polypeptide subunits |
| 11 | Ta.14034.1.A1_at | misc.protease inhibitor/seed storage/lipid transfer protein (LTP) family protein |
| 12 | Ta.14475.1.S1_at | protein.degradation.ubiquitin.ubiquitin |
| 13 | Ta.14543.2.A1_at | protein.glycosylation |
| 14 | Ta.1725.3.S1_at | misc.plastocyanin-like |
| 15 | Ta.1929.1.S1_at | stress.biotic |
| 16 | Ta.1953.1.S1_x_at | RNA.regulation of transcription.unclassified |
| 18 | Ta.20949.1.A1_at | amino acid metabolism.degradation.aspartate family.threonine |
| 20 | Ta.21342.1.S1_x_at | stress.biotic |
| 21 | Ta.22984.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 22 | Ta.23366.2.S1_x_at | misc.peroxidases |
| 24 | Ta.24304.2.S1_a_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 25 | Ta.25600.1.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 26 | Ta.261.1.S1_at | stress.abiotic.heat |
| 27 | Ta.26907.1.S1_at | RNA.processing.ribonucleases |
| 28 | Ta.2747.1.S1_at | stress.abiotic.heat |
| 29 | Ta.27657.11.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 30 | Ta.27751.2.S1_x_at | PS.lightreaction.photosystem II.LHC-II |
| 31 | Ta.27761.1.S1_x_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 32 | Ta.278.1.S1_x_at | stress.biotic |
| 33 | Ta.28123.1.S1_at | protein.synthesis.ribosomal protein.prokaryotic. chloroplast.50S subunit.L28 |

| Gene no | Affy ID | Function Pathway |
|---|---|---|
| 34 | Ta.28363.3.S1_x_at | PS.lightreaction.photosystem I.PSI polypeptide subunits |
| 35 | Ta.28368.2.S1_at | lipid metabolism.lipid transfer proteins etc |
| 36 | Ta.3.1.S1_at | major CHO metabolism.degradation.starch.starch cleavage |
| 37 | Ta.30501.1.S1_at | stress.biotic |
| 38 | Ta.30727.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 39 | Ta.30808.1.S1_s_at | PS.calvin cyle.GAP |
| 39 | Ta.30808.1.S1_s_at | glycolysis.glyceraldehyde 3-phosphate dehydrogenase |
| 40 | Ta.3361.1.S1_x_at | stress.abiotic.unspecified |
| 41 | Ta.3651.1.S1_at | misc.nitrilases, *nitrile lyases, berberine bridge enzymes, reticuline oxidases, troponine reductases |
| 42 | Ta.488.1.S1_at | redox.ascorbate and glutathione.ascorbate |
| 43 | Ta.488.1.S1_x_at | redox.ascorbate and glutathione.ascorbate |
| 44 | Ta.581.2.S1_a_at | PS.photorespiration.glycine cleavage |
| 45 | Ta.601.1.A1_at | lipid metabolism.lipid degradation.lipases |
| 46 | Ta.6123.1.A1_s_at | stress.abiotic.heat |
| 47 | Ta.6572.1.S1_a_at | redox.peroxiredoxin |
| 48 | Ta.7378.18.S1_x_at | DNA.synthesis/chromatin structure.histone |
| 49 | Ta.7963.2.S1_x_at | stress.biotic |
| 50 | Ta.8665.1.S1_at | stress.abiotic.heat |
| 51 | Ta.9226.1.S1_at | stress.biotic |
| 52 | Ta.9409.1.S1_at | RNA.regulation of transcription.General Transcription |
| 53 | Ta.9574.1.S1_at | tetrapyrrole synthesis.magnesium chelatase |
| 54 | Ta.9599.1.S1_a_at | misc.glutathione S transferases |
| 55 | Ta.9679.1.A1_at | misc.peroxidases |
| 56 | Ta.9718.1.S1_at | PS.lightreaction.photosystem II.LHC-II |
| 57 | Taaffx.18332.1.S1_at | stress.abiotic.heat |
| 58 | Taaffx.3720.7.S1_at | protein.degradation |
| 59 | Taaffx.38476.1.S1_at | misc.UDP glucosyl and glucoronyl transferases |
| 60 | Taaffx.449.1.A1_at | PS.calvin cyle.rubisco small subunit |

Table A.3: Correspondence of genes number sand affymetrix names together with the functions indicated by Mapman in unique-network 3 (non-stress) for the wheat dataset.

## A.2 Chapter 6 additional tables

The tables listed in here refer to the results found applying UNIP to the cancer datasets.They represent the correspondence of genes numbers in the networks, affymetrix id and symbol. Table A.4 details the genes in the unique-network for breast cancer.

