Bayesian Inference of gene-miRNA regulatory networks

Samuel Touchard

Thesis submitted to the University of Sheffield for the degree of Doctor of Philosophy

School of Mathematics and Statistics,

University of Sheffield

Sheffield, U.K.

September 2014

Acknowledgements

First, I would like to thank my supervisor Dr. Kostas Triantafyllopoulos for his valuable advice and guidance throughout these last four years.

I also would like to thank Professor Chris Newman and Dr. Marta Milo for making this project possible, providing the data and for their help throughout.

I am also grateful to all the Department, students and staff, for creating such a friendly atmosphere. Those laughing Friday nights were really enjoyable and helped escape work.

I would like to thank all the friends I happened to meet throughout these years, for their support and company, from the Department, PROGRESS, Latin American Society, Athletics Club, etc. There are too many people to name them all, but I would like to mention Nick, Paul, Alastair, Perla, Hugh, Joseph, Joanna.

Finally, I would like to thank my family, especially my parents, for all their permanent support in these nine years of higher education.

Abstract

Nowadays, in the post-genomics era, one of the major tasks and challenges is to decipher how genes are regulated. The miRNAs play an essential regulatory role in both plants and animals. It has been estimated that about 30% of the genes in the human genome are down-regulated by microRNAs (miRNAs), short RNA molecules which repress the translation of proteins of mRNAs in animals and plants. Genes which are regulated by a miRNA are called targets of this given miRNA. Hence, the task is to try to determine which miRNAs regulate which genes, in order then to build a network of these DNA components. Knowledge of the functional miRNAs-genes interactions can help find the source or reason of a genetic disease, then we can focus on drugs and their effects such we get more efficient treatments.

In this thesis, we aim to build a Bayesian graphical model that infers a regulatory network by integrating miRNAs expression levels with their potential mRNA targets. We incorporate biological information, such as structure and sequence information, via the prior probability model. The method is broken down to 3 stages. First, a dimensionality reduction is performed; the gene expressions are narrowed down by using biological information (association scores and type of probe set), and distance similarity procedures such as clustering of correlated or co-expressed variables. Second, a Bayesian graphical model is proposed, according to which associations of gene and miRNA expressions are inferred, and an association matrix is extracted. The methodology uses simulation-based methods, as Markov Chain Monte Carlo, and benefits by managing uncertainty at a complex network. Finally, using the association matrix, the regulatory network is constructed.

Contents

1 Introduction and motivations **2**

1.1 Introduction and aim of the thesis 2

1.2 Thesis layout 4

2 Background and Literature Review **7**

2.1 Introduction-Background 7

2.1.1 Non-technical introduction to genetics 7

2.1.2 DNA and synthesis of proteins 8

2.1.3 Gene 12

2.1.4 Messenger RNAs 13

2.1.5 MicroRNAs 14

2.1.6 Next-generation sequencing 16

2.2 Literature review 18

2.2.1 MiRNA target prediction 18

2.2.2 miRNA target prediction algorithms 28
2.2.3 General presentation of networks 31
2.2.4 Review of statistical models for biological networks . . 38
2.3 Conclusion 39

3 Presentation of the data and study 40

3.1 Introduction 40
3.2 Clinical study 43
3.2.1 Acute Coronary Syndromes 43
3.2.2 MiRNA profiles 45
3.2.3 TargetScan scores 45
3.2.4 Target genes profiles 47
3.3 Reducing the dimensionality 49
3.3.1 Reducing the number of target genes 49
3.3.2 Clustering 52
3.4 Conclusion 56

4 Bayesian Graphical Modeling 57

4.1 Introduction to graphical models 57
4.2 Model 59
4.2.1 Framework 59
4.2.2 Selection of regressors under the regulatory network . . 63
4.2.3 Target scores in the prior model 64
4.2.4 Graphical representation 65
4.3 Estimation procedure 67
4.3.1 Posterior inference 67

4.3.2 Estimation of the regression coefficients 72

4.3.3 MCMC algorithm 72

4.4 Summary and pseudo-code of the algorithm 75

4.5 Setting up the parameters 75

4.6 Case study on simulated data 80

4.6.1 Building of the network 81

4.6.2 Covariance matrix assumed to be known 84

4.6.3 Genes assumed to be independent 87

4.6.4 Estimation of the covariance matrix 91

4.6.5 Monte Carlo study 106

4.7 Conclusion 108

5 A regulatory network for Acute Coronary Syndromes 110

5.1 Clustering of the data 111

5.2 Application to the data in clusters 112

5.3 Discussion 121

5.3.1 Special case: cluster of independent genes 121

5.3.2 Estimation of $\mathbf{\Sigma}$ 121

5.4 Interpretation of the different networks 123

5.4.1 Precautions to take into account 123

5.4.2 Main observations from the graphs 125

5.5 Conclusion 127

6 Conclusion and Discussion 129

6.1 Conclusions 129

6.2 Extensions and future work 131

A Appendix **134**

A.1 Kronecker product and vec operator 134

A.2 Distribution theory 137

A.3 Graphs from the different chains run in Chapter 5 145

List of Figures

2.1 Representation of DNA 10

2.2 Synthesis of proteins 12

2.3 Structure of a mature mRNA 13

2.4 Biogenesis of miRNAs 15

2.5 Principal types of miRNA and target matches: from top to bottom, 6mer site, 7mer-A1 site, 7mer-m8 site, 8mer site. For each match, the first line represent the poly(A) tail of the mRNA in the 3' UTR end, and the second line the 5' UTR end of the miRNA. The red nucleotides represent a complementarity between these nucleotides (A with U and C with T), where the blue color shows the lack of complementarity. . 21

2.6 Atypical miRNA target sites: 3' supplementary site and 3' complementarity site. For each match, the first line represent the poly(A) tail of the mRNA in the 3' UTR end, and the second line the 5' UTR end of the miRNA. Red nucleotides show complementarity in the seed of the miRNA, green nucleotides supplementary or compensatory complementarity outside the seed region, blue nucleotides no complementarity. The black nucleotides are the non complementory nucleotides between the two matching regions, and the mismatch in the seed region in the case of the compensatory site. 22

2.7 Directed or undirected networks 34

3.1 Histograms of miRNAs expression levels 46

3.2 Histograms of genes expression levels 49

3.3 Histograms of genes and miRNAs expression levels 50

3.4 QQ-plots for miRNAs and genes datasets, under each condition: STEMI, NSTEMI, and Unstable Angina 50

3.5 Subset of genes composed of two different clusters with different patterns: one cluster with an average expression level close to 8, a second one with an average close to 4. 53

4.1 A graphical representation of regulatory network between 3 miRNAs and 7 genes. 60

4.2 Graphical representation of the model, with the dependencies between the parameters and variables within the model 66

4.3 Simulated correlations of $\boldsymbol{Y}|\boldsymbol{X}$. 82

4.4 Number of included arrows (left) and number of selected arrows (right). 85

4.5 Log-relative probabilities of the visited models, for all iterations (left), and for after the burn-in (right). 87

4.6 Log-likelihood $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$, for all iterations (left), and for after the burn-in (right). 88

4.7 Values of τ. 88

4.8 Log-probability (left panel), and number of arrows included in the network (right panel), for all iterations. 90

4.9 Number of arrows selected, with a threshold of 50% (left panel), and 75% (right panel), after the burn-in. 91

4.10 Number of selected arrows, at a 75% confidence level. 95

4.11 Log-probability of the model and relative log-likelihood of the covariance matrix 96

4.12 Number of arrows included in the model (left panel), and number of arrows selected at a 50% confidence level (right panel). 96

4.13 Values of τ. 97

4.14 Number of arrows included in the network (left panel), and the log-likelihood of the model (right panel). 100

4.15 Log-relative likelihood of $\boldsymbol{\Sigma}$ (left panel), and log-likelihood $f(\boldsymbol{Y}|\boldsymbol{X,R,\Sigma})$ (right panel), after the burn-in. 101

4.16 Number of arrows selected, with a threshold of 50% (left panel), and 75% (right panel), after the burn-in. 102

4.17 Log-relative probabilities of the visited models, through all iterations (left) and after burn-in (right). 104

4.18 Number of arrows included in the network (left panel), and number of arrows selected with $t_h = 75\%$ (right panel), after the burn-in. 104

4.19 Boxplots of the sensitivity (left panel) and false positive rate (right panel) of the Monte Carlo study. 108

5.1 Number of arrows included in the visited models, STEMI condition 115

5.2 Number of arrows included in the visited models, NSTEMI condition 116

5.3 Number of arrows included in the visited models, Unstable Angina condition 117

5.4 Network for the STEMI condition 118

5.5 Network for the NSTEMI condition 119

5.6 Network for the UA condition 120

A.1 Log-relative probabilities of the visited models, STEMI condition146

A.2 Log-relative probabilities of the visited models, NSTEMI condition 147

A.3 Log-relative probabilities of the visited models, Unstable Angina condition 148

A.4 Log-relative probabilities of the visited models after burn-in, STEMI condition 149

A.5 Log-relative probabilities of the visited models after burn-in, NSTEMI condition 150

A.6 Log-relative probabilities of the visited models after burn-in, Unstable Angina condition 151

List of Tables

4.1 Framework of the model 76

4.2 Posterior inference: derivation of $f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})$ 77

4.3 Pseudo-code of the MCMC approach 78

4.4 Comparison of the different chains, with a threshold $t_h = 0.75$ 105

4.5 Summary of the Monte Carlo study performed on 100 analyses of simulated data: mean and standard of sensitivity and false positive rate. 107

5.1 Summary of the regulatory networks for each condition. For example, the regulatory network created for the STEMI condition contains 98 interactions, between 50 different genes and 12 different miRNAs. 114

5.2 Interactions present under the 3 conditions 125

CHAPTER 1

Introduction and motivations

1.1 Introduction and aim of the thesis

Nowadays, in the post-genomics era, one of the major tasks and challenges is to decipher how genes are regulated. The microRNAs (miRNAs) play an essential regulatory role in both plants and animals. As an illustration, it has been estimated that about 30% of the genes in the human genome are down-regulated by miRNAs, short RNA molecules which repress the translation of proteins of mRNAs in animals and plants. Genes which are regulated by a miRNA are called targets of this given miRNA. Hence, the task is to try to determine which miRNAs regulate which genes, in order then to build a network of these DNA components. The aim of the study is then to identify with high confidence a small set of potential miRNA-gene interactions, which will then require more investigation to determine if they can be considered functional and as a genetic cause of a disease. Knowledge of the functional miRNAs-genes interactions can help find the biomarker (source or reason) of a genetic disease, then we can focus on drugs and their effects such we get more efficient treatments.

Several algorithms have already been developed to determine and predict potential miRNA-messengerRNA (mRNA) interactions, based on the sequence and structure characteristics of the miRNAs and their target sites. The main factors used by these algorithms are the sequence complementarity, hybridization energy and comparison across species. Generally, these algorithms predict loads of potential miRNA-mRNA interactions, sometimes different ones as they use different factors to establish their targets. So it can quickly become too difficult for researchers to find, among these hundreds of thousand of potential interactions, those who are functional under particular clinical conditions and thus play a crucial regulatory role under these conditions.

The aim of the thesis is to develop a statistical model which identifies with high confidence a set of potential targets and functional interactions. The regulatory relationships can be inferred by integrating expression levels of both miRNAs and their candidate target genes. The proposed methodology consists of a directed Bayesian graphical model, in which biological knowledge is incorporated, such as negative regression coefficients, as it is believed that miRNAs down-regulate the expression of the genes. We also take into account in our prior model the target scores predicted by TargetScan [19] based on sequence and structure information. A Markov Chain Monte Carlo (MCMC) procedure is proposed to select the variables of interest from the association matrix, before creating the networks.

In this thesis, we propose that correlated genes (which are likely to share some functional relationship) should not be treated as independent. One challenge of our approach is that we wish to relax an assumption that is often made in genomics: the independence of the genes given the miRNAs. Indeed, even if it is commonly admitted that some genes, especially the ones part of a group of genes responsible of one specific biological function, are correlated, this characteristic is often omitted, partly for computational convenience. We apply a gene-clustering algorithm in order to form groups of correlated genes, these groups assumed to be independent from each other. We then perform a Bayesian graphical modeling approach to each of these clusters, where we attempt to estimate the main parameters of the model, such as the matrix of interactions. From these analyses, we then draw the regulatory networks, which are composed of much more arrows than the ones built assuming the genes independent [58].

1.2 Thesis layout

The remainder of the thesis is organised as follows. Chapter 2 offers a literature review of the different topics present in the thesis. In a first section, we give an introduction to the genetics, especially a description of genes and miRNAs, what are their role in the living cell, etc. In a second section, we present the principles of miRNA target prediction algorithms, like the sequence information, and details about two of them, TargetScan and MiRanda. We then give a presentation of the characteristics of networks and graphs, as well as a description of the Metropolis-Hastings procedure. Finally, an introduction to graphical models is provided.

This PhD thesis is motivated by considering interaction networks of miRNA-gene for Acute Coronary Syndromes (ACS). Chapter 3 describes the ACS data and provides an exploratory data analysis. We explain the characteristics of the different conditions we are interested in, as well as the different methods we use to narrow down the number of genes, in order to make the dataset manageable and perform an efficient study. This will serve as a preparatory data analysis before we apply the model in Chapter 5.

Chapter 4 develops the proposed Bayesian graphical modeling estimation approach. After describing the model (framework, assumptions, parameters, etc), we perform our methodology on a case study of simulated data, evaluate its efficiency and also to compare it with the approach from Stingo et al.[58], where the genes are assumed independent given the miRNAs. This provides an aid to fix and set up some parameters before we apply the model to the real data.

In Chapter 5, we apply our proposed approach to the real data, divided into the different clusters, and under the three different Acute Coronary

Syndromes conditions. Regulatory networks are then constructed, and their main characteristics are commented. This allows us to make suggestions about the meaning of these networks, which can be seen as guidelines for further investigation to find a potential genetic source of the given condition. However, we also explain that these ideas can not be considered as final conclusions.

Chapter 6 gives a summary of the thesis, with the advantages and limitations of the methodology. We also discuss some suggestions for further work and/or improvement of the proposed methodology.

The main analysis has been performed with the R software, when the drawing of the different networks has been realized with the Cytoscape software [57]. Both softwares are freely available.

CHAPTER 2

Background and Literature Review

2.1 Introduction-Background

In this chapter, we are going to describe some biological knowledge which is required to assimilate and understand the biological project of research: what is DNA, what is a gene and miRNA, how do they link... Also, as we aim to build a network of functional interactions, we will provide a small introduction to networks and graph theory.

2.1.1 Non-technical introduction to genetics

Genetics is the science related to the study of genes, whose purpose is to find out and explain what they are and how they work. Genes are molecular units of heredity of living organisms, and, technically, stretches of DNA (deoxyribonucleic acid) and RNA (ribonucleic acid) that code for a type of protein or for an RNA chain that has a function in the organism. This is the reason why children usually look like their parents because they inherited their parents' genes. Genes are made from a long molecule called DNA, which is copied and inherited across generations. DNA is made of simple units that line up in a particular order within this large molecule. These units are ordered in such a way that they carry the genetic information of the organisms, using the genetic code. This is what provides any living organism to be "constructed" and functional. The information within a particular gene is not always exactly the same between two organisms, which means that different copies of a same gene can carry different instructions. Each variant of a given gene is called an allele. An easy example to illustrate this point is to consider the eye color. There exist various alleles for the eye color, this is the reason why some people have blue eyes (the ones who have the allele responsible on the blue eyes), green eyes, etc. Such changes in genomic sequences creating new alleles are called mutations. They can occur randomly but also due to environment, and this is a key point to evolution. More details about we are going to present or further information on the topic can be found in [1, 2].

2.1.2 DNA and synthesis of proteins

DNA is a double-stranded helix

As previously mentioned, the hereditary information of all living cells, without any exception, is stored in DNA. DNA is formed of double-stranded molecules, long unbranched paired polymer chains, always formed of four different monomers. Each monomer, also called nucleotide, consists of two parts: a sugar (deoxyribose) with a phosphate group attached to it, and a base, which can be either adenine (A), guanine (G), cytosine (C), or thymine (T). The phosphate group allows to link each sugar to another one, creating polymer chain which can be extended by adding monomers at one end. In theory, if we consider a single isolated strand, since the link between any two monomers is the same, any nucleotide should be able to join the chain. However, in reality, DNA is synthesized on a template formed by a preexisting DNA strand. Each base from one strand has to link with another base from the second strand, according to the rule of the complementary structures of the bases: A and T binds to each other while C and G binds to each other, as it can be seen in Figure (2.1). This base-pairing controls which monomer has to added to the new strand. Hence, two complementary sequences form a double-stranded structure. Since both strands twist around each other, they finally form a double helix.

To carry the hereditary information, DNA has to be replicated. The bonds between the base pairs are weak compared with the sugar-phosphate links. Then the two strands can be pulled apart, and then each strand can serve as a template to synthesize a new DNA strand complementary to itself.