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 12 | 1553183_at | UMODL1 |
| 21 | 1553622_a_at | FSIP1 |
| 22 | 1553678_a_at | ITGB1 |
| 48 | 1555801_s_at | ZNF385B |
| 53 | 1556012_at | KLHDC7A |
| 62 | 1558034_s_at | CP |
| 63 | 1558048_x_at | NA |
| 66 | 1558678_s_at | MALAT1 |
| 78 | 1564479_a_at | NA |
| 118 | 201438_at | COL6A3 |
| 130 | 201744_s_at | LUM |
| 135 | 201852_x_at | COL3A1 |
| 140 | 201893_x_at | DCN |
| 176 | 202450_s_at | CTSK |
| 229 | 203477_at | COL15A1 |
| 236 | 203559_s_at | ABP1 |
| 260 | 203908_at | SLC4A4 |
| 269 | 203980_at | FABP4 |
| 285 | 204260_at | CHGB |
| 315 | 204533_at | CXCL10 |
| 319 | 204563_at | SELL |
| 339 | 204733_at | KLK6 |
| 407 | 205402_x_at | PRSS2 |
| 450 | 205825_at | PCSK1 |
| 469 | 206022_at | NDP |
| 487 | 206228_at | PAX2 |
| 498 | 206407_s_at | CCL13 |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 508 | 206561_s_at | AKR1B10 |
| 532 | 207142_at | KCNJ3 |
| 534 | 207175_at | ADIPOQ |
| 538 | 207430_s_at | MSMB |
| 585 | 209116_x_at | HBB |
| 587 | 209138_x_at | IGLV3-21 |
| 587 | 209138_x_at | IGL@ |
| 605 | 209335_at | DCN |
| 623 | 209469_at | GPM6A |
| 639 | 209612_s_at | ADH1B |
| 643 | 209676_at | TFPI |
| 650 | 209720_s_at | SERPINB3 |
| 651 | 209728_at | HLA-DRB4 |
| 671 | 209937_at | TM4SF4 |
| 672 | 209942_x_at | MAGEA3 |
| 682 | 210072_at | CCL19 |
| 688 | 210145_at | PLA2G4A |
| 690 | 210163_at | CXCL11 |
| 693 | 210297_s_at | MSMB |
| 694 | 210338_s_at | HSPA8 |
| 702 | 210467_x_at | MAGEA12 |
| 713 | 210665_at | TFPI |
| 723 | 210906_x_at | AQP4 |
| 728 | 211074_at | FOLR1 |
| 732 | 211161_s_at | COL3A1 |
| 747 | 211621_at | AR |
| 748 | 211634_x_at | IGHV1-69 |
| 748 | 211634_x_at | IGHM |
| 750 | 211637_x_at | IGHV4-59 |
| 750 | 211637_x_at | IGHV4-31 |
| 750 | 211637_x_at | IGHV3-23 |
| 750 | 211637_x_at | IGHA1 |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 750 | 211637_x_at | IGHA2 |
| 750 | 211637_x_at | IGHD |
| 750 | 211637_x_at | IGHG1 |
| 750 | 211637_x_at | IGHG3 |
| 750 | 211637_x_at | IGHG4 |
| 750 | 211637_x_at | IGHM |
| 751 | 211643_x_at | IGKC |
| 751 | 211643_x_at | IGKV3D-15 |
| 751 | 211643_x_at | IGK@ |
| 752 | 211644_x_at | IGKC |
| 752 | 211644_x_at | IGK@ |
| 753 | 211645_x_at | NA |
| 754 | 211650_x_at | IGK@ |
| 754 | 211650_x_at | IGHV4-31 |
| 754 | 211650_x_at | IGHV3-23 |
| 754 | 211650_x_at | IGHV1-69 |
| 754 | 211650_x_at | IGHA1 |
| 754 | 211650_x_at | IGHD |
| 754 | 211650_x_at | IGHG1 |
| 754 | 211650_x_at | IGHG3 |
| 754 | 211650_x_at | IGHM |
| 769 | 211796_s_at | IL23A |
| 769 | 211796_s_at | TRBC2 |
| 769 | 211796_s_at | TRBC1 |
| 773 | 211881_x_at | IGLJ3 |
| 774 | 211896_s_at | DCN |
| 782 | 212092_at | PEG10 |
| 791 | 212298_at | NRP1 |
| 803 | 212667_at | SPARC |
| 804 | 212671_s_at | HLA-DQA1 |
| 804 | 212671_s_at | HLA-DQA2 |
| 813 | 212950_at | GPR116 |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 825 | 213258_at | TFPI |
| 829 | 213350_at | RPS11 |
| 841 | 213674_x_at | IGHG1 |
| 841 | 213674_x_at | IGHD |
| 853 | 213813_x_at | NA |
| 855 | 213844_at | HOXA5 |
| 870 | 214078_at | NA |
| 876 | 214183_s_at | TKTL1 |
| 885 | 214359_s_at | HSP90AB1 |
| 887 | 214414_x_at | HBA1 |
| 887 | 214414_x_at | HBA2 |
| 889 | 214433_s_at | SELENBP1 |
| 894 | 214461_at | LBP |
| 908 | 214836_x_at | IGKC |
| 908 | 214836_x_at | IGKV1-5 |
| 908 | 214836_x_at | IGK@ |
| 912 | 214973_x_at | IGHD |
| 920 | 215176_x_at | IGKC |
| 920 | 215176_x_at | IGK@ |
| 923 | 215304_at | NA |
| 936 | 215946_x_at | IGLL3P |
| 938 | 216207_x_at | IGKV1D-13 |
| 938 | 216207_x_at | IGKV1-5 |
| 938 | 216207_x_at | IGKV1D-8 |
| 938 | 216207_x_at | IGKC |
| 940 | 216401_x_at | NA |
| 942 | 216491_x_at | IGHM |
| 943 | 216510_x_at | IGHV4-31 |
| 943 | 216510_x_at | IGHV3-23 |
| 943 | 216510_x_at | IGHA1 |
| 943 | 216510_x_at | IGHG1 |
| 943 | 216510_x_at | IGHM |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 944 | 216557_x_at | LOC100291917 |
| 944 | 216557_x_at | IGHV4-31 |
| 944 | 216557_x_at | IGHV3-48 |
| 944 | 216557_x_at | IGHA1 |
| 944 | 216557_x_at | IGHD |
| 944 | 216557_x_at | IGHG1 |
| 944 | 216557_x_at | IGHG3 |
| 944 | 216557_x_at | IGHM |
| 946 | 216576_x_at | IGKC |
| 946 | 216576_x_at | IGK@ |
| 960 | 217157_x_at | IGKC |
| 960 | 217157_x_at | IGK@ |
| 967 | 217281_x_at | LOC100290036 |
| 967 | 217281_x_at | IGHV4-31 |
| 967 | 217281_x_at | IGHA1 |
| 967 | 217281_x_at | IGHA2 |
| 967 | 217281_x_at | IGHG1 |
| 967 | 217281_x_at | IGHG2 |
| 967 | 217281_x_at | IGHG3 |
| 967 | 217281_x_at | IGHM |
| 984 | 217590_s_at | TRPA1 |
| 1036 | 219508_at | GCNT3 |
| 1042 | 219612_s_at | FGG |
| 1054 | 219850_s_at | EHF |
| 1063 | 220133_at | ODAM |
| 1067 | 220196_at | MUC16 |
| 1069 | 220232_at | SCD5 |
| 1089 | 221577_x_at | GDF15 |
| 1090 | 221651_x_at | IGKC |
| 1090 | 221651_x_at | IGK@ |
| 1091 | 221671_x_at | IGKC |
| 1091 | 221671_x_at | IGK@ |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1110 | 222281_s_at | LOC100505650 |
| 1150 | 223642_at | ZIC2 |
| 1161 | 223940_x_at | MALAT1 |
| 1169 | 224321_at | TMEFF2 |
| 1174 | 224559_at | MALAT1 |
| 1178 | 224568_x_at | MALAT1 |
| 1183 | 224795_x_at | IGK@ |
| 1183 | 224795_x_at | IGKC |
| 1207 | 225645_at | EHF |
| 1229 | 226192_at | AR |
| 1318 | 228143_at | CP |
| 1333 | 228592_at | MS4A1 |
| 1342 | 228821_at | ST6GAL2 |
| 1381 | 229638_at | IRX3 |
| 1385 | 229782_at | RMST |
| 1404 | 230319_at | NA |
| 1405 | 230323_s_at | TMEM45B |
| 1433 | 231597_x_at | NA |
| 1438 | 231771_at | GJB6 |
| 1452 | 232360_at | EHF |
| 1464 | 232944_at | NA |
| 1480 | 234764_x_at | IGLV1-44 |
| 1480 | 234764_x_at | IGLV1-36 |
| 1510 | 236085_at | CAPSL |
| 1519 | 236308_at | VSTM2A |
| 1525 | 237086_at | FOXA1 |
| 1535 | 238021_s_at | CRNDE |
| 1550 | 239006_at | SLC26A7 |
| 1566 | 240065_at | FAM81B |
| 1567 | 240161_s_at | CDC20B |
| 1579 | 241617_x_at | NA |
| 1592 | 242517_at | KISS1R |

| Breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1600 | 243489_at | NA |
| 1604 | 243929_at | NA |
| 1608 | 244745_at | RERG |
| 1613 | 37512_at | HSD17B6 |
| 1627 | AFFX-HUMRGE/M10098_5_at | NA |

Table A.4: Correspondence of genes numbers, affymetrix names and symbols for the **breast** cancer dataset.

| Ovarian cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 27 | 1554679_a_at | LAPTM4B |
| 47 | 1555800_at | ZNF385B |
| 69 | 1559459_at | LOC613266 |
| 84 | 1567458_s_at | RAC1 |
| 93 | 200641_s_at | YWHAZ |
| 104 | 201118_at | PGD |
| 162 | 202310_s_at | COL1A1 |
| 169 | 202403_s_at | COL1A2 |
| 170 | 202404_s_at | COL1A2 |
| 193 | 202831_at | GPX2 |
| 280 | 204146_at | RAD51AP1 |
| 287 | 204272_at | LGALS4 |
| 299 | 204415_at | IFI6 |
| 313 | 204508_s_at | CA12 |
| 326 | 204620_s_at | VCAN |
| 369 | 205009_at | TFF1 |
| 378 | 205064_at | SPRR1B |
| 388 | 205239_at | AREG |
| 415 | 205483_s_at | ISG15 |
| 436 | 205650_s_at | FGA |
| 440 | 205696_s_at | GFRA1 |
| 446 | 205767_at | EREG |
| 488 | 206239_s_at | SPINK1 |
| 514 | 206641_at | TNFRSF17 |
| 560 | 208310_s_at | FSTL1 |
| 560 | 208310_s_at | CCZ1B |
| 560 | 208310_s_at | CCZ1 |
| 600 | 209290_s_at | NFIB |
| 605 | 209335_at | DCN |
| 618 | 209437_s_at | SPON1 |
| 716 | 210735_s_at | CA12 |
| 730 | 211110_s_at | AR |

| Ovarian cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 746 | 211571_s_at | VCAN |
| 747 | 211621_at | AR |
| 760 | 211682_x_at | UGT2B28 |
| 761 | 211696_x_at | HBB |
| 774 | 211896_s_at | DCN |
| 779 | 211991_s_at | HLA-DPA1 |
| 793 | 212344_at | SULF1 |
| 794 | 212353_at | SULF1 |
| 804 | 212671_s_at | HLA-DQA1 |
| 804 | 212671_s_at | HLA-DQA2 |
| 850 | 213796_at | SPRR1A |
| 863 | 213993_at | SPON1 |
| 874 | 214135_at | CLDN18 |
| 875 | 214164_x_at | CA12 |
| 878 | 214218_s_at | XIST |
| 887 | 214414_x_at | HBA1 |
| 887 | 214414_x_at | HBA2 |
| 891 | 214451_at | TFAP2B |
| 904 | 214768_x_at | IGKC |
| 904 | 214768_x_at | IGKV1-5 |
| 908 | 214836_x_at | IGKC |
| 908 | 214836_x_at | IGKV1-5 |
| 908 | 214836_x_at | IGK@ |
| 931 | 215646_s_at | VCAN |
| 934 | 215867_x_at | CA12 |
| 940 | 216401_x_at | NA |
| 969 | 217294_s_at | ENO1 |
| 977 | 217480_x_at | IGKC |
| 1042 | 219612_s_at | FGG |
| 1054 | 219850_s_at | EHF |
| 1093 | 221728_x_at | XIST |
| 1096 | 221731_x_at | VCAN |