Synthesis of mRNAs and proteins: transcription and translation

DNA also has to express its information, for then this information to allow the production of other molecules. That mechanism, visible in Figure

[1] From [1] Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell. 4th edition. New York: Garland Science; 2002. Figure 4-3, DNA and its building blocks. Available from: http://www.ncbi.nlm.nih.gov/books/NBK26821/figure/A598/

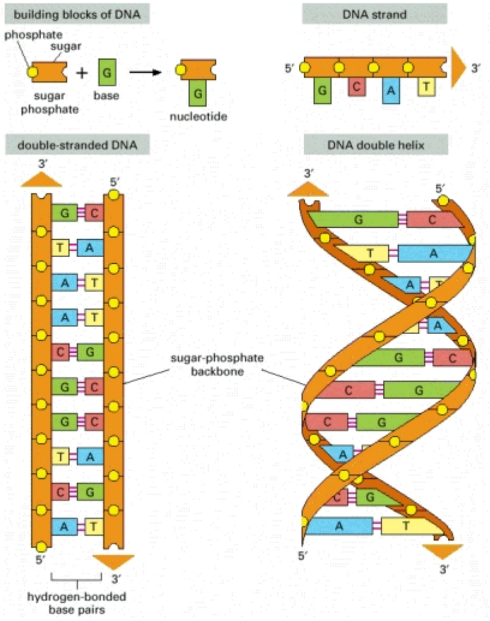

Figure 2.1: Representation of DNA: (top left) composition of a nucleotide, (top right) DNA strand, with the four different bases, (bottom left) complementary DNA strands, (bottom right) double helix structure [1]

(2.2), consists of the production of two other classes of polymers: RNAs and proteins. The first step of this process is called transcription, in which segments of the DNA sequence are used as templates for the synthesis of shorter molecules called RNA. Then, in the second step of the process, the translation, these RNA molecules direct the synthesis of another class of polymers, the proteins. RNA is slightly differently formed from DNA. The main difference for our point is that uracil (U) replaces thymine (T), while the three other bases are the same with the same pairing: A with U and C with G. The RNA outcome of the transcription is a polymer molecule whose sequence of nucleotides represent the cell's genetic information, just with RNA monomers instead of DNA monomers. The same segment of DNA can be used repeatedly to guide the synthesis of many identical RNA transcripts. These RNA transcripts work as intermediates in the transfer of genetic information. This is the reason why we often call them messenger RNA (mRNA) to guide the synthesis of proteins. The translation (synthesis of proteins) is a bit more complex. The information in the sequence of a mRNA is read out in groups of three nucleotides at a time: each triplet of nucleotides, called codon, codes for a single amino acid in a corresponding protein. There exist 20 different amino acids, which means that several codons code for the same amino acid. This reading process is complex, we can just say it is carried by a giant multimolecular machine called ribosome and more than 50 other different proteins. At the end, the amino acid are linked together to form a new protein chain. Proteins are the principal catalysts for most of the chemical reactions in the cell; their specific function depends of the amino acid sequence, specified by the nucleotide sequence of the segment

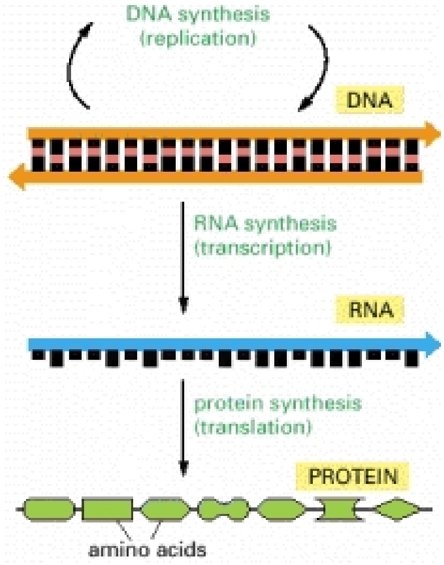

Figure 2.2: Synthesis of proteins: RNA molecules are first transcribed from DNA segments and then guide the synthesis of proteins. [2]

of DNA which codes for that protein.

2.1.3 Gene

As just mentioned, individual segments of DNA are transcribed into separate mRNA molecules, with each segment coding for a different protein. Each such DNA segment represents one gene. However, it is actually a bit more complicated since RNA molecules transcribed from the same DNA segment can often be processed in several ways. This is why we generally define a gene as the segment of DNA sequence corresponding to a single protein.

[2] From [1] Alberts B, Johnson A, Lewis J, et al. Molecular Biology of the Cell. 4th edition. New York: Garland Science; 2002. Figure 1-4, From DNA to protein. Available from: http://www.ncbi.nlm.nih.gov/books/NBK26864/figure/A11/)

2.1.4 Messenger RNAs

We already discussed about the function of the mRNAs in the previous sections, when we described the synthesis of proteins. Here in this section, we will briefly describe the structure of mRNAs. A fully processed mRNA is composed of: a 5' cap, 5' UTR, a coding region, 3' UTR, and poly(A) tail. The 5' cap is a modified guanine nucleotide added to the "front" (5' end) of the mRNA . It provides recognition and proper attachment to the ribosome. The 3' poly(A) tail is a long sequence of adenine nucleotides added to the 3' end of the mRNA. One of its function is to protect the mRNA from degradation. The 5' UTR is the section before the coding region. It begins at the transcription start site and ends one nucleotide before the start codon of the coding region. The 3' UTR follows the coding region. These untranslated regions (UTRs) are transcribed with the coding region but are not translated. They have been attributed several roles in gene expression, for example in our case, the miRNAs bind to the 3' UTR. On average, the 5' UTR is 150 nucleotide long, and the 3' UTR tends to be twice longer. The coding region is the region which codes for protein. This is also commonly called open reading frames (ORFs).

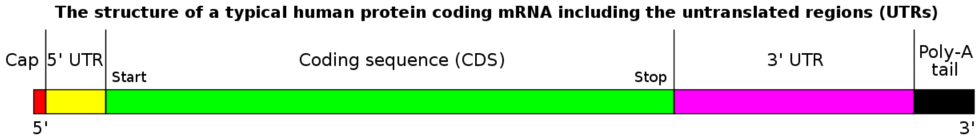
The structure of a typical human protein coding mRNA including the untranslated regions (UTRs)
Cap
5' UTR
Start
Coding sequence (CDS)
Stop
3' UTR
Poly-A tail
5'
3'

Figure 2.3: Structure of a mature mRNA.[3]

[3] From "MRNA structure" by Daylite - Own work. Licensed under Public domain via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:MRNA_structure.svg

2.1.5 MicroRNAs

Biogenesis

MicroRNAs are short RNA molecules, 22 nucleotides long on average. Binding to the 3'UTR of mRNAs, they are post-transcriptional regulators, repressing the translation of proteins or degrading the mRNA (cleavage). The miRNA precursors (pre-miRNA) are first synthesized by the enzyme RNA polymerase II (at that stage, they are called primary miRNAs (pri-miRNA)), then they are cleaved by Drosha RNAse III. These pre-miRNAs are about 60-70 nucleotides long and form an imperfect stem loop structure. Then the loop and the terminal base pairs are cut off by the enzyme Dicer, so this stage is called dicing, resulting in a miRNA:miRNA duplex, the mature miRNA and its complementary strand. That duplex is then assembled with a set of proteins to form an RNA-induced silencing complex (RISC). Finally, the duplex is separated, the mature miRNA is conserved while the complementary strand is degraded. The main steps of this process are briefly illustrated in Figure (2.4).

Biological function

Once formed, the RISC looks for target mRNAs by searching for complementary nucleotide sequences, in order to bind its 5' region to the 3' region

#mediaviewer/File:MRNA_structure.svg

[4] From "MiRNA-biogenesis" by Narayanese (talk) - Own work (Original text: "I created this work entirely by myself.")References:Esquela-Kerscher A, Slack FJ (2006) Oncomirs - microRNAs with a role in cancer. Nat Rev Cancer 6: 259-69. PubMedOkamura K, Hagen JW, Duan H, Tyler DM, Lai EC (2007) The mirtron pathway generates microRNA-class regulatory RNAs in Drosophila. Cell 130: 89-100. PubMed. Licensed under Creative Commons Attribution-Share Alike 3.0 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:MiRNA-biogenesis.jpg #mediaviewer/File:MiRNA-biogenesis.jpg

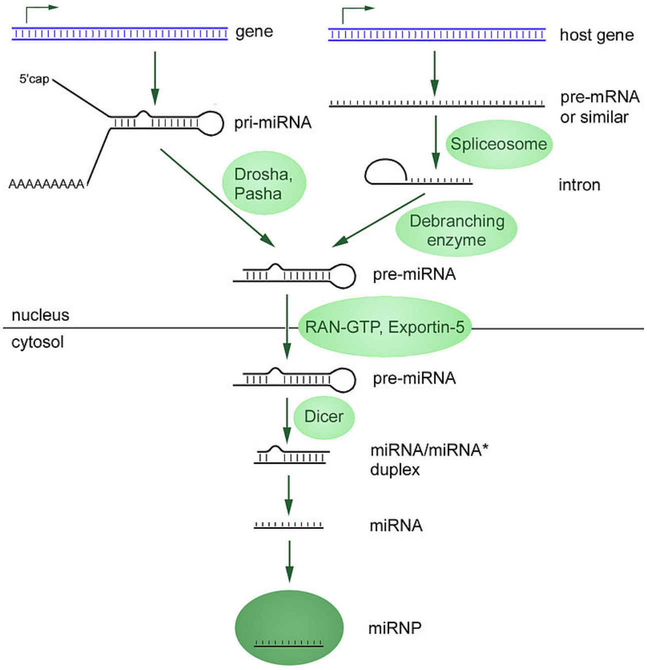

Figure 2.4: Biogenesis of miRNAs: pri-miRNA, pre-miRNA, then incorporation to the RISC complex, before repressing the mRNA translation or degrading the mRNA. [4]

of the miRNA and hence regulate the expression of its target site. This regulation can be done in two possible ways, depending on how extensive the base-pairing is.

If the base-pairing is extensive, the mRNA is cleaved by the Argonaute protein present in the RISC, by removing the poly-A tail, leading to the degradation of the mRNA. After cleaving one mRNA, the RISC and its associated miRNA are released and can search for other mRNAs, meaning that a single miRNA can cleave many mRNAs.

If the base-pairing is less extensive, the translation of the mRNA is repressed and the mRNA destabilized. In few words, the poly-A tail is shortened, the mRNA separated from the ribosome and eventually degraded.

2.1.6 Next-generation sequencing

The technical framework which is commonly used to sequence the genome is the microarray (collection of microscopic DNA spots attached to a solid surface) technology. DNA microarrays can also be used to measure changes in expression levels, or to detect single nucleotide polymorphisms (SNPs), besides of sequencing genomes. However, after few decades of continuous improvement, new alternative techniques have emerged: the second-generation DNA sequencing techniques (or next-generation DNA sequencing, or NGS). The application of any next-generation technique is called RNA sequencing (RNA-seq). A technique using miccroarray technology is PCR amplification whereas Illumina or SOLiD use second-generation sequencing. Although the latest have clear advantages and bring improvements, they also have limitations. Two disadvantages of the second-generation techniques are the read-length and the raw accuracy. Indeed, read-lengths for all the new platforms are much shorter than conventional sequencing, and the base predictions are about tenfold less accurate than base predictions from microarrays. So it is pretty clear that these limitations imply challenges for the future. However, as conventional techniques kept improving over the years, we can think that these new techniques will improve with respect to these issues to reach an higher level of performance. Meanwhile, microarrays also have disadvantages. If we don't perform a very careful data analysis, it can lead to misleading results. But even if the data analysis is perfectly carefully done, the next-generation sequencing techniques still can bring improvements. The main advantage of new technologies is clearly their cost. Indeed, because the high interest in this area and the need to decrease the cost, these second-generation sequencing platforms are able to parallelize the sequencing process, which means they can produce thousands of sequences at once. Consequently, in the immediate future, quite small-scale projects will still depend on conventional technologies. However, larger-scale projects, as miRNA profiling, will quickly become dependent on second-generation sequencing. Hence it is possible to believe that in the future, second-generation sequencing techniques will become as widespread as microarrays and conventional platforms are, and will be more useful to achieve ambitious and challenging projects. For more details about NGS and mRNA-seq, reviews can be found in [41], [17], [7].

In this section, we have presented the main biological concepts we need to have in mind for the understanding of the study, in particular which DNA components (gene and miRNA) are involved, and what is their main functions in the living cell. In the next section, we will describe how these two entities can be linked together.

2.2 Literature review

2.2.1 MiRNA target prediction

Introduction

In this section, we are going to present the main principles of miRNA target prediction, such as the sequence complementarity and conservation across species. A good and non too technical review can be found in [38].

MicroRNAs (miRNAs) are small non-coding endogenous RNAs, around 22 nucleotides long on average, that play an important role in the gene down-regulation in both plants and animals. They pair to the messenger RNAs (mRNAs) of protein-coding genes to control their post-transcriptional repression. They generally bind their 5' UTRs to the 3' untranslated region (3' UTR) of the mRNA, even if it has been found that some of them can bind to the open reading frames (ORFs) or to the 5' UTRs [4, 56]. However, these target sites are less effective and less frequent than target sires located in the 3' UTRs, especially the 5' UTR targeting which is very rare.

Historically, the first miRNA discovered is *lin-4* in 1993, a miRNA involved in the timing of larval development in worms *C.elegans*[36]. The second miRNA, *let-7*, was discovered 7 seven years later [47]. It has a similar role as *lin-4*, regulating developmental timing in *C. elegans*. It has been reported quite soon after that both *lin-4* and *let-7* are part of a popular class of small endogenous RNAs we can find in worms, flies and mammals, and that is when they were officially called microRNAs [33, 34, 37]. Since then, thousands of miRNAs have been identified in various and diverse organisms, through sequencing and/or computational prediction.

Some miRNAs can have hundreds of targets, if they are highly conserved across several species [4]. The identification of miRNA-target interactions is made easier when we observe a perfect complementarity (or near perfect complementarity) between the miRNA and target site sequences. This feature is particularly true for plants [48]. For animals, it is a little bit more complicated because only few miRNAs present perfect complementarity to their targets.

Principles of miRNA target identification

Sequence complementarity

The Watson-Crick sequence complementarity (or pairing) is probably the major criterion for target identification. Indeed, it highly improves the performance of the prediction, especially reduces the false positive rate, and that is why this criterion is used in the most renowned prediction algorithms.

Watson-Crick pairing implies sequence complementarity between the mRNA (target) and the "seed" of the miRNA, which is located in the 5' end of the miRNA, on nucleotides 2-7. Then we can define several types of "matches" or sites. We can make the distinction between the most common and principal types of matches, and some atypical matches which remain rare. The main matches are illustrated in Figure (2.5) and are composed of the following 4 types of matches:

- 6mer site: seed match. 6 nucleotides sites match the seed region. Such sites are conserved by chance more frequently than the other sites and have a low efficacy. Thus, prediction algorithms that involve stringent seed-pairing do not take 6mer sites into account.

- 7mer-A1 site: seed match + A at position 1. An outperforming site occurs when we require a 7nt match, with an A across position 1 (1A-anchor) of the miRNA over a Watson-Crick match. For example, the algorithm TargetScan uses it.

- 7mer-m8 site: seed match + match at position 8. Instead of requiring an 1A-anchor, other algorithms, as miRanda, rewards 7nt match sites with a supplementary match at position 8.

- 8mer site: seed match + A at position 1 + match at position 8. These are the result of both 7mer-A1 and 7mer-m8 sites, which means Watson-Crick pairing at position 2-8 plus an 1A-anchor.

We can note that most miRNAs targets have a 7nt match. Requiring perfect pairing (8mer) increases specificity whereas a "simple" 6mer site increases sensitivity. Finally, if we wish to describe which site is the most outperforming one, we can write that 8mer > 7mer-m8 > 7mer-1A > 6mer.