| Ovarian cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1120 | 222835_at | THSD4 |
| 1134 | 223307_at | CDCA3 |
| 1145 | 223565_at | MZB1 |
| 1161 | 223940_x_at | MALAT1 |
| 1174 | 224559_at | MALAT1 |
| 1177 | 224567_x_at | MALAT1 |
| 1179 | 224588_at | XIST |
| 1180 | 224589_at | XIST |
| 1181 | 224590_at | XIST |
| 1210 | 225664_at | COL12A1 |
| 1230 | 226197_at | AR |
| 1290 | 227550_at | GFRA1 |
| 1332 | 228582_x_at | MALAT1 |
| 1368 | 229218_at | COL1A2 |
| 1392 | 229975_at | BMPR1B |
| 1403 | 230291_s_at | NFIB |
| 1413 | 230585_at | NA |
| 1414 | 230673_at | PKHD1L1 |
| 1421 | 230865_at | LIX1 |
| 1427 | 231181_at | NA |
| 1439 | 231879_at | COL12A1 |
| 1453 | 232361_s_at | EHF |
| 1459 | 232578_at | CLDN18 |
| 1462 | 232855_at | NA |
| 1464 | 232944_at | NA |
| 1471 | 233388_at | NA |
| 1504 | 235904_at | UGT3A1 |
| 1512 | 236163_at | LIX1 |
| 1523 | 236773_at | NA |
| 1530 | 237625_s_at | IGKC |
| 1537 | 238103_at | LOC100505989 |
| 1551 | 239010_at | DUXAP10 |

| Ovarian cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1569 | 240253_at | NA |
| 1572 | 240331_at | NA |
| 1591 | 242468_at | NA |
| 1593 | 242546_at | FLJ39632 |
| 1594 | 242579_at | BMPR1B |
| 1604 | 243929_at | NA |

Table A.5: Correspondence of genes numbers, affymetrix names and symbols for the **ovarian** cancer dataset.

| Medullary breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1 | 1405_i_at | CCL5 |
| 5 | 1552507_at | KCNE4 |
| 6 | 1552508_at | KCNE4 |
| 42 | 1555730_a_at | CFL1 |
| 78 | 1564479_a_at | NA |
| 85 | 1567628_at | CD74 |
| 118 | 201438_at | COL6A3 |
| 130 | 201744_s_at | LUM |
| 135 | 201852_x_at | COL3A1 |
| 141 | 201909_at | RPS4Y1 |
| 144 | 201971_s_at | ATP6V1A |
| 169 | 202403_s_at | COL1A2 |
| 170 | 202404_s_at | COL1A2 |
| 192 | 202768_at | FOSB |
| 208 | 203065_s_at | CAV1 |
| 214 | 203153_at | IFIT1 |
| 220 | 203324_s_at | CAV2 |
| 278 | 204114_at | NID2 |

| Medullary breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 287 | 204272_at | LGALS4 |
| 298 | 204409_s_at | EIF1AY |
| 415 | 205483_s_at | ISG15 |
| 434 | 205625_s_at | CALB1 |
| 436 | 205650_s_at | FGA |
| 445 | 205765_at | CYP3A5 |
| 462 | 205941_s_at | COL10A1 |
| 479 | 206164_at | CLCA2 |
| 481 | 206166_s_at | CLCA2 |
| 502 | 206488_s_at | CD36 |
| 543 | 207663_x_at | GAGE3 |
| 544 | 207739_s_at | GAGE2C |
| 544 | 207739_s_at | GAGE12F |
| 544 | 207739_s_at | GAGE8 |
| 544 | 207739_s_at | GAGE1 |
| 544 | 207739_s_at | GAGE3 |
| 544 | 207739_s_at | GAGE4 |
| 544 | 207739_s_at | GAGE5 |
| 544 | 207739_s_at | GAGE6 |
| 544 | 207739_s_at | GAGE7 |
| 544 | 207739_s_at | GAGE12I |
| 544 | 207739_s_at | GAGE2E |
| 544 | 207739_s_at | GAGE2B |
| 544 | 207739_s_at | GAGE12G |
| 544 | 207739_s_at | GAGE12J |
| 544 | 207739_s_at | GAGE2D |
| 544 | 207739_s_at | GAGE2A |
| 558 | 208235_x_at | GAGE7 |
| 558 | 208235_x_at | GAGE12F |
| 558 | 208235_x_at | GAGE5 |
| 558 | 208235_x_at | GAGE12I |
| 558 | 208235_x_at | GAGE12G |

| Medullary breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 591 | 209189_at | FOS |
| 624 | 209480_at | HLA-DQB1 |
| 631 | 209555_s_at | CD36 |
| 649 | 209719_x_at | SERPINB3 |
| 674 | 209987_s_at | ASCL1 |
| 680 | 210065_s_at | UPK1B |
| 704 | 210511_s_at | INHBA |
| 715 | 210728_s_at | CALCA |
| 732 | 211161_s_at | COL3A1 |
| 752 | 211644_x_at | IGKC |
| 752 | 211644_x_at | IGK@ |
| 753 | 211645_x_at | NA |
| 754 | 211650_x_at | IGK@ |
| 754 | 211650_x_at | IGHV4-31 |
| 754 | 211650_x_at | IGHV3-23 |
| 754 | 211650_x_at | IGHV1-69 |
| 754 | 211650_x_at | IGHA1 |
| 754 | 211650_x_at | IGHD |
| 754 | 211650_x_at | IGHG1 |
| 754 | 211650_x_at | IGHG3 |
| 754 | 211650_x_at | IGHM |
| 769 | 211796_s_at | IL23A |
| 769 | 211796_s_at | TRBC2 |
| 769 | 211796_s_at | TRBC1 |
| 775 | 211906_s_at | SERPINB4 |
| 779 | 211991_s_at | HLA-DPA1 |
| 797 | 212488_at | COL5A1 |
| 813 | 212950_at | GPR116 |
| 848 | 213768_s_at | ASCL1 |
| 874 | 214135_at | CLDN18 |
| 880 | 214235_at | CYP3A5 |
| 901 | 214657_s_at | NEAT1 |

| Medullary breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 902 | 214669_x_at | IGKC |
| 920 | 215176_x_at | IGKC |
| 920 | 215176_x_at | IGK@ |
| 923 | 215304_at | NA |
| 927 | 215454_x_at | SFTPC |
| 939 | 216238_s_at | FGB |
| 944 | 216557_x_at | LOC100291917 |
| 944 | 216557_x_at | IGHV4-31 |
| 944 | 216557_x_at | IGHV3-48 |
| 944 | 216557_x_at | IGHA1 |
| 944 | 216557_x_at | IGHD |
| 944 | 216557_x_at | IGHG1 |
| 944 | 216557_x_at | IGHG3 |
| 944 | 216557_x_at | IGHM |
| 960 | 217157_x_at | IGKC |
| 960 | 217157_x_at | IGK@ |
| 962 | 217227_x_at | IGLV1-44 |
| 962 | 217227_x_at | IGLV1-40 |
| 966 | 217258_x_at | IGLV1-44 |
| 966 | 217258_x_at | IGLV1-40 |
| 977 | 217480_x_at | IGKC |
| 978 | 217495_x_at | CALCA |
| 1042 | 219612_s_at | FGG |
| 1074 | 220425_x_at | ROPN1 |
| 1074 | 220425_x_at | ROPN1B |
| 1079 | 220624_s_at | ELF5 |
| 1080 | 220625_s_at | ELF5 |
| 1086 | 221423_s_at | YIPF5 |
| 1120 | 222835_at | THSD4 |
| 1157 | 223806_s_at | NAPSA |
| 1175 | 224565_at | NEAT1 |
| 1179 | 224588_at | XIST |

| Medullary breast cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1184 | 224823_at | MYLK |
| 1215 | 225782_at | MSRB3 |
| 1238 | 226311_at | ADAMTS2 |
| 1295 | 227671_at | XIST |
| 1368 | 229218_at | COL1A2 |
| 1376 | 229542_at | C20orf85 |
| 1455 | 232458_at | COL3A1 |
| 1459 | 232578_at | CLDN18 |
| 1468 | 233203_at | ROPN1 |
| 1513 | 236203_at | HLA-DQA1 |
| 1567 | 240161_s_at | CDC20B |
| 1569 | 240253_at | NA |
| 1579 | 241617_x_at | NA |
| 1618 | 40284_at | FOXA2 |
| 1624 | AFFX-HUMGAPDH/M33197_5_at | GAPDH |

Table A.6: Correspondence of genes numbers, affymetrix names and symbols for the **medullary breast** cancer dataset.

| Lung cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 29 | 1554899_s_at | FCER1G |
| 41 | 1555728_a_at | MS4A4A |
| 44 | 1555758_a_at | CDKN3 |
| 51 | 1555854_at | AKR1C1 |
| 51 | 1555854_at | AKR1C2 |
| 63 | 1558048_x_at | NA |
| 66 | 1558678_s_at | MALAT1 |
| 107 | 201149_s_at | TIMP3 |
| 108 | 201150_s_at | TIMP3 |
| 136 | 201858_s_at | SRGN |
| 169 | 202403_s_at | COL1A2 |
| 219 | 203323_at | CAV2 |
| 242 | 203649_s_at | PLA2G2A |
| 248 | 203764_at | DLGAP5 |
| 263 | 203915_at | CXCL9 |
| 271 | 204006_s_at | FCGR3B |
| 271 | 204006_s_at | FCGR3A |
| 304 | 204439_at | IFI44L |
| 315 | 204533_at | CXCL10 |
| 335 | 204688_at | SGCE |
| 370 | 205014_at | FGFBP1 |
| 378 | 205064_at | SPRR1B |
| 391 | 205267_at | POU2AF1 |
| 398 | 205350_at | CRABP1 |
| 412 | 205475_at | SCRG1 |
| 450 | 205825_at | PCSK1 |
| 467 | 205982_x_at | SFTPC |
| 494 | 206378_at | SCGB2A2 |
| 514 | 206641_at | TNFRSF17 |
| 520 | 206799_at | SCGB1D2 |
| 546 | 207802_at | CRISP3 |
| 566 | 208627_s_at | YBX1 |

| Lung cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 587 | 209138_x_at | IGLV3-21 |
| 587 | 209138_x_at | IGL@ |
| 607 | 209351_at | KRT14 |
| 660 | 209810_at | SFTPB |
| 694 | 210338_s_at | HSPA8 |
| 700 | 210432_s_at | SCN3A |
| 702 | 210467_x_at | MAGEA12 |
| 736 | 211421_s_at | RET |
| 753 | 211645_x_at | NA |
| 755 | 211653_x_at | AKR1C2 |
| 765 | 211735_x_at | SFTPC |
| 779 | 211991_s_at | HLA-DPA1 |
| 784 | 212094_at | PEG10 |
| 785 | 212097_at | CAV1 |
| 797 | 212488_at | COL5A1 |
| 803 | 212667_at | SPARC |
| 816 | 212998_x_at | HLA-DQB1 |
| 835 | 213502_x_at | GUSBP11 |
| 848 | 213768_s_at | ASCL1 |
| 851 | 213797_at | RSAD2 |
| 861 | 213936_x_at | SFTPB |
| 872 | 214087_s_at | MYBPC1 |
| 876 | 214183_s_at | TKTL1 |
| 878 | 214218_s_at | XIST |
| 884 | 214354_x_at | SFTPB |
| 886 | 214387_x_at | SFTPC |
| 902 | 214669_x_at | IGKC |
| 903 | 214677_x_at | IGLJ3 |
| 903 | 214677_x_at | CYAT1 |
| 903 | 214677_x_at | IGLV1-44 |
| 924 | 215379_x_at | IGLV3-21 |
| 924 | 215379_x_at | IGLV1-44 |

| Lung cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 924 | 215379_x_at | IGLC7 |
| 936 | 215946_x_at | IGLL3P |
| 940 | 216401_x_at | NA |
| 946 | 216576_x_at | IGKC |
| 946 | 216576_x_at | IGK@ |
| 961 | 217179_x_at | NA |
| 971 | 217378_x_at | NA |
| 977 | 217480_x_at | IGKC |
| 1069 | 220232_at | SCD5 |
| 1070 | 220269_at | ZBBX |
| 1075 | 220445_s_at | CSAG2 |
| 1075 | 220445_s_at | CSAG3 |
| 1081 | 220782_x_at | KLK12 |
| 1094 | 221729_at | COL5A2 |
| 1133 | 223278_at | GJB2 |
| 1145 | 223565_at | MZB1 |
| 1161 | 223940_x_at | MALAT1 |
| 1180 | 224589_at | XIST |
| 1215 | 225782_at | MSRB3 |
| 1254 | 226811_at | FAM46C |
| 1295 | 227671_at | XIST |
| 1340 | 228780_at | NA |
| 1376 | 229542_at | C20orf85 |
| 1385 | 229782_at | RMST |
| 1423 | 231077_at | C1orf192 |
| 1424 | 231084_at | WDR96 |
| 1438 | 231771_at | GJB6 |
| 1457 | 232523_at | MEGF10 |
| 1473 | 233586_s_at | KLK12 |
| 1478 | 234316_x_at | KLK12 |
| 1480 | 234764_x_at | IGLV1-44 |
| 1480 | 234764_x_at | IGLV1-36 |

| Lung cancer dataset | | |
|---|---|---|
| Gene number | Affy ID | Symbol |
| 1488 | 235060_at | LOC100190986 |
| 1510 | 236085_at | CAPSL |
| 1515 | 236256_at | NA |
| 1540 | 238320_at | NEAT1 |
| 1553 | 239150_at | SNTN |
| 1570 | 240303_at | TMC5 |
| 1571 | 240304_s_at | TMC5 |
| 1615 | 38691_s_at | SFTPC |

Table A.7: Correspondence of genes numbers, affymetrix names and symbols for the **lung** cancer dataset.

# References


Abdi, H. (2007), 'The kendall rank correlation coefficient', *Encyclopedia of Measurement and Statistics. Sage, Thousand Oaks, CA* pp. 508–510.

Akaike, H. (1974), 'A new look at the statistical model identification', *Automatic Control, IEEE Transactions on* **19**(6), 716–723.

Alakwaa, F. M., Solouma, N. H. & Kadah, Y. M. (2011), 'Construction of gene regulatory networks using biclustering and bayesian networks', *Theoretical Biology and Medical Modelling* **8**(1), 39.

Altman, D. G. (1990), *Practical statistics for medical research*, CRC Press.

Altshuler, D., Daly, M. & Kruglyak, L. (2000), 'Guilt by association', *Nature genetics* **26**(2), 135–138.

Andersson, U., Heddad, M. & Adamska, I. (2003), 'Light stress-induced one-helix protein of the chlorophyll a/b-binding family associated with photosystem i', *Plant physiology* **132**(2), 811–820.

Ando, K. & Grumet, R. (2010), 'Transcriptional profiling of rapidly growing cucumber fruit by 454-pyrosequencing analysis', *Journal of the American Society for Horticultural Science* **135**(4), 291–302.

Angelopoulos, N. & Wessels, L. (2011), 'Effective priors over model structures applied to dna binding assay data', *Probabilistic Problem Solving in BioMedicine* p. 83.

Anvar, S. Y., AC't Hoen, P. & Tucker, A. (2010), 'The identification of informative genes from multiple datasets with increasing complexity', *BMC bioinformatics* **11**(1), 32.

*ArrayExpress* (2014).

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T. et al. (2000), 'Gene ontology: tool for the unification of biology', *Nature genetics* **25**(1), 25–29.

Baalmann, E., Scheibe, R., Cerff, R. & Martin, W. (1996), 'Functional studies of chloroplast glyceraldehyde-3-phosphate dehydrogenase subunits a and b expressed in escherichia coli: formation of highly active a4 and b4 homotetramers and evidence that aggregation of the b4 complex is mediated by the b subunit carboxy terminus', *Plant molecular biology* **32**(3), 505–513.

Baek, D., Jin, Y., Jeong, J. C., Lee, H.-J., Moon, H., Lee, J., Shin, D., Kang, C. H., Kim, D. H., Nam, J. et al. (2008), 'Suppression of reactive oxygen species by glyceraldehyde-3-phosphate dehydrogenase', *Phytochemistry* **69**(2), 333–338.

Baldi, P. & Brunak, S. (2001), *Bioinformatics: the machine learning approach*, MIT press.

Baldi, P. & Long, A. D. (2001), 'A bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes', *Bioinformatics* **17**(6), 509–519.

Banerjee, O., El Ghaoui, L. & d'Aspremont, A. (2008), 'Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data', *The Journal of Machine Learning Research* **9**, 485–516.

Bang, W. Y., Jeong, I. S., Kim, D. W., Im, C. H., Ji, C., Hwang, S. M., Kim, S. W., Son, Y. S., Jeong, J., Shiina, T. et al. (2008), 'Role of arabidopsis chl27 protein for photosynthesis, chloroplast development and gene expression profiling', *Plant and cell physiology* **49**(9), 1350–1363.

Barabási, A.-L. & Albert, R. (1999), 'Emergence of scaling in random networks', *science* **286**(5439), 509–512.

Barabasi, A.-L. & Oltvai, Z. N. (2004), 'Network biology: understanding the cell's functional organization', *Nature Reviews Genetics* **5**(2), 101–113.

Beinlich, I. A., Suermondt, H. J., Chavez, R. M. & Cooper, G. F. (1989), *The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks*, Springer.

Bellman, R., Bellman, R. E., Bellman, R. E. & Bellman, R. E. (1961), *Adaptive control processes: a guided tour*, Vol. 4, Princeton University Press Princeton.

Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M. & Yakhini, Z. (2000), 'Tissue classification with gene expression profiles', *Journal of Computational Biology* **7**(3-4), 559–583.

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

Binder, J., Koller, D., Russell, S. & Kanazawa, K. (1997), 'Adaptive probabilistic networks with hidden variables', *Machine Learning* **29**(2-3), 213–244.

Bo, V., Lysenko, A., Saqi, M., Habash, D. & Tucker, A. (2013), Integrating multiple studies of wheat microarray data to identify treatment-specific regulatory networks, *in* 'Advances in Intelligent Data Analysis XII', Springer, pp. 104–115.

Bouckaert, R. R. (1995), Bayesian belief networks: from inference to construction, PhD thesis, PhD Thesis, Faculteit Winskunde en Informatica, Utrech Universiteit.

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

Butte, A. J. & Kohane, I. S. (2000), Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements, *in* 'Pac Symp Biocomput', Vol. 5, pp. 418–429.

Castillo, E. (1997), *Expert systems and probabilistic network models*, Springer.

Causton, H., Quackenbush, J. & Brazma, A. (2009), *Microarray gene expression data analysis: a beginner's guide*, John Wiley & Sons.

Cheng, Y. & Church, G. M. (2000), Biclustering of expression data, *in* 'Proceedings of the eighth international conference on intelligent systems for molecular biology', Vol. 8, pp. 93–103.

Choi, J. K., Yu, U., Kim, S. & Yoo, O. J. (2003), 'Combining multiple microarray studies and modeling interstudy variation', *Bioinformatics* **19**(suppl 1), i84–i90.

Cleveland, W. S. (1979), 'Robust locally weighted regression and smoothing scatterplots', *Journal of the American statistical association* **74**(368), 829–836.

Damkjær, J. T., Kereïche, S., Johnson, M. P., Kovacs, L., Kiss, A. Z., Boekema, E. J., Ruban, A. V., Horton, P. & Jansson, S. (2009), 'The photosystem ii light-harvesting protein lhcb3

affects the macrostructure of photosystem ii and the rate of state transitions in arabidopsis', *The Plant Cell Online* **21**(10), 3245–3256.

Dash, S., Van Hemert, J., Hong, L., Wise, R. P. & Dickerson, J. A. (2012), 'Plexdb: gene expression resources for plants and plant pathogens', *Nucleic acids research* **40**(D1), D1194–D1201.

Dempster, A. P. (1972), 'Covariance selection', *Biometrics* pp. 157–175.

DeRisi, J. L., Iyer, V. R. & Brown, P. O. (1997), 'Exploring the metabolic and genetic control of gene expression on a genomic scale', *Science* **278**(5338), 680–686.

Deshpande, R., Sharma, S., Verfaillie, C. M., Hu, W.-S. & Myers, C. L. (2010), 'A scalable approach for discovering conserved active subnetworks across species', *PLoS computational biology* **6**(12), e1001028.

Dudoit, S., Fridlyand, J. & Speed, T. P. (2002), 'Comparison of discrimination methods for the classification of tumors using gene expression data', *Journal of the American statistical association* **97**(457), 77–87.

Dudoit, S., Shaffer, J. P. & Boldrick, J. C. (2003), 'Multiple hypothesis testing in microarray experiments', *Statistical Science* pp. 71–103.

Edgar, R., Domrachev, M. & Lash, A. E. (2002), 'Gene expression omnibus: Ncbi gene expression and hybridization array data repository', *Nucleic acids research* **30**(1), 207–210.

Edwards, D. (2000), *Introduction to graphical modelling*, Springer.

Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. (1998), 'Cluster analysis and display of genome-wide expression patterns', *Proceedings of the National Academy of Sciences* **95**(25), 14863–14868.

Erdős, P. & Rényi, A. (1959), 'On random graphs i.', *Publ. Math. Debrecen* **6**, 290–297.

Fawcett, T. (2006), 'An introduction to roc analysis', *Pattern recognition letters* **27**(8), 861–874.

Filkov, V. (2005), 'Identifying gene regulatory networks from gene expression data', *Handbook of Computational Molecular Biology* pp. 27–1.

Fox, R. J. & Dimmic, M. W. (2006), 'A two-sample bayesian t-test for microarray data', *BMC bioinformatics* **7**(1), 126.

Friedman, J., Hastie, T., Höfling, H., Tibshirani, R. et al. (2007), 'Pathwise coordinate optimization', *The Annals of Applied Statistics* **1**(2), 302–332.

Friedman, J., Hastie, T. & Tibshirani, R. (2008), 'Sparse inverse covariance estimation with the graphical lasso', *Biostatistics* **9**(3), 432–441.

Friedman, J., Hastie, T. & Tibshirani, R. (2014), *glasso: Graphical lasso- estimation of Gaussian graphical models*. R package version 1.8.
**URL:** *http://CRAN.R-project.org/package=glasso*

Friedman, N., Linial, M., Nachman, I. & Pe'er, D. (2000), 'Using bayesian networks to analyze expression data', *Journal of computational biology* **7**(3-4), 601–620.

Friedman, N., Nachman, I. & Peér, D. (1999), Learning bayesian network structure from massive datasets: the sparse candidate algorithm, *in* 'Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence', Morgan Kaufmann Publishers Inc., pp. 206–215.

Gao, S. & Wang, X. (2011), 'Quantitative utilization of prior biological knowledge in the bayesian network modeling of gene expression data', *BMC bioinformatics* **12**(1), 359.

Gautier, L., Cope, L., Bolstad, B. M. & Irizarry, R. A. (2004), 'affy—analysis of affymetrix genechip data at the probe level', *Bioinformatics* **20**(3), 307–315.

Geer, L. Y., Marchler-Bauer, A., Geer, R. C., Han, L., He, J., He, S., Liu, C., Shi, W. & Bryant, S. H. (2009), 'The ncbi biosystems database', *Nucleic acids research* p. gkp858.

Gentleman, R. C., Carey, V. J., Bates, D. M. & others (2004), 'Bioconductor: Open software development for computational biology and bioinformatics', *Genome Biology* **5**, R80.
**URL:** *http://genomebiology.com/2004/5/10/R80*

Gillis, J. & Pavlidis, P. (2012), 'guilt by association is the exception rather than the rule in gene networks', *PLoS computational biology* **8**(3), e1002444.

Glymour, C., Scheines, R. & Spirtes, P. (2001), *Causation, prediction, and search*, MIT Press.

Goodman, L. A. & Kruskal, W. H. (1954), 'Measures of association for cross classifications*', *Journal of the American Statistical Association* **49**(268), 732–764.

Greenshtein, E., Ritov, Y. et al. (2004), 'Persistence in high-dimensional linear predictor selection and the virtue of overparametrization', *Bernoulli* **10**(6), 971–988.

Grigorova, B., Vaseva, I. I., Demirevska, K. & Feller, U. (2011), 'Expression of selected heat shock proteins after individually applied and combined drought and heat stress', *Acta Physiologiae Plantarum* **33**(5), 2041–2049.

Grigoryev, Y. (2011), 'Grigoryev 2011,introduction to dna microarrays'. [Online; accessed 11-11-2014].
**URL:** *http://bitesizebio.com/7206/introduction-to-dna-microarrays/*

Gyftodimos, E. & Flach, P. A. (2002), Hierarchical bayesian networks: A probabilistic reasoning model for structured domains, *in* 'Proceedings of the ICML-2002 Workshop on Development of Representations. University of New South Wales', pp. 23–30.

Hanahan, D. & Weinberg, R. A. (2011), 'Hallmarks of cancer: the next generation', *Cell* **144**(5), 646–674.

Hänzelmann, S., Castelo, R. & Guinney, J. (2013), 'Gsva: gene set variation analysis for microarray and rna-seq data', *BMC bioinformatics* **14**(1), 7.

Hartemink, A. J., Gifford, D. K., Jaakkola, T. S. & Young, R. A. (2002), 'Bayesian methods for elucidating genetic regulatory networks', *IEEE Intelligent Systems* **17**(2), 37–43.

Hartigan, J. A. (1975), 'Clustering algorithms'.

Hartigan, J. A. & Wong, M. A. (1979), 'Algorithm as 136: A k-means clustering algorithm', *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **28**(1), 100–108.

Hartl, D. L. & Jones, E. W. (2009), *Genetics: Analysis of Genes and Genomes*, Jones and Bartlett Pub.
**URL:** *http://books.google.co.uk/books?id=cfvILxY9tCIC*

Heckerman, D., Geiger, D. & Chickering, D. M. (1995), 'Learning bayesian networks: The combination of knowledge and statistical data', *Machine learning* **20**(3), 197–243.

Henrion, M. (1988), Propagation of uncertainty by probabilistic logic sampling in bayes networks, *in* 'Uncertainty in Artificial Intelligence', Vol. 2, pp. 149–164.