As mentioned earlier, two other kinds of "atypical" sites can also be considered, as illustrated in Figure (2.6). The first one is called 3' supplementary site. It supplements seed pairing and therefore improves the chances of binding. Such sites ideally centers on miRNA nucleotides 13-16 and are at least 3 or 4 nucleotides long, uninterrupted by mismatches or wobbles. Sites with supplementary pairing are predicted with a significant better specificity, but it appears that they are rare and only have a slight effect. Consequently, it

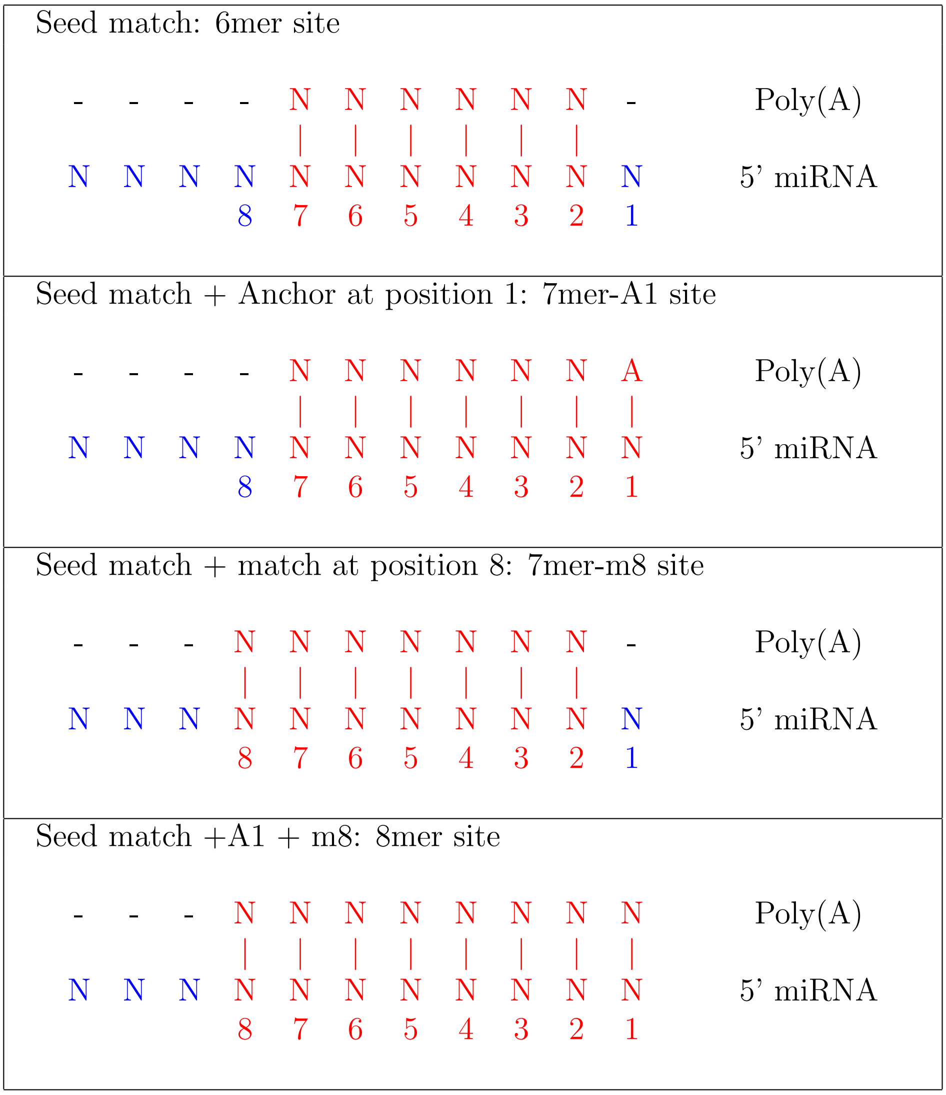
Seed match: 6mer site
- - - - N N N N N N - Poly(A)
N N N N N N N N N N N 5' miRNA
8 7 6 5 4 3 2 1
Seed match + Anchor at position 1: 7mer-A1 site
- - - - N N N N N N A Poly(A)
N N N N N N N N N N N 5' miRNA
8 7 6 5 4 3 2 1
Seed match + match at position 8: 7mer-m8 site
- - - N N N N N N N - Poly(A)
N N N N N N N N N N N 5' miRNA
8 7 6 5 4 3 2 1
Seed match +A1 + m8: 8mer site
- - - N N N N N N N N Poly(A)
N N N N N N N N N N N 5' miRNA
8 7 6 5 4 3 2 1

Figure 2.5: Principal types of miRNA and target matches: from top to bottom, 6mer site, 7mer-A1 site, 7mer-m8 site, 8mer site. For each match, the first line represent the poly(A) tail of the mRNA in the 3' UTR end, and the second line the 5' UTR end of the miRNA. The red nucleotides represent a complementarity between these nucleotides (A with U and C with T), where the blue color shows the lack of complementarity.

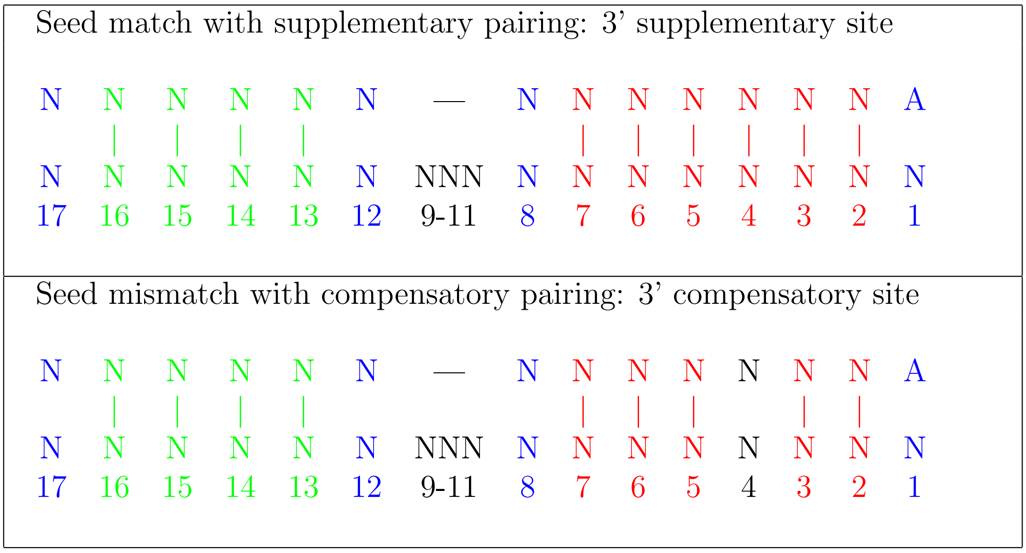
Seed match with supplementary pairing: 3' supplementary site
N N N N N N — N N N N N N N A
N N N N N N NNN N N N N N N N N
17 16 15 14 13 12 9-11 8 7 6 5 4 3 2 1
Seed mismatch with compensatory pairing: 3' compensatory site
N N N N N N — N N N N N N N A
N N N N N N NNN N N N N N N N N
17 16 15 14 13 12 9-11 8 7 6 5 4 3 2 1

Figure 2.6: Atypical miRNA target sites: 3' supplementary site and 3' complementarity site. For each match, the first line represent the poly(A) tail of the mRNA in the 3' UTR end, and the second line the 5' UTR end of the miRNA. Red nucleotides show complementarity in the seed of the miRNA, green nucleotides supplementary or compensatory complementarity outside the seed region, blue nucleotides no complementarity. The black nucleotides are the non complementory nucleotides between the two matching regions, and the mismatch in the seed region in the case of the compensatory site.

is suggested and assumed that supplementary 3' pairing only plays a modest role in miRNA-target identification.

The other atypical site is the "3' compensatory site". They have been given that name for the simple reason that they compensate for a single-nucleotide mismatch in the seed region. Such sites center on nucleotides 13-17 of the miRNA, last at least 4 or 5 pairs, and can extend to 9 consecutive Watson-Crick pairs. However, 3' compensatory sites are quite rare.

Conservation

Another feature for target identification is the principle of conservation. It appears that some binding sites are conserved across several species. In that case, they are likely to be biologically functional, and thus these sites are potential miRNA target sites. That requires knowing several genomes. Fortunately, enough genomes have been sequenced and aligned such that study is feasible. Therefore, sites can be predicted as targets with more confidence. Indeed, the use of conserved site sequences reduces significantly the false-positive rate.

Thus, a protocol to predict evolutionarily conserved targets for a miRNA can be split in three steps, as follows:

- identify 7 nt matches (either 7mer-m8 or 7mer-1A) to the seed region.
- use whole-genome alignments from other species to draw up a list of orthologous 3' UTRs.
- within these orthologous UTRs, search for conserved occurrence of either 7 nt match. These are predicted regulatory sites.

Some miRNAs present the same seed region, the same sequence at positions 2-8. These form a miRNA family, and hence all share the same predicted targets.

However, we need to precise what we mean by conservation, because some prediction programs do not always use the same definition for that concept. In general, sites are regarded as conserved when they appear at the exact same position in the 3' UTRs alignments. But it is sometimes considered sufficient when the regions matching the region seed fall in overlapping positions. They also can be considered as conserved just if the matching region is located somewhere not in the aligned positions. Finally, when we are studying conservation in several genomes, it sometimes appear that the site of interest is missing or has changed (because of a mutation for example) in just one of the organisms taken into account. In that case, the site is called poorly conserved.

Thermodynamics and site accessibility

Another principle used in some prediction algorithms is the thermodynamics stability, especially the free energy E_{duplex} of the miRNA-target duplex. From an energetic point of view, it is preferable when two complementary strands of RNA are hybridized, together. That means that, the lower (more negative) the free energy of the duplex or two RNA strands, the more energy is needed to break this structure. Consequently, the binding link between a miRNA and a target mRNA is stronger when E_{duplex} is low, and therefore the duplex of interest has an higher probability to be biologically functional.

We also might have to look at the secondary structure of the mRNA, since it also plays quite an important role. To facilitate the binding, the target site has to be accessible. That means it has to be opened and not to interact with other sites within the mRNA. That includes that it is better when a length of 15nt upstream and downstream the target site are also open. That requires an energetic cost E_{open} to open the site, a cost that we need to consider. We then introduce the total free energy score E_{total}, which represents a score for the accessibility of the site, and hence the probability for the miRNA to bind to this target site of interest. This total free energy score is defined as the difference between the free energy of the duplex and the energy required to open the site:

$$E_{total} = E_{duplex} - E_{open}. \tag{2.1}$$

Evaluating the performance

As already said, several prediction programs and algorithms exist to predict miRNA target sites. Since they do not have the same approaches and use different principles in their implementations, they can not get the exact same results and conclusions. For example, one program will predict a miRNA-target interaction to be functional whereas a second program will conclude that this same interaction is not. Then, how can we decide between two results, and how can we know if a given method is more likely better than the others?

To do so, we use some quantities already mentioned. The first one is called the sensitivity, or also true positive rate (TPR), and is defined as follows:

$$\text{Sensitivity} = \frac{\text{True positives}}{\text{True positives} + \text{ False negatives}}, \tag{2.2}$$

where "true positives" (TP) is the number of predicted miRNA-target interactions that do actually exist (number of interactions the program correctly predicted), and "false negatives" (FN) is the number of miRNA-target interactions that exist but that the algorithm did not predict.

A second quantity used to estimate the performance of a prediction program is the specificity, which can be seen as the ratio between the correctly non-predicted interactions and the number of total non-existing interactions (correctly non-predicted and uncorrectly predicted). Thus it is defined as:

$$\text{Specificity} = \frac{\text{True negatives}}{\text{True negatives} + \text{False positives}}, \tag{2.3}$$

where "true negatives" (TN) is the number of correctly non-predicted interactions, and "false positives" is the number of interactions which have been predicted as functional by the algorithm but actually do not exist.

The false-positive-rate (FPR) is also often mentioned in studies, most often instead of the specificity. That is defined as the ratio between the false positives and the total number of functional miRNA-target interactions:

$$\text{FPR} = \frac{\text{False positives}}{\text{False positives} + \text{True negatives}} = 1 - \text{Specificity}. \tag{2.4}$$

Then, for a given program to be performing, we need to get as few as possible false positives and false negatives, which means we need to maximize both sensitivity and specificity at the same time. Sensitivity can for instance be improved by setting less stringent thresholds, but meanwhile specificity will be reduced because we will have gotten more false positives. That is why we need to get the best compromise between these two quantities.

The most suitable to optimize the relation between sensitivity and specificity is a Receiver Operating Characteristic Analysis (ROC) analysis [63]. In a ROC curve, sensitivity, which is the true positive rate, is plotted in function of the false-positive rate (1- specificity) for different points, where each point represents a sensitivity-FPR pair corresponding to a particular decision threshold. A perfect test's ROC curve, with no overlap between both distributions, passes through the upper left corner, point and conditions which give both perfect 100% sensitivity and specificity. Hence the bigger the area curve is, the more accurate the method is.

To be able to determine sensitivity and specificity, the data set of interest has to contain a sufficient number of unbiased miRNA-target interactions which have previously been defined experimentally, either verified or refuted, thus we get enough knowledge to compute the false positive number, true positive...

A final tool used to evaluate the efficiency of a prediction algorithm is the signal-to-noise ratio (SNR). This is done by using shuffled miRNA sequences, which means randomly permuted [39]. The signal, number of predicted targets, is compared to the noise, number of targets predicted for loads shuffled miRNAs, since we can get loads of different shuffled sequences for one single miRNA. Since these shuffled sequences are unlikely to be biologically relevant, the noise is lower than the signal. Thus, that suggests that most of the predicted conserved targets are biologically functional. Finally, the higher the SNR is, the more significant the results of the method of interest are.

Experimental verification

Since the existing prediction tools still lack sensitivity and specificity, and since interactions need to be confirmed or infirmed to compute these quantities, it is essential to verify experimentally the predictions done by the algorithms. Few techniques are currently available to do so. The most common one is the reporter gene assay, which provides direct evidence about the functionality of a miRNA-mRNA pair. We call a reporter gene a protein or enzyme which allow us to say if a gene is expressed in a cell. The two most common reporter genes are the Green Fluorescent Protein (GFP) and

Luciferase. When genes are expressed in a cell and exposed to blue light, GFP fluoresces while Luciferase causes bioluminescence. Hence the expression of the genes can be quite easily quantified. So, when we wish to verify a miRNA-mRNA interaction, we attach the 3' UTR of observed mRNA downstream of the reporter gene and introduce it into a cell of interest. We then measure the expression level of the reporter gene, in both absence and presence of a specific miRNA. It is thus quite easy to draw conclusions about the miRNA-mRNA interaction.

Other techniques can also be used, as microarray analysis for example, which measures changes of mRNA levels, which then allows us to detect interactions which cause mRNA cleavage and degradation. The main drawback of this method is that it only provides indirect evidence of interaction because it just detects changes in expression profiles and not the direct interaction of a miRNA-mRNA pair. Another possibility is to overwhelm the miRNA gene and observe the effects on rotein changes. However, we can not deduce with confidence an interaction with this method since a miRNA can target a large number of genes.

2.2.2 miRNA target prediction algorithms

Introduction

It is now established that predicting miRNA target genes is very important for a better understanding of the genes regulation. That is the reason why several algorithms have been developed. Since we will use some of these prediction target scores, we need to have an idea how these scores are computed. Thus now, we will give the main steps of two of these algorithms.

TargetScan

TargetScan has first been developed in 2003 by Lewis et al [39]. This algorithm uses a set of perfectly conserved miRNAs, and several sets of orthologous 3' UTRs from different organisms (human, mouse, rat, *Fugu*). The idea is that it compares and analyses the sequences to predict miRNA targets conserved across these different genomes, it also the modeling of RNA-RNA duplex interactions, modeling which is based on thermodynamics.

In more details, given the different sets of UTR sequences, and a conserved miRNA in the organisms of interest, TargetScan can be split into several steps:

- in the first organism, after numbering the miRNA bases from the 5' end, it looks for segments of the UTRs which perfectly match the bases 2-8 of the miRNA (in the sense of Watson-Crick complementarity). If at least one match is found between the miRNA and the UTR, the 7 nucleotides segment of the miRNA is called the "miRNA seed" and the one(s) of the UTR the "seed matches".

- tries to extend the seed matches in both directions, stopping when it finds a mismatch.

- optimizes base pairing of the remaining 3' portion of the miRNA to the bases of the UTR immediately 5' of each seed match, using RNAfold program, thus extending each seed match to a longer potential target site.

- assigns a folding free energy G to each miRNA-target site duplex, using RNAeval (RNAlib). For further information on these two programs just mentioned, see Hofacker et al [22].

- assigns a score, Z, to each UTR, score defined as $Z = \sum_{k=1}^{n} \exp(-G_k/T)$, where n is the number of seed matches in the UTR, G_k the folding free energy, calculated in the previous step, of the k^{th} miRNA-target site for that UTR, and T a preassigned parameter. If an UTR does not have any seed match, is is assigned a Z score of 1.

- the UTRs of the organism are then arranged by Z score and are thus assigned a rank $R_i, i = 1, \ldots,$ number of seed matches in the UTR.

- it repeats these first six steps for each organism.

- the genes which are finally predicted as targets are the ones for which both $Z_i \geq Z_c$ and $R_i \leq R_c$ for an orthologous UTR sequence in all organisms, where Z_c and R_c are prechosen Z score and rank cutoffs.

Later, an updated and simplified version of this algorithm, called TargetScanS, was published in 2005 [39]. The main difference with the first version is that in this procedure, we only require a 6mt Watson-Crick seed match at positions 2-7. This 6nt match can be followed by a match at position 8 (so we have a 7 nt match) and/or by an "A anchor" (nucleotide A at position 1). It was also required that these matches occur at conserved positions in a multiple alignment of orthologous UTRs. However, the thermodynamics criterion described in the first version of the algorithm is no longer taken into account.

A more recent version was published in 2008 [19]. As the others, it looks for conserved 7 or 8 nt sites which math the miRNA seed. This algorithm allows us to predict nonconserved sites, as well as nonperfect sites (mismatches with the miRNA seed) which have 3' compensatory pairing. In details, a score is computed by studying four differents features: the site-type contribution, the 3' pairing contribution, the local AU contribution, and the position contribution. The site-type contribution represents the type of seed match, while the 3' pairing contribution reflects any consequential complementarity outside the seed region, in particular between nucleotides 12 and 17. The local AU contribution refers to the concentration of A and U nucleotides flanking the corresponding seed region of the miRNA, as it is believed that the match in the 3' end of the mRNA is more likely to accur in a rich AU context [19]. Finally, the position contribution analyzes the position of the target site within the mRNA. For all these features, a more negative score is associated with a more favorable site, and the contect score is the sum of these four scores. Even if most of the targets predicted are quite the same as those predicted in the earlier versions of TargetScan, it considers site conservation in more genomes (10 in total), conservation is better detected, and gives when needed probabilities of preferentially conserved targeting.

In this section, we described the principles for miRNA targeteing, how miRNAs can bind to target genes, and we also gave the main details of one particular algorithm, TargetScan, which we will use in our framework, as we will in Chapter 4.

2.2.3 General presentation of networks

In this section, we are about to describe a concept more mathematical than biological, with noneless applications to biology. As our aim is to infer a network of biological interactions, we need to present the main properties of networks, especially metabolic or biological networks, such as their scale-free property.

Introduction

A living cell is composed of many molecules, molecules which interact together. It is essential to understand how these molecules determine the function of the cell, both on their own but also together. At the beginning of biological research, people were focused on reductionism, which means they were more interested in individual components and their functions. It has been very successfull and provided a lot of knowledge. But it is now quite clear that most of the biological functions are due to complex interactions between the several cell's components, such as proteins, DNA, RNA etc, and not to an individual one. Thus, it is getting more and more important and challenging to understand the structure and the dynamics of that complex intercellular network of interactions, which determine the structure and the function of a cell.

Basic vocabulary

Nodes and links

As said earlier, most of the complex systems, from the cell to the Internet, work from the synergistic activity of many components which interact through pairwise interactions. Mathematically speaking, each component is called a node, and the interactions between two nodes is called a link. In biological networks, the links between nodes represent the chemical reactions that can convert a substrate into another one. Then the nodes and the links together form a network, or a graph in a more formal mathematical language. Each node k can be characterized by a number Deg, which represents the number of nodes linked to this node of interest. This number is called the degree, or connectivity, of the node.

Directed or undirected networks

According to the nature of the interactions between nodes, networks can be directed or not directed. In undirected networks, the links do not have an assigned direction, and the relationship can go in both ways, which means that both node of the link can have an effect on the other one. On the other hand, in directed networks, the relationship has a well-defined direction. The first node can affect the second one, but the second one does not have any effect on the first one. Biological networks are directed because chemical reactions are irreversible. For example, if we focus on a chemical or metabolic reaction, the direction can represent the direction of flow from the substrate to a product. In a miRNA-gene network, since it is well-known that miRNAs down-regulate genes, the direction can mean that the expression level of the miRNA has an effect on the gene expression level. To give a graphical representation, if we have a look at Figure (2.7), we can see that the link between nodes B and D is undirected (two arrows in both ways) so they mutually can affect each other, whereas the one between E and F is directed, and thus E can have an effect on F but this is not reciprocal.

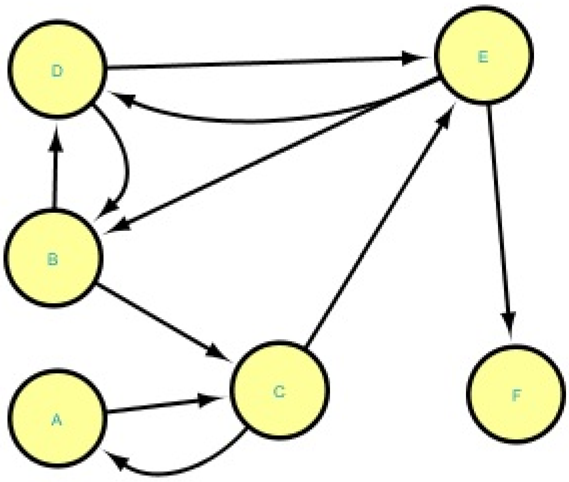

Figure 2.7: Directed or undirected networks

Random, scale-free and cellular networks

Random networks

Some quite complex networks can be modelled by simple models, even sometimes by completely random networks. Random networks were introduced by Erdos and Renyi in 1959 [10]. That defines a graph with N nodes and with n links which are chosen randomly from the $N(N-1)/2$ possible links. So we have $C^n_{N(N-1)/2}$ equiprobable possible graphs. We can model it with a binomial model. We have N nodes and let p be the probability of connecting each pair of node ($N(N-1)/2$ Bernoulli experiments). As a result, the total number of links in the network follows a Binomial distribution, and the expectation is naturally the mean of a Binomial distribution: $pN(N-1)/2$. The maximum number of possible links is $N-1$ for each node. Thus, if we define Deg the degree of a node, Deg follows a Binomial distribution with parameters p and $N-1$:

$$D = Pr(Deg = k) = C_{N-1}^{k} p^{k} (1-p)^{N-1-k}.$$

Then the expected number $E(X_k)$, where X_k is the random variable of the number of nodes having k links (nodes of degree k) can be derived quite simply: $ND = \lambda_k$. If we consider that the nodes are independent, so are their degrees, we can use a theorem from Bollobas' probability used on graphs which states that X_k follows a Poisson distribution with parameter λ_k:

$$P(X_k = r) = e^{-\lambda_k} \frac{\lambda_k^r}{r!},$$

If we simplify, we can say that $X_k = ND = \lambda_k$, for large N, the Binomial distribution with probability p and N nodes, with Np fixed and p small, can be approximated by a Poisson distribution, thus the probability for a node to get exactly k links is:

$$P(Deg = k) = e^{-p(N-1)} \frac{(p(N-1))^k}{k!} = e^{<k>} \frac{< k >^k}{k!}.$$

where $< k >= p(N-1)$ is the average degree of the network. More details can be found in [44, 5]. That means that most of the nodes have roughly the same number of links, close to the network's average degree $< k >$ $(= p(N-1))$ and that there are no (or very rare) nodes with significantly more or less links.

Scale-free networks

The problem is that random networks can not explain the topologies of real networks, because ”real life is not randomly regulated”. That is why we need to introduce another kind of networks: the scale-free networks. They are very important, because it has been observed that many networks are scale-free. The main characteristic of such networks is that the number a nodes with a given degree follows a power law. Therefore, the probability that a chosen node has exactly k links follows $P(k) \sim k^{-\gamma}$, where γ is the degree exponent, comprised between 2 and 3 for most of the networks. It results that such networks are characterized by the fact that most nodes have only a few links, and only a small number of nodes have many links. These nodes with many links are called hubs, and hold all the nodes of the network together. Furthermore, we can even say that it is very difficult to find a typical node which could be used to describe all the others, contrary to the random networks, where most of the nodes have roughly the same number of links.

We will keep considering that model because it has been found that most cellular networks are approximately scale-free. The nodes represent the metabolites, while the links represent the enzyme-catalysed biochemical reactions. Since these chemical reactions are irreversible, it is admitted that cellular networks are directed. Finally, to do the link between the cellular and the scale-free networks, it is essential to note most metabolic substrates participate in only one or two reactions only, whereas only a few substrates participate in loads of reactions as metabolic hubs. For example, genetic regulatory networks are thought to be scale-free, since most of the miRNAs regulate only a few genes, but some of them regulate many genes.

Some properties of scale-free networks

We can mention few properties of scale-free networks. First, the properties of growth and preferial attachment consist of the fact that networks are not fixed, that new nodes can join the existing network, and that these new nodes will tend to connect to hubs. These two characteristics are jointly the reasons of the emergence of the scale-free networks. Indeed, the nodes with many links, which are the oldest nodes to appear in the network, are more likely to get even more connections thanks to that ”rich-gets-richer” mechanism.

Then, modularity refers to a group of physically or functionally linked nodes that work together to achieve a distinct and precise function, as for instance in biology, protein-RNA complexes which are the core of many basic biological functions. Indeed, most of the cellular molecules are either part of an intracellular complex such as the ribosome, with a modular activity, either they contribute to a distinct process, in an extended module. When studying the modularity of a given network, we need to clearly identify the different modules, their relationships, how they interact...

Robustness is the system's ability to respond to changes in the external conditions and/or internal organization while maintaining normal behaviour. Scale-free networks are amazingly robust against accidental failures. Indeed, random failure will mainly affect the numerous small degree nodes, the absence of which does not affect the whole network's integrity. However, it is true that if some important hubs are affected, the network will be very vulnerable, not having the suitable behaviour, maybe leading to the collapse of the network.

We can mention many other properties of scale-free networks, as dissortativity, clustering, hierarchy, small-world effect, etc. Further information on all of this is available in [3].

After this presentation of networks, we are going to provide a brief review of statistical models which have been proposed in the similar topic of biological networks, one of them being the base of our proposed methodology.

2.2.4 Review of statistical models for biological networks

Bayesian graphical models have already been used to study miRNA targeting. Huang, Morris and Frey in [25], Huang, Frey and Morris in [23], proposed a Bayesian model for the regulatory process of targets and miRNAs. In these papers, the authors proposed a variational learning procedure, with a minimized Kullback-Leibler divergence and EM algorithm to predict a set of functional interactions. In this approach, the regression coefficients are assumed constant for each miRNA, meaning that one given miRNA will have the same regulatory effect on all its potential targets. In [58], Stingo et al. proposed a MCMC (Monte Carlo Markov Chain) approach with regression coefficients different for every single functional gene-miRNA pair, where each target gene is assumed independent of the others given the miRNAs. This paper was the starting point of our methodology, where we try to relax this major assumption made in the latter matter. We can also mention that in [59], Stingo and Vanucci proposed a variable selection with a Markov random field prior to infer undirected gene-gene networks, where the subjects are classified according to their phenotypes. In this approach, genes part of a same functional group are assumed correlated with a Inverse-Wishart prior, an assumption similar to the one we will present in the following chapter.

2.3 Conclusion

In this chapter, we have presented the most important biological knowledge we require to understand the project and study we want to perform, including both genetics and target recognition in one side, and some properties of graphs on the other side. Finally, we provided a small review of similar studies with statistical models which aim to infer biological networks. In the next chapter, we will describe in more details the study and the available data, before we explain in Chapter 4, the proposed framework to infer a gene-miRNA network.

CHAPTER 3

Presentation of the data and study

3.1 Introduction

Trying to understand how genes and proteins are regulated is one of the major tasks in genomics and biology, if not the most challenging. Regulation can happen at transcriptional, post-transcriptional, translation and post-translational levels. Transcription is the process in which segments of the DNA sequence are used as templates for the synthesis of shorter molecules called RNA. Then, in the translation process, these RNA molecules direct the synthesis of proteins. MicroRNAs (miRNAs) are endogenous short non-coding RNA molecules, 22 nucleotides long on average, which play an important regulatory role in living cells [11], as it has been estimated that at least 30% of the genes in the human genome are regulated by miRNAs [45]. Genes regulated by miRNAs are commonly called targets. The exact mechanism of miRNA regulation is still unclear and to be understood, thus consists of an important area of genomics research, so is the complete process of regulation. According to current knowledge, it is believed that miRNAs are post-transcriptional down-regulators, which bind to 3'-untranslated region (UTR) of its target mRNAs, leading to either the degradation of the mRNA, or the repression of the translation of proteins, depending on how extensive the base-pairing between the mature miRNA and the target mRNA is.

Several algorithms have already been developed to predict and determine potential miRNA-mRNA interactions, based on the sequence and structure characteristics of the miRNAs and their target sites. The main factors used by these algorithms are the sequence complementarity, hybridization energy and comparison across species. But they also often take into account different other factors that can influence the interactions, such as different seed matches complementarity, the conservation, thermodynamics stability, site accessibility. We can briefly mention some of the more widely used prediction algorithms: TargetScan [40, 39, 19, 13], miRanda [9, 27], DIANA-microT [30], PicTar [32], and PITA [29]. Some reviews of these methods and factors can be found in [4], [62] and [38]. Generally, these algorithms predict loads of potential miRNA-mRNA interactions, sometimes different ones as they use different factors to establish their targets. So it can quickly become too difficult for researchers to find, among these hundreds of thousands of potential interactions, those who are functional under particular clinical conditions and thus play a crucial regulatory role under these conditions.

Our aim is to develop a statistical model which identifies with high confidence a set of potential targets and functional interactions. The regulatory relationships can be inferred by integrating expression levels of both miRNAs and their candidate target genes. Our approach consists of a directed Bayesian graphical model, in which we also incorporate biological knowledge, such as the negative regression coefficients, as it is believed that miRNAs down-regulate the expression of the genes. We also take into account in our prior model the target scores that have been predicted by TargetScan [19] based on the sequence and structure information. We then perform MCMC (Monte Carlo Markov Chain) methodology to select the variables of interest from the association matrix, before creating the network.

Bayesian graphical models have been introduced to study the regulatory process of target genes by miRNAs by Huang, Morris and Frey [25] in 2007 and Huang, Frey and Morris [23] in 2008. In their approach, these authors conducted a variational learning method by minimizing the KL-divergence, where the regression coefficients were considered constant with respect to the miRNAs, meaning that one given miRNA will have the same regulatory affect on all the target mRNAs. This is the reason why in 2010, Stingo et al [58] proposed a full MCMC procedure which allows different regression coefficients for every candidate gene-miRNA pair, making the variable selection more effective. Our approach is similar to the latter, in the sense that we also want to predict gene-miRNA interactions. However, the important difference is that Stingo et al. [58] assume that the genes are independent given the miRNAs, while we are trying to relax this assumption. Indeed, in the living cells, groups of genes are often part of group responsible of a distinct biological function, and thus are likely to be correlated. However, due to the size of data and thus computational issues, we still have to assume some independence between genes, by separating the genes in clusters of correlated variables, the clusters being independent of each other. The idea of this thesis is to compare both approaches, and study the differences, the advantages and inconveniences of each method.

In this chapter, first we describe the clinical study and the data, consisting of expression levels of miRNAs and potential target genes, and the scores from the TargetScan algorithm corresponding to our set of miRNAs and mRNAs. Then we describe few approaches we used for the required dimensionality reduction, as well as a clustering method to group correlated genes together.

3.2 Clinical study

3.2.1 Acute Coronary Syndromes

Acute Coronary Syndromes (ACS) is a term which refers to several conditions attributed to the obstruction of the coronary arteries. The common problem of these conditions is the formation of a blood clot in a coronary artery, leading to a sudden reduction of blood flow to the heart muscle. A myocardial infarction (MI), also called heart attack, occurs when a coronary artery is blocked. The blood clot prevents the blood flow reaching the heart muscle, which is then at risk of dying if the blockage is not quickly undone, since the heart muscle beyond the blockage is strained of oxygen. The two main types of MI are ST elevation MI (STEMI) and non-ST elevation MI (NSTEMI). The difference, determined after an electrocardiograph, lies in the fact that in a NSTEMI, the artery which supplies one part of the heart muscle is partially blocked, not totally as it is in a STEMI. Apart from these two MI conditions, we also need to distinguish the less severe unstable angina (UA) condition, which occurs when the blood clot does not completely block the blood flow, and cell death is not seen. The blood flow is reduced but still effective, preventing the infarction of the heart muscle supplied by the affected artery. For further information or details about ACS, a well-explained review is available on the health and disease website http://www.patient.co.uk/health/acute-coronary-syndrome.

In this thesis, we consider a clinical study on ACS. Patients were admitted to hospital with acute chest pain. After the patients got provided a diagnostic of myocardial infarction, and after giving written informed consent, 30 patients with ACS were recruited less than 24 hours after their admission to hospital: 10 with unstable angina, 10 with STEMI and 10 with NSTEMI. Expression of miRNAs was quantified in individual patients by qRT-PCR at 7 and 30 days after the admission to hospital, and global gene expression was quantified in the same samples using Affymetrix human arrays. After RNA got extracted from these blood samples, we determine the expression levels of miRNAs and mRNAs, defined by the Ct number. The Ct number is the number of cycles which are necessary to reach a predetermined threshold level of log2-based fluorescence. These expression levels were quantified with Affymetrix genechip arrays and normalised using puma package for microarray data analysis. Since some patients did not come back after their first admission, and/or a few samples could not be exploited, the numbers of different samples are 19 for unstable angina, 19 for NSTEMI and 16 for STEMI.

Also, due to the lack of patients and time points, each different sample was treated as a single biological sample.

3.2.2 MiRNA profiles

MiRNA data was analysed using Sequence Detection System and DataAssist (Applied Biosystems). The expression levels were estimated based on the comparative threshold cycle (Ct) method.

As a regulatory network can quickly become complex due to the size of the data, and also in our methodology to the model, we need to focus on small sets of miRNAs and mRNAs. That is why, after personal communication with collaborators from the Department of Cardiovascular Science who work on a similar topic, we decided to focus on a set of 13 miRNAs, which are believed to have a crucial role in the regulation process of genes responsible of ACS. A previous study, using the same type of data, has indeed provided evidence of changes in the expression of miRNAs after a myocardial infection, indicating the role of miRNAs as biomarkers for risk estimation, classification of disease and therapeutic interventions. Further information can be found in [51].

The levels of these 13 miRNAs, under each condition and without regarding the disease condition in the last panel, can be visualized in the following histograms in Figure (3.1).

3.2.3 TargetScan scores

As we mentioned and we will detail in Section 3, we want to integrate prediction scores in our prior model. The scores we use were computed from

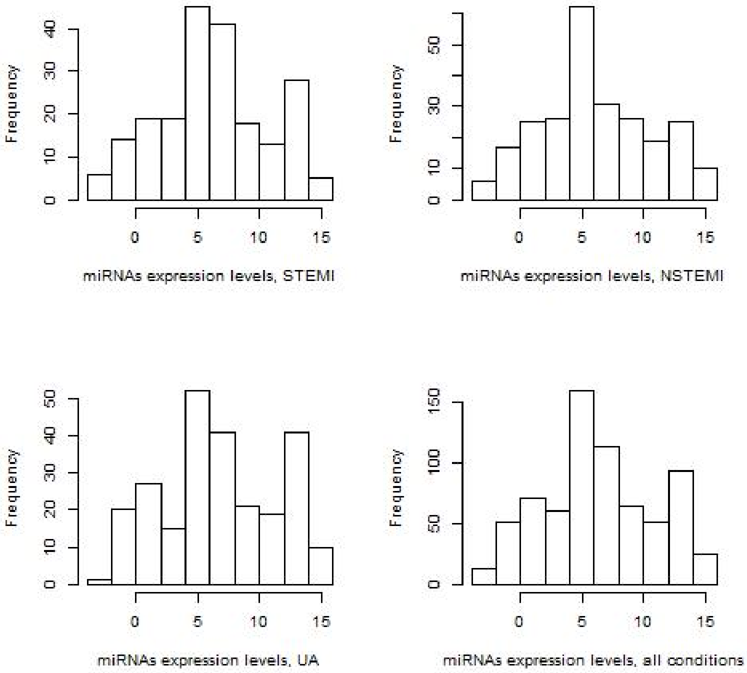

Figure 3.1: Histograms of miRNAs expression levels, patients suffering from STEMI (top-left), NSTEMI (top-right), Unstable Angina (bottom-left), and all levels for the three conditions (bottom-right)

the TargetScan algorithm [19], one of the most widely used target prediction algorithm. In this release of the algorithm, a score, called the total context score, for a specific miRNA-target gene is computed, being the sum of the contribution of four features: the site-type contribution, the 3' pairing contribution, the local AU contribution, and the position contribution. The site-type contribution represents the type of seed match (8mer, 7mer-m8, 7mer-1A or 6mer, the types described in Figure 2.5), while the 3' pairing contribution reflects any consequential complementarity outside the seed region, in particular between nucleotides 12 and 17 (like the atypical matches described in Figure 2.6). The local AU contribution refers to the concentration of A and U nucleotides flanking the corresponding seed region of the miRNA, as it is believed that the match in the 3' end of the mRNA is more likely to accur in a rich AU context (complementarity between A and U) [19]. Finally, the position contribution analyzes the position of the target site within the mRNA. For all these features, a more negative score is associated with a more favorable site, and the contect score is the sum of these four scores. More information can be found in the original paper [19], and in other publications from other releases of the algorithm [40, 39, 13]. These scores can be downloaded from the TargetScan website: http://www.targetscan.org/. If a given miRNA-mRNA interaction does not have a score, it indicates that the algorithm did not predict any regulatory association between the miRNA and mRNA of interest.