Højsgaard, S. (2012), 'Graphical independence networks with the grain package for r'.

Horvath, S. (2005), 'An overview of weighted gene co-expression network analysis', `www.genetics.ucla.edu/courses/hg236b/Horvath_Presentation_GN.pdf`. [Online; accessed 10-02-2014].

Horvath, S. (2011), *Weighted Network Analysis: Applications in Genomics and Systems Biology*, Springer.

Hu, H., Yan, X., Huang, Y., Han, J. & Zhou, X. J. (2005), 'Mining coherent dense subgraphs across massive biological networks for functional discovery', *Bioinformatics* **21**(suppl 1), i213–i221.

Ideker, T., Ozier, O., Schwikowski, B. & Siegel, A. F. (2002), 'Discovering regulatory and signalling circuits in molecular interaction networks', *Bioinformatics* **18**(suppl 1), S233–S240.

Ihalainen, J. A., Jensen, P. E., Haldrup, A., van Stokkum, I. H., van Grondelle, R., Scheller, H. V. & Dekker, J. P. (2002), 'Pigment organization and energy transfer dynamics in isolated photosystem i (psi) complexes from arabidopsis thaliana depleted of the psi-g, psi-k, psi-l, or psi-n subunit', *Biophysical journal* **83**(4), 2190–2201.

Infanger, S., Bischof, S., Hiltbrunner, A., Agne, B., Baginsky, S. & Kessler, F. (2011), 'The chloroplast import receptor toc90 partially restores the accumulation of toc159 client proteins in the arabidopsis thaliana ppi2 mutant', *Molecular plant* **4**(2), 252–263.

Irizarry, R. A., Bolstad, B. M., Collin, F., Cope, L. M., Hobbs, B. & Speed, T. P. (2003), 'Summaries of affymetrix genechip probe level data', *Nucleic acids research* **31**(4), e15–e15.

Isella, C., Renzulli, T., Cora, D. & Medico, E. (2011), 'Mulcom: a multiple comparison statistical test for microarray data in bioconductor', *BMC bioinformatics* **12**(1), 382.

Jafari, P. & Azuaje, F. (2006), 'An assessment of recently published gene expression data analyses: reporting experimental design and statistical factors', *BMC Medical Informatics and Decision Making* **6**(1), 27.

Jansen, R., Yu, H., Greenbaum, D., Kluger, Y., Krogan, N. J., Chung, S., Emili, A., Snyder, M., Greenblatt, J. F. & Gerstein, M. (2003), 'A bayesian networks approach for predicting protein-protein interactions from genomic data', *Science* **302**(5644), 449–453.

Jensen, F. V. (1996), *An introduction to Bayesian networks*, Vol. 210, UCL press London.

Jensen, P. E., Gilpin, M., Knoetzel, J. & Scheller, H. V. (2000), 'The psi-k subunit of photosystem i is involved in the interaction between light-harvesting complex i and the photosystem i reaction center core', *Journal of Biological Chemistry* **275**(32), 24701–24708.

Kaiser, S., Santamaria, R., Theron, R., Quintales, L. & Leisch, F. (2009), 'biclust: Bicluster algorithms', *R package version 0.7* **2**.

Kanehisa, M. et al. (2000), *Post-genome informatics*, Oxford University Press (OUP).

Karlebach, G. & Shamir, R. (2008), 'Modelling and analysis of gene regulatory networks', *Nature Reviews Molecular Cell Biology* **9**(10), 770–780.

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

Kauffman, S. A. (1993), *The origins of order: Self-organization and selection in evolution*, Oxford university press.

Khanin, R. & Wit, E. (2006), 'How scale-free are biological networks', *Journal of computational biology* **13**(3), 810–818.

Kim, Y.-J., Lee, J.-H., Lee, O. R., Shim, J.-S., Jung, S.-K., Son, N.-R., Kim, J.-H., Kim, S.-Y. & Yang, D.-C. (2010), 'Isolation and characterization of a type ii peroxiredoxin gene from panax ginseng ca meyer', *J Ginseng Res* **34**, 296–303.

Kinney, J. B. & Atwal, G. S. (2014), 'Equitability, mutual information, and the maximal information coefficient', *Proceedings of the National Academy of Sciences* **111**(9), 3354–3359.

Kirk, P., Griffin, J. E., Savage, R. S., Ghahramani, Z. & Wild, D. L. (2012), 'Bayesian correlated clustering to integrate multiple datasets', *Bioinformatics* **28**(24), 3290–3297.

Kluger, Y., Basri, R., Chang, J. T. & Gerstein, M. (2003), 'Spectral biclustering of microarray data: coclustering genes and conditions', *Genome research* **13**(4), 703–716.

Koller, D. & Friedman, N. (2009), *Probabilistic graphical models: principles and techniques*, MIT press.

Korb, K. B. & Nicholson, A. E. (2003), *Bayesian artificial intelligence*, cRc Press.

Kwon, Y., Kim, J. H., Nguyen, H. N., Jikumaru, Y., Kamiya, Y., Hong, S.-W. & Lee, H. (2013), 'A novel arabidopsis myb-like transcription factor, mybh, regulates hypocotyl elongation by enhancing auxin accumulation', *Journal of experimental botany* **64**(12), 3911–3922.

Lamesch, P., Berardini, T. Z., Li, D., Swarbreck, D., Wilks, C., Sasidharan, R., Muller, R., Dreher, K., Alexander, D. L., Garcia-Hernandez, M. et al. (2012), 'The arabidopsis information resource (tair): improved gene annotation and new tools', *Nucleic acids research* **40**(D1), D1202–D1210.

Langfelder, P. & Horvath, S. (2008), 'Wgcna: an r package for weighted correlation network analysis', *BMC bioinformatics* **9**(1), 559.

Langfelder, P. & Horvath, S. (2012), 'Fast r functions for robust correlations and hierarchical clustering', *Journal of statistical software* **46**(11).

Larranaga, P., Sierra, B., Gallego, M. J., Michelena, M. J. & Picaza, J. M. (1997), Learning bayesian networks by genetic algorithms: a case study in the prediction of survival in malignant skin melanoma, *in* 'Artificial Intelligence in Medicine', Springer, pp. 261–272.

Lauritzen, S. L. (1996), *Graphical models*, Oxford University Press.

Lee, I., Date, S. V., Adai, A. T. & Marcotte, E. M. (2004), 'A probabilistic functional network of yeast genes', *science* **306**(5701), 1555–1558.

Li, Y. & Patra, J. C. (2010), 'Integration of multiple data sources to prioritize candidate genes using discounted rating system', *BMC bioinformatics* **11**(Suppl 1), S20.

Llorente, F., López-Cobollo, R. M., Catalá, R., Martínez-Zapater, J. M. & Salinas, J. (2002), 'A novel cold-inducible gene from arabidopsis, rci3, encodes a peroxidase that constitutes a component for stress tolerance', *The Plant Journal* **32**(1), 13–24.

Lu, J. & Bushel, P. R. (2010), *pvac: PCA-based gene filtering for Affymetrix arrays*. R package version 1.12.0.

Lysenko, A., Defoin-Platel, M., Hassani-Pak, K., Taubert, J., Hodgman, C., Rawlings, C. J. & Saqi, M. (2011), 'Assessing the functional coherence of modules found in multiple-evidence networks from arabidopsis', *BMC bioinformatics* **12**(1), 203.

Madeira, S. C. & Oliveira, A. L. (2004), 'Biclustering algorithms for biological data analysis: a survey', *Computational Biology and Bioinformatics, IEEE/ACM Transactions on* **1**(1), 24–45.

Magrane, M., Consortium, U. et al. (2011), 'Uniprot knowledgebase: a hub of integrated protein data', *Database* **2011**, bar009.

Margaritis, D. (2003), Learning Bayesian network model structure from data, PhD thesis, University of Pittsburgh.

Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D. & Califano, A. (2006), 'Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context', *BMC bioinformatics* **7**(Suppl 1), S7.

Margulies, M., Egholm, M., Altman, W. E., Attiya, S., Bader, J. S., Bemben, L. A., Berka, J., Braverman, M. S., Chen, Y.-J., Chen, Z. et al. (2005), 'Genome sequencing in microfabricated high-density picolitre reactors', *Nature* **437**(7057), 376–380.

Marri, L., Sparla, F., Pupillo, P. & Trost, P. (2005), 'Co-ordinated gene expression of photosynthetic glyceraldehyde-3-phosphate dehydrogenase, phosphoribulokinase, and cp12 in arabidopsis thaliana', *Journal of experimental botany* **56**(409), 73–80.

Marri, L., Trost, P., Trivelli, X., Gonnelli, L., Pupillo, P. & Sparla, F. (2008), 'Spontaneous assembly of photosynthetic supramolecular complexes as mediated by the intrinsically unstructured protein cp12', *Journal of biological chemistry* **283**(4), 1831–1838.