In this study, we consider only one source of target scores. However, as there exist other sources, it is possible to use another source, and also to combine different sources for our prior model, as we will see in Section (4.2.3).

3.2.4 Target genes profiles

Gene expression was estimated using probabilistic models implemented in puma package [43, 46]. These models provide estimates for the variance and credibility interval for probe level errors of each transcript and generate accurate estimates of low gene expression. Further information can be found in the original papers.

RNA was extracted from each sample to quantify the expression level of the genes. After selecting the unique probe sets among our set of genes, the Gene Symbols of these mRNAs were linked to the potential targets of the 13 miRNAs we previously selected. Genes were conserved in the analysis only if the TargetScan algorithm predicted at least one potential regulatory miRNA among our set. This is done in order to reduce the complexity of our approach, to reduce our number of target genes. Indeed, the initial number of target genes, few thousands, was too high and not reasonable in order to conduct an efficient analysis. This the reason why we apply few methods, described in Section 3.3.1, to narrow down the number of genes to 897. The expression levels of these 897 target genes are shown in Figure 3.2, in the same order as the miRNAs expression levels.

As we will see in Section 4.2.1, we assume in our model that the expression levels data, composed of both genes and miRNAs, are normally distributed. To validate the approach, we can check this assumption is reasonable, by looking at the distribution of all the data, under each condition, in Figure 3.3. However, since we may observe some slight skewness, we can look at the QQ-plots of the data, under each condition, in Figure 3.4 to see that the data can be assumed to be normal. Also, by computing marginal Shapiro tests on each of the different variables, we find that the average p-value is $\simeq 0.37$.

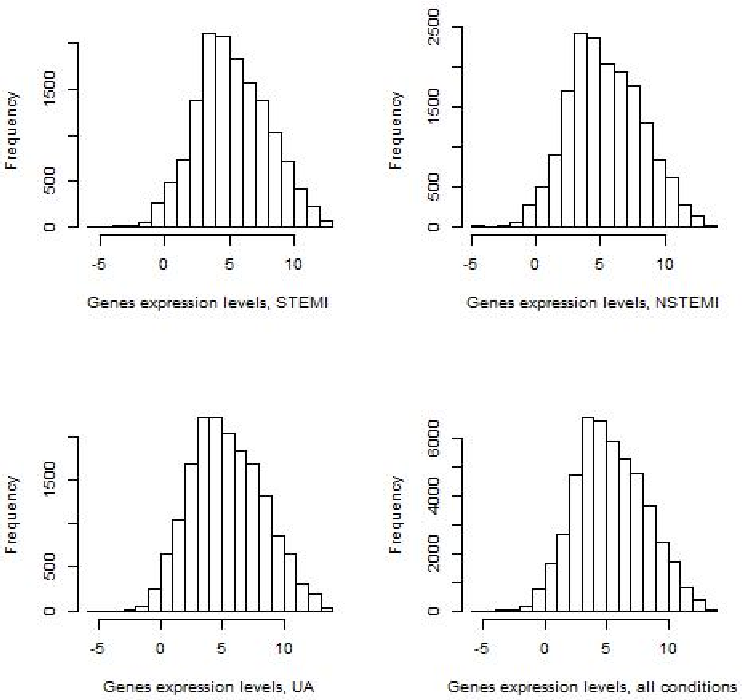

Figure 3.2: Histograms of genes expression levels, patients suffering from STEMI (top-left), NSTEMI (top-right), Unstable Angina (bottom-left), and all levels for the three conditions (bottom-right)

3.3 Reducing the dimensionality

3.3.1 Reducing the number of target genes

As mentioned, the number of genes is very large. As an illustration, we initially had available expression levels of over 54,000 transcripts with 38,500 different probes. This high number number of variables would definitely lead to computational issues: length, cost, feasibility etc. This is particularly true when it comes to compute the multivariate cumulative density function (cdf) $\Phi_{G*M}(\mathbf{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U})$ from Equation (4.12) in Section 4.3.1, and the inversion

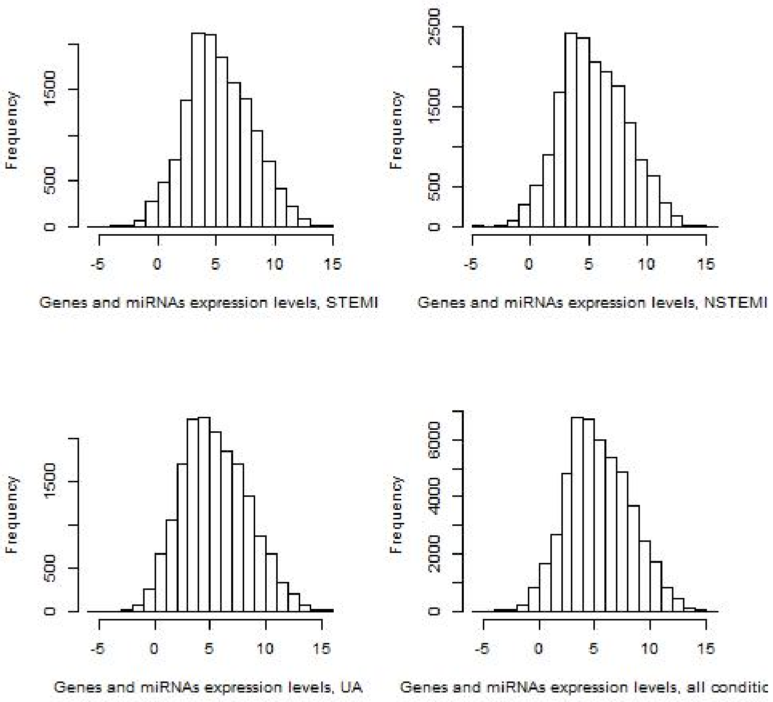

Figure 3.3: Histograms of genes and miRNAs expression levels, patients suffering from STEMI (top-left), NSTEMI (top-right), Unstable Angina (bottom-left), and all levels for the three conditions (bottom-right)

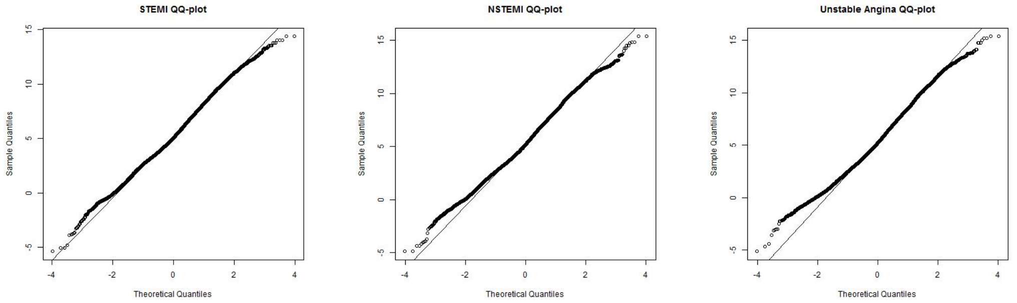

Figure 3.4: QQ-plots for miRNAs and genes datasets, under each condition: STEMI, NSTEMI, and Unstable Angina

of the covariance matrix $\mathbf{\Sigma}$. This is why the genes first need to be filtered before we introduce the statistical model for infering the regulatory network in Chapter 4.

We first select the genes transcripts which are unique probe sets. A probe set is a sequence of DNA used to detect the presence of a complementary sequence by binding (hybridizing) to that site. There exists different types of probes sets, which depend on the specificities of the probe set. A unique probe set is designed to detect a unique sequence of a single gene, while other types can detect different sequences of one gene, or two different sequences of two different genes, or one sequence that can be detected by several other probe sets. As unique probe sets are characteristic of a specific gene, we choose to conserve only these variables. Further information about probe sets can be found on the Affymetrix website http://www.affymetrix.com/support/help/faqs/mouse_430/faq_8.jsp.This selection of unique probe sets reduced the number of potential genes to 8,225.

Then, as mentioned in the description of the data, we conserve the genes in the analysis only if they had at least one potential regulating miRNA with a positive TargetScan score, the other genes being disregarded. Indeed, if it is biologically not possible for any miRNA in our dataset to bind to a given gene based on their sequence information, it is not relevant to study a possible regulatory association involving that given gene. Once this second filtering process done, it remains 4,215 potential targets out of the previous 8,225.

Thirdly, we assume that targets which have an impact on different conditions are the ones with a different behaviour given a given specific condition. Thus, comparing the expression levels of patients suffering from STEMI (most severe condition) and UA (less severe condition), we conserve the genes which show a significant different behaviour, either an increase or a decrease of their expression level. This last pre-processing step allows us to disregard more than 3,000 other genes.

Finally, these three different filters allow us to narrow down the number of our variables to $G = 897$ target genes.

3.3.2 Clustering

The high dimension G of the genes can still introduce computational problems, in particular regarding the inversion of covariance matrices, leading to long running times and/or crashes. A possible approach to reduce the complexity is to define clusters of more reasonable size, composed of correlated genes. This classification might be suitable, especially when genes have different patterns. For example, in Figure 3.5, by checking the expression levels of a subset of genes, we distinctly observe two groups of genes, with different average levels. It would then be suitable to divide this subset in two smaller clusters, where the genes within the same cluster are assumed correlated. On the other hand, genes belonging to different clusters will be assumed independent.

With this approach, we can divide our set of data into several clusters of correlated genes, then assume that these clusters are independent of each other. In this case the covariance matrix of the genes is block diagonal:

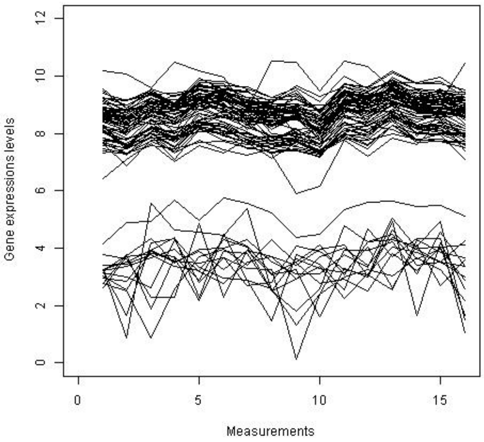

Figure 3.5: Subset of genes composed of two different clusters with different patterns: one cluster with an average expression level close to 8, a second one with an average close to 4.

$$\boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\Sigma}_C \end{pmatrix},$$

where C is the number of different clusters. However, it is likely that several genes will not fit in any cluster, or many clusters will be very small, composed of a few genes. All these genes can then be assumed independent and grouped together in the last cluster, with a diagonal covariance matrix $\boldsymbol{\Sigma}_C$, resulting in $C - 1$ clusters of correlated variables. Our approach can still be applied, however, the proposed covariance matrix sampled from the Inverse-Wishart distribution would be non-diagonal, indicating correlations between the variables, which is not the case for that cluster. Therefore, it is more suitable to apply the approach from Stingo et al. [58]. Once the clustering is performed, it is then possible to run each cluster independently.

The clustering is independent from the model, then any suitable clustering approach can be performed. For this study, we choose to use the AutoSOME algorithm, proposed by Newman and Cooper [42], where they described a Self-Organizing Map (SOM) which can identify up-regulated genes associated within the same biological function by analysing the co-expression of the data. However, other methods have been developped to cluster mircroarray data. A first approach uses hierarchical clustering [16, 21], to build a dendrogram of clusters and sub-clusters, the number of clusters which can vary between 1 and the number of data point (between all data in one cluster and each variable in its own cluster). A second approach consists of K-means clustering [16, 21], which separates the dataset by minimizing the statistical variance within k clusters. Another method is the non-Negative matrix Factorization [12], which efficiently identify well-defined clusters. However, these three methods share a common inconvenience, in the sense they need a priori knowledge or an external method to predict the number of clusters.

A machine learning method used for high-dimensional data is the Self-Organizing Map (SOM) [31, 61]. To identify k clusters, it randomly initializes a lattice of k nodes, and then data points iteratively move similar points towards each other and move similar points away from each other. However, the recurrent inconvenience of the a priori knowledge of the number of clusters still remains.

AutoSOME is a SOM-based method. It uses its advantages for dimensionality reduction and spatial organization of large datasets. After that initial organization of the input data, it computes an error surface which measure the similarity between nodes. A high error represents a high dissimilarity between adjacent nodes, while a low error represents a high similarity between nodes which are likely to be part of the same cluster. A density-equalization procedure then forces highly similar data to aggregate, while separating the dissimilar variables. Discrete clusters are then identified from this resulting SOM using the minimum spanning tree from graph theory. Using Monte Carlo sampling, only the relationships between variables which reach a specified threshold are conserved, ensuring the statistical significance of the clustering identification. To avoid the recurrent issue of output variation in clustering methods, by merging several iterations of the procedure, the method uses an ensemble strategy to send uncertain data points, whose clusters may vary among the scheme iterations, to the clusters they are most frequently part of.

The main advantage of that method is that it does not need prior knowledge of cluster number, however, parameters can be modified to obtain clusters of desired size for example. More details can be found in the original paper [42].

The choice of the clustering method is not actually crucial. It is fair to say that our approach depends on the clustering method, as another method would produce different clusters of genes. However, it does not rely on that choice, as the inference can then still be performed on each cluster. That is the reason why it is absolutely possible to chose another clustering method if it is more appropriate.

3.4 Conclusion

In this chapter, we described the study and the data, as well as the process which allowed us to reduce the number of genes in the study. In a first section, a description of the different diseases was provided, underlying the differences between MI and unstable angina, and alo between STEMI and NSTEMI, the two types of MI. Then, a description of the data collected is given, explaining how and when the data were collected, how many patients were recruited per condition, and which techniques were used to quantify the expression levels. Also, the target scores from TargetScan were introduced, as we will use these scores as prior information in our approach, as we can see in Section [**?**]. Finally, we described the three steps we performed to reduce the number of potential target genes, as dimensionality can quickly become, so we had to select the most likely genes. That is the reason why we first chose the transcripts characteristic of a gene (unique probe sets), before selecting only the ones which were among potential targets (via the target scores), before selecting the genes which showed a significant change in their expression level within the different ACS conditions.

Now that we have a final dataset, we can focus on the statistical model and framework we will present in details In Chapter 4, in order to try to answer the main investigation: can we estimate an ACS regulatory network between miRNAs and genes? To do so, we want to study if it is possible to infer these functional gene-miRNA interactions with high confidence so it can become possible to identify such biological interactions as the potential source of a genetic condition.

CHAPTER 4

Bayesian Graphical Modeling

4.1 Introduction to graphical models

Graphical models have been introduced and developed around the end of the 20th century, by Dawid and Lauritzen in [8] and by Lauritzen in [35], before Jensen introduced Bayesian networks in 1996 in [26]. Bayesian graphical models have then been used to study the regulatory process of target genes by miRNAs by Huang, Morris and Frey [25] in 2007 and Huang, Frey and Morris [23] in 2008. In their approach, the authors conducted a variational learning method by minimizing the KL-divergence, where the regression coefficients were considered constant with respect to the miRNAs, meaning that one given miRNA will have the same regulatory effect on all the target mRNAs. This is why in 2010, Stingo et al [58] proposed a full MCMC approach which allows different regression coefficients for every candidate gene-miRNA pair, making the variable selection more effective. Our approach is very similar to the latter, the main difference being that Stingo et al. [58] assume that the genes are independent given the miRNAs, assumption we are trying to relax, as in the living cells, groups of genes are often part of group responsible of a distinct biological function. However, due to the size of data and thus computational issues, we still have to assume some independence between genes, by separating the genes in clusters of correlated variables, the clusters being independent of each other. This clustering approach, AUTOSOM [42], is described in more details in Section 3.3.2. The idea of this thesis is to compare both approaches, and study the differences, the advantages and inconveniences of each method.

In this chapter, we will define the framework of the proposed methodology, describing the assumptions and choices of priors. We will then present the estimation procedure via a MCMC algorithm. After setting up the parameters, we will apply the approach first to case study of simulated data, where we will compare in details its performance with the performance of the algorithm where the genes are assumed independent. Finally, we will perform a Monte Carlo study of 100 simulated cases, which will allow us to have a better idea of the general performance of our methodology.

4.2 Model

4.2.1 Framework

Having expression levels of miRNAs and potential targets, we aim to build a directed graphical model which allows us to identify a small number of regulatory associations, in order to be able to answer the underlying question: "which miRNAs regulate which genes". The inference of the regulatory network is performed by integrating the expression profiles and the sequence information of our variables in the prior probability model. The model needs to be adapted to the data, but also to biological considerations: concept of sparsity, down-regulation of the genes by the miRNAS, etc.