Meinshausen, N. & Bühlmann, P. (2006), 'High-dimensional graphs and variable selection with the lasso', *The Annals of Statistics* pp. 1436–1462.

Mochizuki, N., Brusslan, J. A., Larkin, R., Nagatani, A. & Chory, J. (2001), 'Arabidopsis genomes uncoupled 5 (gun5) mutant reveals the involvement of mg-chelatase h subunit in plastid-to-nucleus signal transduction', *Proceedings of the National Academy of sciences* **98**(4), 2053–2058.

Moreau, Y. & Tranchevent, L.-C. (2012), 'Computational tools for prioritizing candidate genes: boosting disease gene discovery', *Nature Reviews Genetics* **13**(8), 523–536.

Morgat, A., Coissac, E., Coudert, E., Axelsen, K. B., Keller, G., Bairoch, A., Bridge, A., Bougueleret, L., Xenarios, I. & Viari, A. (2011), 'Unipathway: a resource for the exploration and annotation of metabolic pathways', *Nucleic acids research* p. gkr1023.

Morozova, O. & Marra, M. A. (2008), 'Applications of nextgeneration sequencing technologies in functional genomics', *Genomics* **92**(5), 255–264.

Morrow, J., Qiu, W., He, W., Wang, X. & Lazarus, R. (2012), *GeneSelectMMD: Gene selection based on the marginal distributions of gene profiles that characterized by a mixture of three-component multivariate distributions*. R package version 2.2.0.

Mueller, L. A., Kugler, K. G., Graber, A., Emmert-Streib, F. & Dehmer, M. (2011), 'Structural measures for network biology using quacn', *BMC bioinformatics* **12**(1), 492.

Murali, T. & Kasif, S. (2003), Extracting conserved gene expression motifs from gene expression data, *in* 'Pacific Symposium on Biocomputing', Vol. 8, pp. 77–88.

Murphy, K. (2001), 'An introduction to graphical models', *A Brief Introduction to Graphical Models and Bayesian Networks* **10**.

Nagarajan, R., Scutari, M. & Lèbre, S. (2013), *Bayesian Networks in R*, Springer.

*NCBI Microarray* (2014). [Online; accessed 17-07-2014].
**URL:** *http://www.ncbi.nlm.nih.gov/projects/genome/probe/doc/TechMicroarray.shtml*

Newton, M. A., Kendziorski, C. M., Richmond, C. S., Blattner, F. R. & Tsui, K.-W. (2001), 'On differential variability of expression ratios: improving statistical inference about gene expression changes from microarray data', *Journal of computational biology* **8**(1), 37–52.

Nielsen, T. D. & Jensen, F. V. (2009), *Bayesian networks and decision graphs*, Springer.

Nykter, M., Aho, T., Ahdesmäki, M., Ruusuvuori, P., Lehmussola, A. & Yli-Harja, O. (2006), 'Simulation of microarray data with realistic characteristics', *BMC bioinformatics* **7**(1), 349.

Obayashi, T., Nishida, K., Kasahara, K. & Kinoshita, K. (2011), 'Atted-ii updates: condition-specific gene coexpression to extend coexpression analyses and applications to a broad range of flowering plants', *Plant and cell physiology* **52**(2), 213–219.

Oliver, S. (2000), 'Proteomics: guilt-by-association goes global', *Nature* **403**(6770), 601–603.

Parkinson, H., Kapushesky, M., Shojatalab, M., Abeygunawardena, N., Coulson, R., Farne, A., Holloway, E., Kolesnykov, N., Lilja, P., Lukk, M. et al. (2007), 'Arrayexpressa public database of microarray experiments and gene expression profiles', *Nucleic acids research* **35**(suppl 1), D747–D750.

Pearl, J. (1988), *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann.

Pearl, J., Verma, T. et al. (1991), *A theory of inferred causation*, Morgan Kaufmann San Mateo, CA.

Pearl, J. et al. (2009), 'Causal inference in statistics: An overview', *Statistics Surveys* **3**, 96–146.

Pearson, K. (1901), 'Liii. on lines and planes of closest fit to systems of points in space', *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* **2**(11), 559–572.

Peer, D., Regev, A., Elidan, G. & Friedman, N. (2001), 'Inferring subnetworks from perturbed expression profiles', *Bioinformatics* **17**(suppl 1), S215–S224.

Pe'er, D., Tanay, A. & Regev, A. (2006), 'Minreg: A scalable algorithm for learning parsimonious regulatory networks in yeast and mammals', *The Journal of Machine Learning Research* **7**, 167–189.

Pérez-Rodríguez, P., Riano-Pachon, D. M., Corrêa, L. G. G., Rensing, S. A., Kersten, B. & Mueller-Roeber, B. (2009), 'Plntfdb: updated content and new features of the plant transcription factor database', *Nucleic acids research* p. gkp805.

Pirie, W. (1988), 'Spearman rank correlation coefficient', *Encyclopedia of statistical sciences* .

Quackenbush, J. (2001), 'Computational analysis of microarray data', *Nature Reviews Genetics* **2**(6), 418–427.

Rappaport, N., Nativ, N., Stelzer, G., Twik, M., Guan-Golan, Y., Stein, T. I., Bahir, I., Belinky, F., Morrey, C. P., Safran, M. et al. (2013), 'Malacards: an integrated compendium for diseases and their annotation', *Database* **2013**, bat018.

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N. & Barabási, A.-L. (2002), 'Hierarchical organization of modularity in metabolic networks', *science* **297**(5586), 1551–1555.

Reiner, A., Yekutieli, D. & Benjamini, Y. (2003), 'Identifying differentially expressed genes using false discovery rate controlling procedures', *Bioinformatics* **19**(3), 368–375.

RStudio & Inc. (2014), *shiny: Web Application Framework for R*. R package version 0.9.1.

Rudy, J. & Valafar, F. (2011), 'Empirical comparison of cross-platform normalization methods for gene expression data', *BMC bioinformatics* **12**(1), 467.

Russell, S. (2003), *Artificial intelligence: A modern approach, 2/E*, Pearson Education India.

Rustici, G., Kolesnikov, N., Brandizi, M., Burdett, T., Dylag, M., Emam, I., Farne, A., Hastings, E., Ison, J., Keays, M. et al. (2013), 'Arrayexpress updatetrends in database growth and links to data analysis tools', *Nucleic acids research* **41**(D1), D987–D990.

Sachs, K., Itani, S., Carlisle, J., Nolan, G. P., Pe'er, D. & Lauffenburger, D. A. (2009), 'Learning signaling network structures with sparsely distributed data', *Journal of Computational Biology* **16**(2), 201–212.

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A. & Nolan, G. P. (2005), 'Causal protein-signaling networks derived from multiparameter single-cell data', *Science* **308**(5721), 523–529.

Saeys, Y., Inza, I. & Larrañaga, P. (2007), 'A review of feature selection techniques in bioinformatics', *bioinformatics* **23**(19), 2507–2517.

Safran, M., Dalah, I., Alexander, J., Rosen, N., Stein, T. I., Shmoish, M., Nativ, N., Bahir, I., Doniger, T., Krug, H. et al. (2010), 'Genecards version 3: the human gene integrator', *Database* **2010**, baq020.

Samac, D. A., Hironaka, C. M., Yallaly, P. E. & Shah, D. M. (1990), 'Isolation and characterization of the genes encoding basic and acidic chitinase in arabidopsis thaliana', *Plant Physiology* **93**(3), 907–914.

Savage, R. S., Ghahramani, Z., Griffin, J. E., Bernard, J. & Wild, D. L. (2010), 'Discovering transcriptional modules by bayesian data integration', *Bioinformatics* **26**(12), i158–i167.

Schuster, S. C. (2007), 'Next-generation sequencing transforms todays biology', *Nature* **200**(8).

Schwarz, G. et al. (1978), 'Estimating the dimension of a model', *The annals of statistics* **6**(2), 461–464.

*Scitable* (2014). [Online; accessed 11-06-2014].
**URL:** *www.nature.com/scitable*

Scutari, M. (2009), 'Learning bayesian networks with the bnlearn r package', *arXiv preprint arXiv:0908.3817* .

Scutari, M. (2014), 'Bayesian network repository', `www.bnlearn.com/bnrepository/`.

Scutari, M. & Nagarajan, R. (2011), On identifying significant edges in graphical models, *in* 'Proceedings of workshop on probabilistic problem solving in biomedicine. Bled, Slovenia: Springer-Verlag', pp. 15–27.

Segal, E., Pe'er, D., Regev, A. & Koller, D. (2005), 'Learning module networks', *Journal of Machine Learning Research* **6**(4), 557–588.

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D. & Friedman, N. (2003), 'Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data', *Nature genetics* **34**(2), 166–176.

Sengupta, U., Ukil, S., Dimitrova, N. & Agrawal, S. (2009), 'Expression-based network biology identifies alteration in key regulatory pathways of type 2 diabetes and associated risk/complications', *PloS one* **4**(12), e8100.

Shen, R., Mo, Q., Schultz, N., Seshan, V. E., Olshen, A. B., Huse, J., Ladanyi, M. & Sander, C. (2012), 'Integrative subtype discovery in glioblastoma using icluster', *PloS one* **7**(4), e35236.

Shen, R., Olshen, A. B. & Ladanyi, M. (2009), 'Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis', *Bioinformatics* **25**(22), 2906–2912.

Shendure, J. & Ji, H. (2008), 'Next-generation dna sequencing', *Nature biotechnology* **26**(10), 1135–1145.

Shendure, J., Porreca, G. J., Reppas, N. B., Lin, X., McCutcheon, J. P., Rosenbaum, A. M., Wang, M. D., Zhang, K., Mitra, R. D. & Church, G. M. (2005), 'Accurate multiplex polony sequencing of an evolved bacterial genome', *Science* **309**(5741), 1728–1732.

Shi, L., Shi, L., Reid, L. H., Jones, W. D., Shippy, R., Warrington, J. A., Baker, S. C., Collins, P. J., de Longueville, F., Kawasaki, E. S. et al. (2006), 'The microarray quality control (maqc) project shows inter-and intraplatform reproducibility of gene expression measurements', *Nature biotechnology* **24**(9), 1151–1161.

Shojaie, A. & Michailidis, G. (2010), 'Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs', *Biometrika* **97**(3), 519–538.

Slawski, M. & Boulesteix, A.-L. (2009), *GeneSelector: Stability and Aggregation of ranked gene lists*. R package version 2.8.0.

Somorjai, R. L., Dolenko, B. & Baumgartner, R. (2003), 'Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions', *Bioinformatics* **19**(12), 1484–1491.

Spiegelhalter, D. J. & Cowell, R. G. (1992), 'Learning in probabilistic expert systems', *Bayesian statistics* **4**, 447–465.

Srinivasan, R. & Oliver, D. J. (1995), 'Light-dependent and tissue-specific expression of the h-protein of the glycine decarboxylase complex', *Plant physiology* **109**(1), 161–168.

Steele, E. (2010), 'Combining heterogeneous sources of data for the reverse-engineering of gene regulatory networks', *School of Information Systems, Computing and Mathematics* .

Steele, E. & Tucker, A. (2008), 'Consensus and meta-analysis regulatory networks for combining multiple microarray gene expression datasets', *Journal of biomedical informatics* **41**(6), 914–926.

Steele, E., Tucker, A., Schuemie, M. et al. (2009), 'Literature-based priors for gene regulatory networks', *Bioinformatics* **25**(14), 1768–1774.

Steiner (2014), 'Biology with steiner'. [Online; accessed 11-11-2014].
**URL:** *biologywithsteiner.weebly.com*

Stuart, J. M., Segal, E., Koller, D. & Kim, S. K. (2003), 'A gene-coexpression network for global discovery of conserved genetic modules', *science* **302**(5643), 249–255.

Stumpf, M. P., Wiuf, C. & May, R. M. (2005), 'Subnets of scale-free networks are not scale-free: sampling properties of networks', *Proceedings of the National Academy of Sciences of the United States of America* **102**(12), 4221–4224.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S. et al. (2005), 'Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles', *Proceedings of the National Academy of Sciences of the United States of America* **102**(43), 15545–15550.

Sung, J., Kim, P.-J., Ma, S., Funk, C. C., Magis, A. T., Wang, Y., Hood, L., Geman, D. & Price, N. D. (2013), 'Multi-study integration of brain cancer transcriptomes reveals organ-level molecular signatures', *PLoS computational biology* **9**(7), e1003148.

Swift, S., Tucker, A., Vinciotti, V., Martin, N., Orengo, C., Liu, X. & Kellam, P. (2004), 'Consensus clustering and functional interpretation of gene-expression data', *Genome biology* **5**(11), R94.

Tabus, I. & Astola, J. (2005), 'Gene feature selection', *Genomic Signal Processing and Statistics* pp. 67–92.

Tan, P. K., Downey, T. J., Spitznagel Jr, E. L., Xu, P., Fu, D., Dimitrov, D. S., Lempicki, R. A., Raaka, B. M. & Cam, M. C. (2003), 'Evaluation of gene expression measurements from commercial microarray platforms', *Nucleic acids research* **31**(19), 5676–5684.

Taylor, I. W., Linding, R., Warde-Farley, D., Liu, Y., Pesquita, C., Faria, D., Bull, S., Pawson, T., Morris, Q. & Wrana, J. L. (2009), 'Dynamic modularity in protein interaction networks predicts breast cancer outcome', *Nature biotechnology* **27**(2), 199–204.

Thimm, O., Bläsing, O., Gibon, Y., Nagel, A., Meyer, S., Krüger, P., Selbig, J., Müller, L. A., Rhee, S. Y. & Stitt, M. (2004), 'mapman: a user-driven tool to display genomics data sets onto diagrams of metabolic pathways and other biological processes', *The Plant Journal* **37**(6), 914–939.

Thomas, J. G., Olson, J. M., Tapscott, S. J. & Zhao, L. P. (2001), 'An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles', *Genome Research* **11**(7), 1227–1236.

Tibshirani, R. (1996), 'Regression shrinkage and selection via the lasso', *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 267–288.

Tsamardinos, I., Aliferis, C. F., Statnikov, A. R. & Statnikov, E. (2003), Algorithms for large scale markov blanket discovery, *in* 'FLAIRS Conference', Vol. 2003, pp. 376–381.

Tsamardinos, I., Brown, L. E. & Aliferis, C. F. (2006), 'The max-min hill-climbing bayesian network structure learning algorithm', *Machine learning* **65**(1), 31–78.

Uusitalo, L. (2007), 'Advantages and challenges of bayesian networks in environmental modelling', *Ecological modelling* **203**(3), 312–318.

van Bakel, H. & Holstege, F. C. (2007), 'A tutorial for dna microarray expression profiling'.

Varshavsky, R., Gottlieb, A., Linial, M. & Horn, D. (2006), 'Novel unsupervised feature filtering of biological data', *Bioinformatics* **22**(14), e507–e513.

Wang, A. & Gehan, E. A. (2005), 'Gene selection for microarray data analysis using principal component analysis', *Statistics in medicine* **24**(13), 2069–2087.

Watts, D. J. & Strogatz, S. H. (1998), 'Collective dynamics of small-worldnetworks', *nature* **393**(6684), 440–442.

Weiliang, Q., Wenqing, H., Xiaogang, W. & Ross, L. (2008), 'A marginal mixture model for selecting differentially expressed genes across two types of tissue samples', *The International Journal of Biostatistics* **4**(1), 1–28.

Whittaker, J. (2009), *Graphical models in applied multivariate statistics*, Wiley Publishing.

Wientjes, E. & Croce, R. (2011), 'The light-harvesting complexes of higher-plant photosystem i: Lhca1/4 and lhca2/3 form two red-emitting heterodimers', *Biochem. J* **433**, 477–485.

Wolfram, S. (1983), 'Statistical mechanics of cellular automata', *Reviews of modern physics* **55**(3), 601.

Xing, E. P., Jordan, M. I., Karp, R. M. et al. (2001), Feature selection for high-dimensional genomic microarray data, *in* 'ICML', Vol. 1, Citeseer, pp. 601–608.

Yang, R., Daigle, B. J., Petzold, L. R. & Doyle, F. J. (2012), 'Core module biomarker identification with network exploration for breast cancer metastasis', *BMC bioinformatics* **13**(1), 12.

Yaramakala, S. & Margaritis, D. (2005), Speculative markov blanket discovery for optimal feature selection, *in* 'Data mining, fifth IEEE international conference on', IEEE, pp. 4–pp.

Zhang, B., Horvath, S. et al. (2005), 'A general framework for weighted gene co-expression network analysis', *Statistical applications in genetics and molecular biology* **4**(1), 1128.

Zhang, S. & Scheller, H. V. (2004), 'Light-harvesting complex ii binds to several small subunits of photosystem i', *Journal of Biological Chemistry* **279**(5), 3180–3187.

Zhang, J., Lu, K., Xiang, Y., Islam, M., Kotian, S., Kais, Z., Lee, C., Arora, M., Liu, H.-w., Parvin, J. D. et al. (2012), 'Weighted frequent gene co-expression network mining to identify genes involved in genome stability', *PLoS Computational Biology* **8**(8), e1002656.

Zhang, X., Zhao, X.-M., He, K., Lu, L., Cao, Y., Liu, J., Hao, J.-K., Liu, Z.-P. & Chen, L. (2012), 'Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information', *Bioinformatics* **28**(1), 98–104.