Graphical models are graphs or networks, where the random variables are represented by nodes, and the interactions between variables by arrows or edges. The absence of arrow between two variables mean that the interaction of interest is not functional, and the two variables are then conditionally independent given the other variables. Graphs can be undirected, when the dependency between variable is symmetric, or directed when there is only one direction of dependence. A graphical example of a directed network between three miRNAs and seven genes is given in Figure 4.1. In our approach, we consider a directed graph, due to the fact that it is the miRNAs which regulate the targets.

In a directed graph, we need to order our variables, in order to define the conditional independences, such that a target gene can only be regulated by the miRNAs and that a miRNA can only regulate the genes. We then define $\boldsymbol{Z} = (\boldsymbol{Y}, \boldsymbol{X})$, where $\boldsymbol{Y} = (\boldsymbol{Y}_1, \dots, \boldsymbol{Y}_G)$ and $\boldsymbol{X} = (\boldsymbol{X}_1, \dots, \boldsymbol{X}_M)$, are

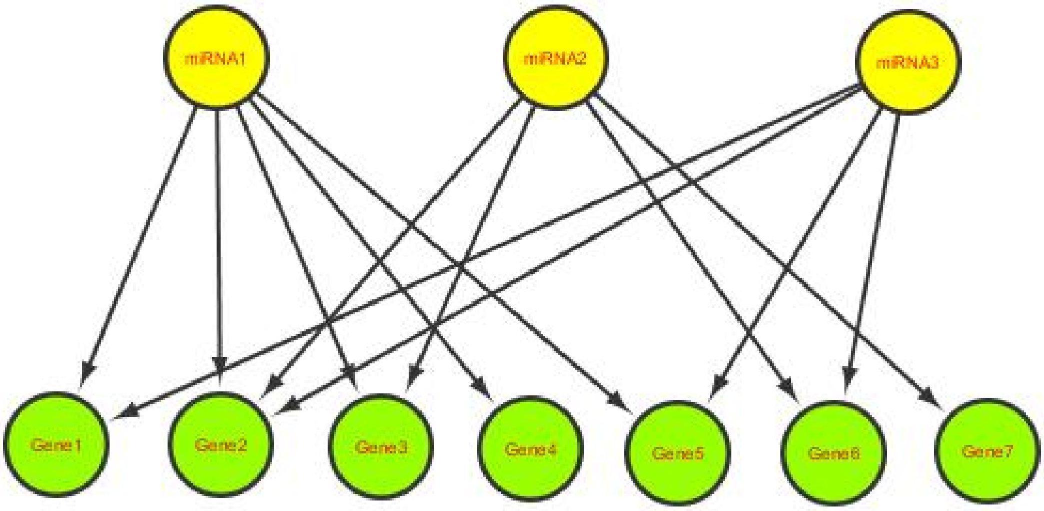

Figure 4.1: A graphical representation of regulatory network between 3 miRNAs and 7 genes.

the matrices representing respectively the genes and miRNAs profiles, G and M being the respective numbers of genes and miRNAs included in the study. Specifically, for each target profile $\boldsymbol{Y}_g$ (respectively miRNA profile $\boldsymbol{X}_m$), y_{ng} (x_{nm}) represent the expression level of gene g (miRNA m) in the nth sample, $n = 1, \ldots, N$. In our study, $G = 897, M = 13$, and we have $N = 16$ for the STEMI condition, $N = 19$ for the NSTEMI condition and $N = 19$ for the unstable angina condition.

Our first assumption is that $\boldsymbol{Z}$ follows a matrix-variate normal distribution (Section A.2), with zero mean, among-row covariance matrix $\boldsymbol{I}_N$, and among-column covariance matrix $\boldsymbol{\Omega}$:

$$\boldsymbol{Z} \sim N_{N,G+M}(\mathbf{0}, \boldsymbol{I}_N, \boldsymbol{\Omega}).$$

Due to the ordering of variables, $\boldsymbol{\Omega}$ can be partitioned in blocks as follows:

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{\Omega}_{YY} & \boldsymbol{\Omega}_{YX} \\ \boldsymbol{\Omega}_{XY} & \boldsymbol{\Omega}_{XX} \end{pmatrix},$$

where $\boldsymbol{\Omega}_{YY}$ and $\boldsymbol{\Omega}_{XX}$ can be seen as the marginal covariance matrices of the genes and the miRNAs respectively, and $\boldsymbol{\Omega}_{YX}$ the $G \times M$ matrix containing the covariances between each gene and miRNA.
This ordering also allows us to factorize the likelihood of $\boldsymbol{Z}$ as:

$$f(\boldsymbol{Z}) = f(\boldsymbol{Y}|\boldsymbol{X})f(\boldsymbol{X}) = f(\boldsymbol{Y}|\boldsymbol{X}) \prod_{m=1}^{M} f(\boldsymbol{X}_m), \tag{4.1}$$

where $\boldsymbol{Y}|\boldsymbol{X} \sim N(\boldsymbol{X}\boldsymbol{\beta}^T, \boldsymbol{I}_N, \boldsymbol{\Sigma})$, and $\boldsymbol{X}_m \sim N(\mathbf{0}, \sigma_m I_N)$, assuming the miRNAs are independent, without losing generality of the gene-miRNA dependencies, as we are not interested in this thesis in the possible relationships between miRNAs. Here, σ_m represents the variance of miRNA m, $\boldsymbol{\Sigma}$ the genes covariance matrix, and $\boldsymbol{\beta} = \{\beta_{gm}\}$, each β_{gm} being the regression coefficient between gene g and miRNA m.

As it is believed that miRNAs down-regulate the expression of their gene targets, it seems legitimate that this knowledge should be included in our statistical model. This is why we impose the regression coefficients to be negative, now leading to $\boldsymbol{Y}|\boldsymbol{X} \sim N(-\boldsymbol{X}\boldsymbol{\beta}^T, \boldsymbol{I}_N, \boldsymbol{\Sigma})$:

$$\boldsymbol{Y} = -\boldsymbol{X}\boldsymbol{\beta}^T + \boldsymbol{\epsilon}, \tag{4.2}$$

where $\boldsymbol{\epsilon}$ follows a matrix-variate normal distribution, with zero mean, among-row covariance matrix $\boldsymbol{I}_N$, and among-column covariance matrix $\boldsymbol{\Sigma}$. We then complete the biological constraint by imposing a gamma prior for each of our regression coefficients, $\beta_{gm}|\gamma_g \sim Ga(1, c\gamma_g)$, where γ_g follows an inverse-gamma distribution, $\gamma_g \sim IGa((\delta+M)/2, 2/d)$. We then define $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_G)$. These parameters γ_g represent the variances of the genes in the model of Stingo et al., [58]. In these distributions, c represents a correction factor, δ the minimum integer such that the expected value of this distribution exists, and d is set up such that the mean is comparable to the error variance. More details will be provided in Section 4.5, where we set up the different hyperparameters of the model. In our approach, we want to study the effect of the covariance matrix on the regulatory network, however we still need to conserve and update those parameters due to their role in the prior for the regression coefficients. Finally, we impose an Inverse-Wishart prior for the covariance matrix $\boldsymbol{\Sigma}$, $\boldsymbol{\Sigma} \sim IW_{\nu_\Sigma}\Big(\nu_\Sigma, (\nu_\Sigma + G + 1)\boldsymbol{I}_G\Big)$, with $\nu_\Sigma = G + 2$, the smaller value such that the expected value of $\boldsymbol{\Sigma}$ exists. Definitions and notations of the different distributions can be found in Section A.2.

4.2.2 Selection of regressors under the regulatory network

The aim of the study is to predict for each gene, a subset of miRNAs which regulate that gene, which is equivalent to determine for each gene g, the β_{gm} which are not zero. These non-zero regression coefficients would then indicate the functional relationships between the corresponding genes and miRNAs. This variable selection problem can be computed by introducing a $(G \times M)$ association matrix $\boldsymbol{R} = \{r_{gm}\}$, with its elements being Bernoulli variables. More precisely, $r_{gm} = 1$ if there is an interaction between the gene and miRNA of interest, or zero otherwise:

$$r_{gm} = \begin{cases} 1 & \text{if the m-th miRNA regulates the g-th gene,} \\ 0 & \text{otherwise.} \end{cases}$$

Given the association table $\boldsymbol{R}$, the regression coefficients β_{gm} are then stochastically independent, thus we can modify our prior and impose the following mixture prior distribution:

$$\pi(\beta_{gm}|\boldsymbol{R}, \gamma_g) = r_{gm} Ga(1, c\gamma_g) + (1 - r_{gm}) I_{\beta_{gm}=0}. \tag{4.3}$$

Then β_{gm} still follows a gamma distribution if the given interaction is functional, but is zero otherwise. We can check from this prior that $r_{gm} = 0$ if and only if $\beta_{gm} = 0$.

Similarly, we can now modify our prior for the parameters γ_g's and say that under the regulatory network, $\gamma_g \sim IGa\big((\delta + k_g)/2, 2/d)\big)$, where k_g is the number of $\beta_{gm} \neq 0$ for a given gene g, which is equivalent to the number of miRNAs included in the regulation of each target g.

4.2.3 Target scores in the prior model

As previously described, we aim to estimate prior probabilities of functional miRNA-gene interactions, via the target scores obtained from sequence/structure information. These scores need to be positive, or zero when it is believed that there is no association between a given miRNA and a given target. Given these scores, prior probabilities of the variables r_{gm}, indicating miRNA m regulating gene g, can be computed via a logistic model:

$$P(r_{gm} = 1|\tau) = \frac{\exp(\eta + \tau s_{gm})}{1 + \exp(\eta + \tau s_{gm})}, \tag{4.4}$$

where s_{gm} is the score of possible association between miRNA m and gene g obtained from sequence information, and where τ is an unknown parameter, following as hyperprior a gamma distribution, $\tau \sim Ga(a_\tau, b_\tau)$. We can check that higher scores, which represent more likely regulating associations, lead to higher prior probabilities of association. However, as the scores we downloaded from the 2003 TargetScan release [19] get more negative and lower when the binding between a miRNA and its target gets more likely, the scores have to be transformed in a suitable way that conserves the ranking of the more likely associations. Our choice of transformation will be discussed in Section 4.5, where we set up the various parameters of the model.

The parameter η gives the prior belief of interaction, especially if we do not have access to the sequence and structure knowledge of the s_{gm}'s, assuming all the scores are equal to 0. Since it is believed that genes are on average regulated by one or two miRNAs, we set η close to $\log(\frac{1}{M-1})$, which gives a prior expected number of regulating miRNAs close to 1 per gene. If we have more information available about the target scores, from one or more sources, we can assume that the prior probability increases. Thus, the parameters a_τ and b_τ are set up in such a way that the prior number of regulating miRNAs per gene is close to 2. Further details will be given in 4.5, after we explained the transformation applied to the target scores.

In the case where more than one source of information is available, as it is in the study by Stingo et al [58], it is sensible to integrate these different sources and thus incorporate the different scores in the prior model. If for example, we have three different sources from three algorithms (TargetScan, miRanda, PicTar for instance), the logistic model (4.4) can be updated as:

$$P(r_{gm} = 1|\tau) = \frac{\exp(\eta + \tau_1 s_{gm,1} + \tau_2 s_{gm,2} + \tau_3 s_{gm,3})}{1 + \exp(\eta + \tau_1 s_{gm,1} + \tau_2 s_{gm,2} + \tau_3 s_{gm,3})}, \tag{4.5}$$

where $s_{gm,i}, i = 1, 2, 3$, represent the scores from TargetScan, miRanda and PicTar respectively, with respective hyperprior parameters $\tau_i, i = 1, 2, 3$, and $\tau = (\tau_1, \tau_2, \tau_3)$.

4.2.4 Graphical representation

The proposed graphical model can then be defined as

$$\begin{aligned} f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \tau, \boldsymbol{\beta}, \boldsymbol{\gamma}) \quad \propto \quad & f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \tau, \boldsymbol{\beta}, \boldsymbol{\gamma}) \times \pi(\boldsymbol{\beta}|\boldsymbol{R}, \boldsymbol{\gamma}) \\ & \times \pi(\boldsymbol{\Sigma}) \times \pi(\boldsymbol{\gamma}) \times \pi(\boldsymbol{R}|\tau) \times \pi(\tau), \end{aligned} \tag{4.6}$$

and a graphical representation of its structure is given in Figure 4.2, where the directed arrows indicate the dependencies between the parameters, rep-

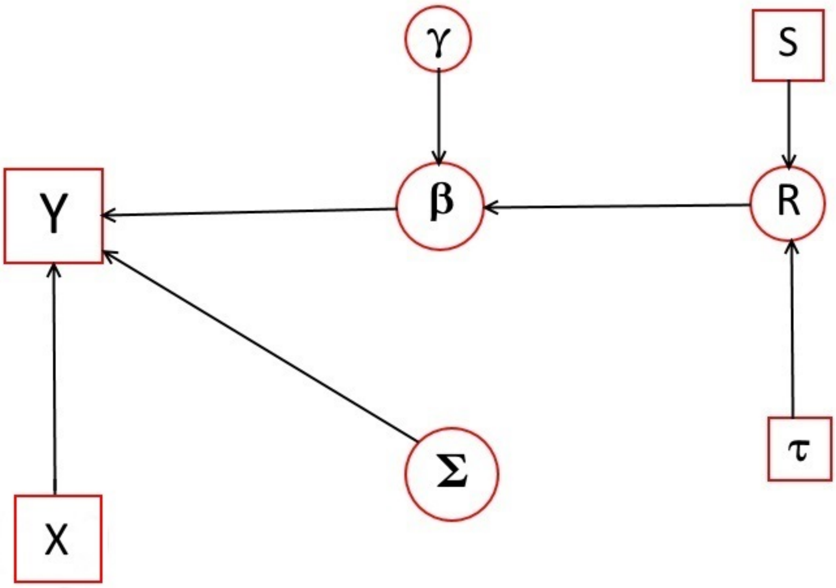

Figure 4.2: A graphical representation of the model, with the dependencies between the parameters and variables within the model. The observed random variables are represented by squares, the parameters by circles.

resented by squares, and the variables, represented by circles. The logarithm of this density will be referred as the log-probability of the model, as we will study its convergence in later sections when we apply the proposed approach.

4.3 Estimation procedure

4.3.1 Posterior inference

The main goal of the thesis is the estimation of the association matrix $\boldsymbol{R}$, in particular the posterior probability $P(\boldsymbol{R}|\boldsymbol{Y},\boldsymbol{X})$, or the marginal posterior probabilities of each single interaction $P(r_{gm}|\boldsymbol{Y},\boldsymbol{X})$. We use a Metropolis-Hastings within Gibbs procedure to perform a Stochastic Search Variable Selection method and to identify the most influential miRNA-target relationships. Such a method allows us to spend more time in the most likely configurations, which are the ones with higher marginal probabilities of $r_{gm} = 1$, which is necessary due to the size and complexity of our model space.

Actually, the complexity of the model, combined with the fact that the covariance-matrix is not diagonal, implies that the contribution of each regression coefficient toward the posterior probability of the entire model is very small. Thus, it is likely that many models have the same very small posterior probability. That is the reason why we choose to perform the posterior inference on the single marginal posterior probabilities of the presence of association $P(r_{gm} = 1|\boldsymbol{Y},\boldsymbol{X})$. These will be estimated directly from the output of the analysis, by computing the proportion of MCMC iterations for which we have $r_{gm} = 1$.

In order to estimate the marginal posterior distribution of $\boldsymbol{R}$, $P(\boldsymbol{R}|\boldsymbol{Y},\boldsymbol{X})$, we need to compute the likelihood $f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\gamma)$ by integrating out $\boldsymbol{\beta}$:

$$f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}) = \int f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\beta},\gamma)\pi\beta d\boldsymbol{\beta} \tag{4.7}$$

To complete this task, it is sensible to vectorize the random and parameter matrices. The relationship between the matrix-variate normal distribution and the multivariate distribution suggests that $(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \sim N(-\boldsymbol{X}\boldsymbol{\beta}^T, \boldsymbol{I}_N, \boldsymbol{\Sigma})$ is equivalent to:

$$f(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \sim N(\boldsymbol{A}\boldsymbol{B}, \boldsymbol{C}), \tag{4.8}$$

with $\boldsymbol{W} = \text{vec}(\boldsymbol{Y})$, $\boldsymbol{A} = -\boldsymbol{I}_G \otimes \boldsymbol{X}$, $\boldsymbol{B} = \text{vec}(\boldsymbol{\beta}^T)$, $\boldsymbol{C} = \boldsymbol{\Sigma} \otimes \boldsymbol{I}_N$, which can be written as:

$$\boldsymbol{W} = \boldsymbol{A}\boldsymbol{B} + \boldsymbol{\epsilon}_2, \tag{4.9}$$

where $\boldsymbol{\epsilon}_2$ follows a multivariate normal distribution, with zero mean and covariance matrix $\boldsymbol{C}$, and where $\otimes$ denotes the Kronecker product, defined in Section A.1.

Conditionally upon $\boldsymbol{R}$, the columns of $\boldsymbol{A}\boldsymbol{B}$ which corresponds to $r_{gm} = 0$ are also zero vectors. We can then select only the columns corresponding to the regressors included in the regulatory network:

$$\boldsymbol{W} = \boldsymbol{A}_R\boldsymbol{B}_R + \boldsymbol{\epsilon}_2, \tag{4.10}$$

where $\boldsymbol{A}_R$ and $\boldsymbol{B}_R$ are respectively the matrix and vector formed by taking only nonzeros columns and elements of $\boldsymbol{A}$ and $\boldsymbol{B}$.

As the coefficients $\boldsymbol{\beta}_{gm}$'s are stochastically independent given $\boldsymbol{R}$, we can write that:

$$\begin{aligned} \pi(\boldsymbol{\beta}|\boldsymbol{R}, \boldsymbol{\gamma}) = \pi(\boldsymbol{B}|\boldsymbol{R}, \boldsymbol{\gamma}) &= \textstyle\prod_{g=1}^{G} \prod_{m=1}^{M} \pi(\beta_{gm}|\gamma_g, \boldsymbol{R}) \\ &= \textstyle\prod_{g=1}^{G} (\frac{1}{c\gamma_g})^{k_g} \exp\big(-\boldsymbol{D}\boldsymbol{B}\big), \end{aligned} \tag{4.11}$$

where for each g, k_g is the number of $\beta_{gm} \neq 0$, and $\boldsymbol{D}$ is the $1 \times GM$ vector, composed of

$$\boldsymbol{D}_{g*m} = \begin{cases} 1/(c\gamma_g) & \text{if } r_{gm} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

We also define $K = \sum_g^G k_g$ the total number of functional associations of our network.

We will prove that the likelihood (4.7) is:

$$\begin{aligned} f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma}) &= f(\boldsymbol{W}|\boldsymbol{A}_R, \boldsymbol{\Sigma}, \boldsymbol{R}, \boldsymbol{\gamma}) = \textstyle\int f(\boldsymbol{W}|\boldsymbol{A}_R, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{B}_R, \boldsymbol{\gamma}) \\ &\quad \times \pi(\boldsymbol{B}_R|\boldsymbol{R}, \boldsymbol{\gamma}) d\boldsymbol{B}_R \\ &\propto \textstyle\prod_{g=1}^G (\frac{1}{c\gamma_g})^{k_g} |\boldsymbol{C}|^{-1} |\boldsymbol{U}|^{1/2} \exp(-\frac{1}{2}\boldsymbol{Q}) \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}) \\ &\propto \textstyle\prod_{g=1}^G (\frac{1}{c\gamma_g})^{k_g} |\boldsymbol{\Sigma}|^{\frac{K-N}{2}} \exp(-\frac{1}{2}\boldsymbol{Q}) \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}), \end{aligned} \tag{4.12}$$

where $\boldsymbol{U} = (\boldsymbol{A}_R^T \boldsymbol{C}^{-1} \boldsymbol{A}_R)^{-1} = \Sigma \otimes (\boldsymbol{X}^T \boldsymbol{X})^{-1}, \boldsymbol{V} = \boldsymbol{W}^T \boldsymbol{C}^{-1} \boldsymbol{A}_R + \boldsymbol{D}, \boldsymbol{Q} = \boldsymbol{W}^T \boldsymbol{C}^{-1} \boldsymbol{W} - \boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T$, and $\Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U})$ is the cumulative density function of a normal multivariate $(G * M)$ distribution, calculated at the zero vector, with mean $-\boldsymbol{U}\boldsymbol{V}^T$ and covariance matrix $\boldsymbol{U}$.

Proof. Given the mixture prior distribution for the $\boldsymbol{\beta}_{gm}$'s, $\pi(\beta_{gm}|\boldsymbol{R}, \gamma_g) = r_{gm} Ga(1, c\gamma_g) + (1 - r_{gm}) I_{\beta_{gm}=0}$, and the fact that given $\boldsymbol{R}$, those regression coefficients are stochastically independent, the prior for $\boldsymbol{\beta}$, and thus for $\boldsymbol{B}_R$, can be derived as:

$$\begin{aligned}
\pi(\boldsymbol{\beta}|\boldsymbol{R},\boldsymbol{\gamma}) &= \textstyle\prod_{g=1}^{G}\prod_{m=1}^{M}\pi(\beta_{gm}|\gamma_g,\boldsymbol{R})\\
&= \textstyle\prod_{g=1}^{G}\Big[\prod_{m=1,r_{gm}=1}^{k_g}\frac{1}{c\gamma_g}\exp(-\frac{\beta_{gm}}{c\gamma_g})\prod_{m=1,r_{gm}=0}^{M-k_g}1\Big]\\
&= \textstyle\prod_{g=1}^{G}(\frac{1}{c\gamma_g})^{k_g}\prod_{m=1,r_{gm}=1}^{k_g}\exp(-\frac{\beta_{gm}}{c\gamma_g})\\
&= \textstyle\prod_{g=1}^{G}(\frac{1}{c\gamma_g})^{k_g}\exp\big(-\frac{1}{c\gamma_g}\sum_{m=1,r_{gm}=1}^{k_g}\beta_{gm}\big)\\
&= \textstyle\prod_{g=1}^{G}(\frac{1}{c\gamma_g})^{k_g}\prod_{g=1}^{G}\exp\big(-\frac{1}{c\gamma_g}\sum_{m=1,r_{gm}=1}^{k_g}\beta_{gm}\big)\\
&= \textstyle\prod_{g=1}^{G}(\frac{1}{c\gamma_g})^{k_g}\exp\big(-\sum_{g=1}^{G}\sum_{m=1,r_{gm}=1}^{k_g}\frac{1}{\gamma_g}\beta_{gm}\big)\\
&= \textstyle\prod_{g=1}^{G}(\frac{1}{c\gamma_g})^{k_g}\exp\big(-\boldsymbol{D}\boldsymbol{B}\big)\\
&= f(\boldsymbol{B}_R)\\
&= \pi(\boldsymbol{B}_R|\boldsymbol{R},\boldsymbol{\gamma}))
\end{aligned}$$

Hence we now have

$$\begin{aligned}
& f(\boldsymbol{W}|\boldsymbol{A}_R,\boldsymbol{\Sigma},\boldsymbol{R},\boldsymbol{B}_R,\boldsymbol{\gamma})\pi(\boldsymbol{B}_R|\boldsymbol{R},\boldsymbol{\gamma})\\
= \;& (2\pi)^{(-NG/2)}|\boldsymbol{C}|^{-1/2}\exp\big(-\tfrac{1}{2}(\boldsymbol{W}-\boldsymbol{A}_R\boldsymbol{B}_R)^T\boldsymbol{C}^{-1}(\boldsymbol{W}-\boldsymbol{A}_R\boldsymbol{B}_R)\big)\\
& \times\textstyle\prod_{g=1}^{G}(\frac{1}{\gamma_g})^{k_g}\exp\big(-\boldsymbol{D}\boldsymbol{B}_R\big)\\
\propto \;& \exp\big(-\tfrac{1}{2}(\boldsymbol{W}-\boldsymbol{A}_R\boldsymbol{B}_R)^T\boldsymbol{C}^{-1}(\boldsymbol{W}-\boldsymbol{A}_R\boldsymbol{B}_R)\big)\exp\big(-\boldsymbol{D}\boldsymbol{B}_R\big)\\
= \;& \exp\Big(-\tfrac{1}{2}\big(\boldsymbol{W}^T\boldsymbol{C}^{-1}\boldsymbol{W}-\boldsymbol{W}^T\boldsymbol{C}^{-1}\boldsymbol{A}_R\boldsymbol{B}_R-\boldsymbol{B}_R^T\boldsymbol{A}_R^T\boldsymbol{C}^{-1}\boldsymbol{W}\\
& +\boldsymbol{B}_R^T\boldsymbol{A}_R^T\boldsymbol{C}^{-1}\boldsymbol{A}_R\boldsymbol{B}_R+2\boldsymbol{D}\boldsymbol{B}_R\big)\Big)\\
= \;& \exp\Big(-\tfrac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{A}_R^T\boldsymbol{C}^{-1}\boldsymbol{A}_R\boldsymbol{B}_R-2(\boldsymbol{W}^T\boldsymbol{C}^{-1}\boldsymbol{A}_R-\boldsymbol{D})\boldsymbol{B}_R+\boldsymbol{W}^T\boldsymbol{C}^{-1}\boldsymbol{W}\big)\Big)\\
= \;& \exp\Big(-\tfrac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{U}^{-1}\boldsymbol{B}_R-2\boldsymbol{V}\boldsymbol{B}_R+\boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T+\boldsymbol{Q}\big)\Big)\\
= \;& \exp(-\tfrac{1}{2}\boldsymbol{Q})\exp\Big(-\tfrac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{U}^{-1}\boldsymbol{B}_R-2\boldsymbol{V}\boldsymbol{U}\boldsymbol{U}^{-1}\boldsymbol{B}_R+\boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T\big)\Big)\\
\propto \;& \exp\Big(-\tfrac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{U}^{-1}\boldsymbol{B}_R-2\boldsymbol{V}\boldsymbol{U}\boldsymbol{U}^{-1}\boldsymbol{B}_R+\boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T\big)\Big).
\end{aligned}$$

Then, by integrating out $\boldsymbol{B}_R$ with the substitution $\boldsymbol{\alpha} = -\boldsymbol{B}_R$, we have:

$$\begin{aligned} f(\boldsymbol{W}|\boldsymbol{A}_R, \boldsymbol{\Sigma}, \boldsymbol{R}, \boldsymbol{\gamma}) \quad & \alpha \quad \textstyle\int_0^\infty \exp\Big(-\frac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{U}^{-1}\boldsymbol{B}_R - 2\boldsymbol{V}\boldsymbol{U}\boldsymbol{U}^{-1}\boldsymbol{B}_R + \boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T\big)\Big) d\boldsymbol{B}_R \\ & = \quad \textstyle\int_{-\infty}^0 \exp\Big(-\frac{1}{2}\big(\boldsymbol{\alpha}^T\boldsymbol{U}^{-1}\boldsymbol{\alpha} + 2\boldsymbol{V}\boldsymbol{U}\boldsymbol{U}^{-1}\boldsymbol{\alpha} + \boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T\big)\Big) d\boldsymbol{\alpha} \\ & = \quad \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}) * (2\pi)^{(G*M)/2} * |\boldsymbol{U}|^{1/2} \end{aligned}$$

So we can now write the distribution $f(\boldsymbol{W}|\boldsymbol{A}_R, \boldsymbol{\Sigma}, \boldsymbol{R}, \boldsymbol{\gamma})$ as:

$$\begin{aligned} f(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{\Sigma}, \boldsymbol{R}) &= (2\pi)^{(-NG/2)} \textstyle\prod_{g=1}^G (\frac{1}{c\gamma_g})^{k_g} |\boldsymbol{C}|^{-1/2} \exp(-\frac{1}{2}\boldsymbol{Q})(2\pi)^{(G*M)/2} \\ & \quad \times |\boldsymbol{U}|^{1/2}\Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}) \\ &= (2\pi)^{G(M-N)/2} \textstyle\prod_{g=1}^G (\frac{1}{c\gamma_g})^{k_g} |\boldsymbol{C}|^{-1/2}|\boldsymbol{U}|^{1/2} \exp(-\frac{1}{2}\boldsymbol{Q}) \\ & \quad \times \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}). \end{aligned}$$

To simplify it, we can show that since $|\boldsymbol{C}| = |\boldsymbol{\Sigma} \otimes I_N| = |\boldsymbol{\Sigma}|^N$ and also that:

$$\begin{aligned} \boldsymbol{U}^{-1} &= \boldsymbol{A}_R^T\boldsymbol{C}^{-1}\boldsymbol{A}_R \\ &= (-\boldsymbol{I}_G \otimes \boldsymbol{A}_R)^T(\boldsymbol{\Sigma} \otimes \boldsymbol{I}_N)^{-1}(-\boldsymbol{I}_G \otimes \boldsymbol{A}_R) \\ &= (\boldsymbol{I}_G \otimes \boldsymbol{A_R}^T)(\boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{I}_N)(\boldsymbol{I}_G \otimes \boldsymbol{A}_R) \\ &= (\boldsymbol{I}_G \otimes \boldsymbol{A}_R^T)(\boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{A}_R) \\ &= \boldsymbol{\Sigma}^{-1} \otimes \boldsymbol{A}_R^T\boldsymbol{A}_R, \end{aligned}$$

which leads to $|\boldsymbol{U}^{-1}|$ proportional to $|\boldsymbol{\Sigma}^{-1}|^K$, thus $|\boldsymbol{U}|$ proportional to $|\boldsymbol{\Sigma}|^K$, so we can write $|\boldsymbol{C}|^{-1/2}|\boldsymbol{U}|^{1/2}$ proportional to $|\boldsymbol{\Sigma}|^{\frac{K-N}{2}}$. That finally allows us to write the equation (4.12) as:

$$f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma}) = \prod_{g=1}^G \left(\frac{1}{c\gamma_g}\right)^{k_g} |\boldsymbol{\Sigma}|^{\frac{K-N}{2}} \exp\left(-\frac{1}{2}\boldsymbol{Q}\right) \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U}),$$

as required. □

4.3.2 Estimation of the regression coefficients

The main objective of the study is to estimate the association matrix $\boldsymbol{R}$, defining the functional interactions. However, if we also wish to infer the regression coefficients $\boldsymbol{B}_R$, these can estimated by their full conditional distribution, which follow a normal distribution. That can allow us to estimate the strength of the functional interactions: the more negative β_{gm} is, the stronger the interaction of interest is.

$$\begin{aligned}\pi(\boldsymbol{B}_R|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}) = &\ \pi(\boldsymbol{B}_R|\boldsymbol{W},\boldsymbol{A}_R,\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma})\\ \propto &\ f(\boldsymbol{W}|\boldsymbol{A}_R,\boldsymbol{\Sigma},\boldsymbol{R},\boldsymbol{B}_R,\boldsymbol{\gamma})\pi(\boldsymbol{B}_R|\boldsymbol{R},\boldsymbol{\gamma})\\ \propto &\ \exp\Big(-\tfrac{1}{2}\big(\boldsymbol{B}_R^T\boldsymbol{U}^{-1}\boldsymbol{B}_R - 2\boldsymbol{V}\boldsymbol{U}\boldsymbol{U}^{-1}\boldsymbol{B}_R + \boldsymbol{V}\boldsymbol{U}\boldsymbol{V}^T\big)\Big).\end{aligned}$$

from Proof 4.3.1. We then have a normal distribution with mean $-\boldsymbol{U}\boldsymbol{V}^T$ and covariance matrix $\boldsymbol{U}$:

$$\boldsymbol{B}_R|\boldsymbol{Y},\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma} \sim N(-\boldsymbol{U}\boldsymbol{V}^T,\boldsymbol{U}).$$

4.3.3 MCMC algorithm

The association matrix $\boldsymbol{R}$ can be estimated via a Metropolis-Hastings within Gibbs algorithm, allowing us to select the variables of interest. The notations $\boldsymbol{R}_{old},\tau_{old},\dots$ represent the current values of the parameters, before their update. Due to the size of the state space, due to the covariance matrix $\boldsymbol{\Sigma}$, computations are quite long, intensive and expensive. We then propose to focus only on the potential interactions, the ones for which $s_{gm} > 0$, assuming that the interactions with an absent TargetScan score are not functional. The algorithm can be divided into four steps.

- First, we update the association matrix $\boldsymbol{R}$ by changing the value of one element for which $s_{gm} > 0$, randomly chosen, resulting in an add or a deletion of an edge. The proposed $\boldsymbol{R}_{new}$ is accepted with probability:

$$\min\left[\frac{f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}_{new}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})\pi(\boldsymbol{R}_{new}|\tau)}{f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}_{old}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})\pi(\boldsymbol{R}_{old}|\tau)}, 1\right]. \tag{4.13}$$

- In the second step, we update the parameter τ, the hyperparameter of the prior model. We sample τ_{new} from a truncated normal distribution $q(\tau_{new}|\tau_{old})$ with mean τ_{old}, truncated at 0, which is accepted with probability:

$$\min\left[\frac{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{old}|\tau_{new})}{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{new}|\tau_{old})}, 1\right]. \tag{4.14}$$

  The truncature at 0 ensures us the positivity of our parameter, while the variance of that proposal distribution has to be set such that we obtain a suitable acceptance rate, in order to efficiently explore the parameter space.

  We can note that other proposal distributions can be chosen. It is for example acceptable to choose a gamma distribution, or also a log-normal distribution, with parameters ensuring that the mean or mode of the proposal distribution is the current value of the parameter.

- Then, we update the prior parameter γ_g of the gene involved in the edge update two steps before. The proposal distribution is a gamma distribution, with parameters $\alpha_{\gamma_g} = \gamma_g^2/e$ and $\beta_{\gamma_g} = e/\gamma_g$, where e is the tuning parameter and has to bet up suitably to obtain a correct acceptance rate. This gamma distribution $q(\gamma_g^{new}|\gamma_g^{old})$ ensures the positivity of the parameter, and is centered on the current value of γ_g. The probability of acceptance is then:

$$\min\left[\frac{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}_g^{new})\pi(\gamma_g^{new})q(\gamma_g^{old}|\gamma_g^{new})}{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}_g^{old})\pi(\gamma_g^{old})q(\gamma_g^{new}|\gamma_g^{old})},1\right]. \tag{4.15}$$

- Finally, we update the covariance matrix $\boldsymbol{\Sigma}$ using a Metropolis step where the proposal distribution $q(\boldsymbol{\Sigma}_{new}|\boldsymbol{\Sigma}_{old})$ is a Wishart with parameters $\nu_{\boldsymbol{\Sigma}}$ and $\boldsymbol{A}_{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}_{old}/\nu_{\Sigma}$. The proposed $\boldsymbol{\Sigma}_{new}$ is then accepted with probability:

$$\min\left[\frac{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma}_{new},\boldsymbol{\gamma})\pi(\boldsymbol{\Sigma}_{new})q(\boldsymbol{\Sigma}_{old}|\boldsymbol{\Sigma}_{new}))}{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma}_{old},\boldsymbol{\gamma})\pi(\boldsymbol{\Sigma}_{old})q(\boldsymbol{\Sigma}_{new}|\boldsymbol{\Sigma}_{old}))},1\right]. \tag{4.16}$$

  The setting of the parameters ensures that the mean of the proposed covariance matrix is centered on the current one. The variance of the proposal distribution, then the acceptance rate, is determined by the degrees of freedom ν_{Σ}, which has to be set suitably.

Then, at the end of the analysis, we compute for each potential interaction P_{gm}, the proportion of MCMC samples for which $r_{gm} = 1$, in order to perform posterior inference. To accomplish this task, we decide of a threshold t_h, between 0 and 1. Then we declare functional all the interactions for which $P_{gm} \geq t_h$. The higher t_h is, the less interactions are considered functional, t_h reflecting the confidence of those links.

4.4 Summary and pseudo-code of the algorithm

In this section, we give a summary of the proposed approach. In Table 4.1, we give a summary of the framework of the methodology, with a description of the data, the model, the parameters and their relative priors, concluded by the relative probability of the model and regulatory network. In Table 4.2, we describe the derivation of the likelihood $f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})$, by giving the principal steps. Finally, in Table 4.3, we describe the MCMC procedure of our methodology: for each type of update, we give the proposal distribution used, and the probability of acceptance.

4.5 Setting up the parameters

Before applying the proposed methodology to simulated and then to real data, we first need to set up the values of our hyperparameters. The target scores s_{gm} available from TargetScan are negative values, the lowest values representing the most likely gene-miRNA interactions. However, the scores in our prior probability model (4.4) are supposed to be positive, the highest values corresponding to the most likely interactions. as a result, we transform the TargetScan scores to new values for our framework: $s_{gm} = cdf(-s^{o}_{gm})$, where s^{0}_{gm} are the original TargetScan scores. That makes our scores follow an uniform distribution between 0 and 1, and importantly give the highest values to the most probable interactions. Another possibility would have been to just change the sign of the scores, so the most likely interactions would still have the highest scores. However, due to the lack of available information

Data:
Genes expression levels: $\boldsymbol{Y}, N \times G$ matrix, miRNAs expression levels: $\boldsymbol{X}, N \times M$ matrix, TargetScan scores: $\boldsymbol{S}, G \times M$ matrix.
Model:
$\boldsymbol{Z} = (\boldsymbol{Y}, \boldsymbol{X}) \sim N(\boldsymbol{0}, \boldsymbol{I}_N, \boldsymbol{\Omega})$, $\boldsymbol{Y} = -\boldsymbol{X}\boldsymbol{\beta}^T + \boldsymbol{\epsilon}$, $\boldsymbol{\beta}, G \times M$ matrix of the regression coefficients, $\boldsymbol{\epsilon} \sim N(\boldsymbol{0}, \boldsymbol{I}_N, \boldsymbol{\Sigma})$.
Parameters and prior distributions::
$\pi(\beta_{gm} \vert \boldsymbol{R}, \gamma_g) = r_{gm} Ga(1, c\gamma_g) + (1 - r_{gm}) I_{\beta_{gm}=0}$, $\gamma_g \sim IGa((\delta + k_g)/2, 2/d)$, k_g = number of miRNAs involved in the regression of the g-th target gene, $K = \sum_g^G k_g$, $P(r_{gm} = 1 \vert \tau) = \frac{\exp(\eta + \tau s_{gm})}{1 + \exp(\eta + \tau s_{gm})}$, $\tau \sim Ga(a_\tau, b_\tau)$, $\boldsymbol{\Sigma} \sim IW_{\nu_\Sigma}\Big(\nu_\Sigma, (\nu_\Sigma + G + 1)\boldsymbol{I}_G\Big)$.
$f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \tau, \boldsymbol{\beta}, \boldsymbol{\gamma}) \propto f(\boldsymbol{Y} \vert \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \tau, \boldsymbol{\beta}, \boldsymbol{\gamma}) \times \pi(\boldsymbol{\beta} \vert \boldsymbol{R}, \boldsymbol{\gamma})$, $\times \pi(\boldsymbol{\Sigma}) \times \pi(\boldsymbol{\gamma}) \times \pi(\boldsymbol{R} \vert \tau) \times \pi(\tau)$.

Table 4.1: Framework of the model

Derivation of $f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})$:
$\boldsymbol{W} = \mathrm{vec}(\boldsymbol{Y})$,
$\boldsymbol{A} = -\boldsymbol{I}_G \otimes \boldsymbol{X}$,
$\boldsymbol{B} = \mathrm{vec}(\boldsymbol{\beta}^T)$,
$\boldsymbol{C} = \boldsymbol{\Sigma} \otimes \boldsymbol{I}_N$,
$f(\boldsymbol{W} \mid \boldsymbol{X}, \boldsymbol{R}, \Sigma, \boldsymbol{\beta}, \boldsymbol{\gamma}) \sim N(\boldsymbol{AB}, \boldsymbol{C})$,
$\boldsymbol{W} = \boldsymbol{A}_R \boldsymbol{B}_R + \boldsymbol{\epsilon}_2$,
$\boldsymbol{A}_R, \boldsymbol{B}_R$: matrix and vector composed of the nonzeros columns and elements of $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively,
$\boldsymbol{\epsilon}_2 \sim N_{N \times G}(\boldsymbol{0}, \boldsymbol{C})$,
$\pi(\boldsymbol{\beta} \mid \boldsymbol{R}, \gamma) = \prod_{g=1}^{G} (\frac{1}{c\gamma_g})^{k_g} \exp\big(-\boldsymbol{DB}\big)$,
$\boldsymbol{D} : 1 \times GM$ vector,
$\boldsymbol{D}_{g*m} = \begin{cases} 1/(c\gamma_g) & \text{if } r_{gm} = 1, \\ 0 & \text{otherwise} \end{cases}$
$\boldsymbol{U} = (\boldsymbol{A}_R^T \boldsymbol{C}^{-1} \boldsymbol{A}_R)^{-1} = \Sigma \otimes (\boldsymbol{X}^T \boldsymbol{X})^{-1}$,
$\boldsymbol{V} = \boldsymbol{W}^T \boldsymbol{C}^{-1} \boldsymbol{A}_R + \boldsymbol{D}$,
$\boldsymbol{Q} = \boldsymbol{W}^T \boldsymbol{C}^{-1} \boldsymbol{W} - \boldsymbol{VUV}^T$,
$\Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{UV}^T, \boldsymbol{U})$: cdf of a normal multivariate $(G * M)$ distribution, calculated at the zero vector, with mean $-\boldsymbol{UV}^T$ and covariance matrix $\boldsymbol{U}$.
$f(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma}) = \prod_{g=1}^{G} \left(\frac{1}{c\gamma_g}\right)^{k_g} \lvert\boldsymbol{\Sigma}\rvert^{\frac{K-N}{2}} \exp\left(-\frac{1}{2}\boldsymbol{Q}\right) \Phi_{G*M}(\boldsymbol{0}, -\boldsymbol{UV}^T, \boldsymbol{U})$.

Table 4.2: Posterior inference: derivation of $f(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma}, \boldsymbol{\gamma})$

```
Pseudo-code of the MCMC procedure:
Update 1: $\boldsymbol{R}$
  elem= element of $\boldsymbol{R}$ randomly chosen,
  $\boldsymbol{R}_{new} = \boldsymbol{R}$,
  $\boldsymbol{R}_{new}[\text{elem}] = 1 - \boldsymbol{R}_{old}[\text{elem}]$,
  $\alpha = \min\left[\frac{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{old}|\tau_{new})}{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{new}|\tau_{old})}, 1\right]$,
  $u = U(0,1)$: $u$ is sampled from a uniform distribution on $(0,1)$,
  if $(\alpha \geq u)$, then $(\boldsymbol{R} = \boldsymbol{R}_{new})$.
Update 2: $\tau$
  Proposal distribution $q(\tau_{new}|\tau_{old})$: truncated normal distribution,
  with mean $\tau_{old}$ and truncated at 0,
  $\tau_{new}$: sampled from $q(\tau_{new}|\tau_{old})$,
  $\alpha = \min\left[\frac{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{old}|\tau_{new})}{\pi(\boldsymbol{R}|\tau_{new})\pi(\tau_{new})q(\tau_{new}|\tau_{old})}, 1\right]$,
  $u = U(0,1)$,
  if $(\alpha \geq u)$, then $(\tau = \tau_{new})$.
Update 3: $\gamma_g$
  Proposal distribution $q(\gamma_g^{new}|\gamma_g^{old}) = Ga(\alpha_{\gamma_g} = \gamma_g^2/e, \beta_{\gamma_g} = e/\gamma_g)$,
  $\gamma_g^{new}$: sampled from $q(\gamma_g^{new}|\gamma_g^{old})$,
  $\alpha = \min\left[\frac{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}_g^{new})\pi(\gamma_g^{new})q(\gamma_g^{old}|\gamma_g^{new})}{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma},\boldsymbol{\gamma}_g^{old})\pi(\gamma_g^{old})q(\gamma_g^{new}|\gamma_g^{old})}, 1\right]$,
  $u = U(0,1)$,
  if $(\alpha \geq u)$, then $(\gamma_g = \gamma_g^{new})$.
Update 4: $\boldsymbol{\Sigma}$
  Proposal distribution $q(\boldsymbol{\Sigma}_{new}|\boldsymbol{\Sigma}_{old}) = W_G(\nu = \nu_\Sigma, \boldsymbol{A}_{\boldsymbol{\Sigma}}\nu_\Sigma\boldsymbol{\Sigma}_{old})$,
  $\boldsymbol{\Sigma}_{new}$: sampled from $q(\boldsymbol{\Sigma}_{new}|\boldsymbol{\Sigma}_{old})$,
  $\alpha = \min\left[\frac{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma}_{new},\boldsymbol{\gamma})\pi(\boldsymbol{\Sigma}_{new})q(\boldsymbol{\Sigma}_{old}|\boldsymbol{\Sigma}_{new}))}{f(\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{R},\boldsymbol{\Sigma}_{old},\boldsymbol{\gamma})\pi(\boldsymbol{\Sigma}_{old})q(\boldsymbol{\Sigma}_{new}|\boldsymbol{\Sigma}_{old}))}, 1\right]$,
  $u = U(0,1)$,
  if $(\alpha \geq u)$, then $(\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{new})$.
```

Table 4.3: Pseudo-code of the MCMC approach

concerning the scores, we chose to change the frequency distribution and make the scores distribution uniform between 0 and 1, so that we have a flat prior.

The parameter η gives the prior belief of interaction, especially if we do not have access to the sequence and structure knowledge of the s_{gm}'s, assuming all the scores are equal to 0. Since it is believed that genes are on average regulated by one or two miRNAs, we set η close to $\log(\frac{1}{M-1})$, which gives a prior expected number of regulating miRNAs close to 1 per gene. If we have more information available about the target scores, from one or more sources, we can assume that the prior probability increases. This is the reason why we set the parameters a_τ and b_τ of the gamma distribution in such a way that the mean $a_\tau b_\tau$ is close to $\frac{1}{0.5}[\log(\frac{2}{M-2}) - \eta]$, and the variance $a_\tau b_\tau^2$ allows high probability to a wide range of values for τ, where 0.5 represents the mean of the scores s_{gm} and 2 the prior number of expected regulating miRNAs per gene.

The hyperparameter ν_Σ is given the value $G + 2$, the minimum value such that the mean of $\mathbf{\Sigma}$ exists. Moreover, that allows the mode of $\mathbf{\Sigma}$ to be equal to the identity matrix, resulting in a vague prior for the covariances, as in practice, we may not know their strength and/or sign. The other hyperparameters are set in a similar way as in [58]. The parameter c which appears in the prior distribution of the regression coefficients β_{gm}'s can be seen as a correction factor. According to Section A.2, accepting only the positive values of a normal distribution with zero mean and variance σ^2 results in a truncated normal distribution with variance close to $0.36\sigma^2 \simeq (0.6\sigma)^2$. The variance of the prior distribution of β_{gm} being $(c\gamma_g)^2$ we can set

$c = 0.6$ or a close value. Finally, we specify a vague prior on the γ_g's, setting $\delta = 3$, the minimum integer value such that the expected value of γ_g exists (in the case $k_g = 0$) and $d = 0.2$ in such a way the mean of the distribution is comparable to the error variances of $\boldsymbol{Y}$ given $\boldsymbol{X}$.

4.6 Case study on simulated data

Before applying our model to real data, we want to study its efficiency, and especially whether it improves on the estimation approach when the genes are supposed to be independent. This is important because, even though some genes may be assumed to be independent, it is generally accepted that some genes are known or expected to be strongly correlated, which can be the case of a group of genes involved in one specific biological function. Therefore we set out to compare both methods on a set of simulated data. We will build a network of gene-miRNA interactions, by sampling the data and parameters under the assumptions of our model. Then, we will apply both methods, whether the genes are assumed independent or not, in order to recreate from a random network the original network built under our model. This will allow us to compare the different ouputs, in particular if the predicted networks are similar to the original one. For a first analysis, we will run our model with the true and known (in this occasion) covariance matrix, without updating it. Then later, we will discuss the whole algorithm, where we need to update and estimate it, as it is the case for real data, when we ignore the true covariance matrix.

4.6.1 Building of the network

We obviously first need to build our network, under our assumptions. To do so, we choose numbers of genes and miRNAs relatively close to the dimensions of our real datasets: $G = 20$ and $M = 8$ We also chose $N = 20$, a number close to the number of samples we have for our three different conditions. We then sample the miRNAs data, $\boldsymbol{X}$, from a normal distribution. As the miRNAs are assumed independent, we can sample them independently from the same normal distribution with zero mean. Since the marginal distribution of $\boldsymbol{X}$, thus its variance, does not affect the regulatory network, we choose a variance of 1. The covariance matrix of $\boldsymbol{Y}|\boldsymbol{X}$, $\boldsymbol{\Sigma}$, is then sampled from an inverse-Wishart distribution, with degrees of freedom $\nu_{\Sigma} = 22$, the minimum number such that the expectation of $\boldsymbol{\Sigma}$ exists, a value which still specifies a vague prior. Since we want to consider the genes be correlated, we deliberately set the scale matrix of the prior to be a covariance matrix of highly correlated genes. That is why we chose a scale matrix such that the mode of the inverse-Wishart prior is:

$$\boldsymbol{\Psi} = \begin{pmatrix} 1 & \rho & \cdots & \rho \\ \rho & 1 & \cdots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \cdots & 1 \end{pmatrix},$$

with ρ being a high correlation, $\rho = 0.9$ for example. That leads, as we can see in Figure 4.3, to high correlations of $\boldsymbol{Y}|\boldsymbol{X}$, with an average correlation of $\simeq 0.78$. The choice of high positive correlations can be explained by the fact as it is believed that genes are correlated, so we really wanted

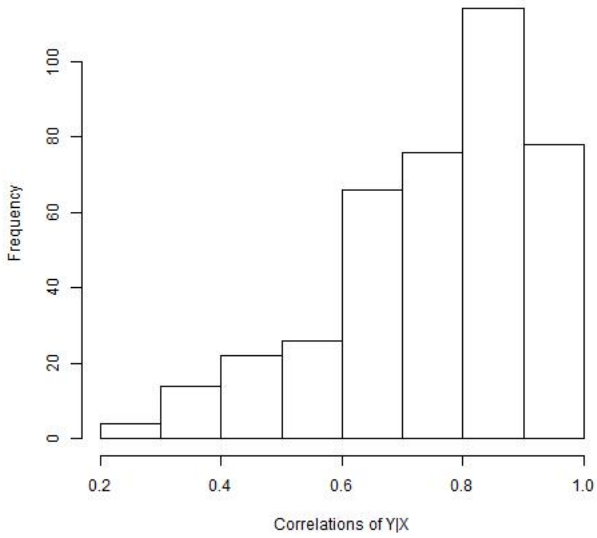

Figure 4.3: Simulated correlations of $\boldsymbol{Y}|\boldsymbol{X}$.

to see the importance of these correlations compared to the model with absence of correlation. Moreover, it would be very unlikely that negatively correlated genes get grouped in the same cluster, especially when using AutoSOME which groups together nodes with high similarities, thus positive correlations. However, it has to be pointed out that, in we consider negative correlations, we might encounter problems when sampling from a Wishart or Inverse-Wishart distribution, as the determinant of such a matrix (with only negative correlations) may be negative.

Now, knowing $\boldsymbol{\Sigma}$, we can then compute $\boldsymbol{\epsilon} = \boldsymbol{I}_N \boldsymbol{\epsilon}_0 \boldsymbol{\Sigma}^{1/2}$, $\boldsymbol{\epsilon}_0$ being a matrix of elements sampled from a standard normal distribution. Indeed, since $\boldsymbol{\epsilon}_0 \sim N_{N,G}(\boldsymbol{0}, \boldsymbol{I}_N, \boldsymbol{I}_G)$, then according to A.2 and [20], we have $\boldsymbol{\epsilon} \sim N_{N,G}(\boldsymbol{0}, \boldsymbol{I}_N, \boldsymbol{\Sigma})$.

We still need to define the actual network of functional interactions. Since we have 20 genes and 8 miRNAs, (160 potential interactions), and it is believed that genes are generally regulated by few miRNAs, we decide to make 25 interactions functional, so that the average number of regulating miRNAs per gene is close to 1. We thus sample 25 elements out of 160 in order to define our true $\boldsymbol{R}$ matrix, containing 25 elements equal to 1, and all other elements being zero.

The last parameters we need to sample for our data are the regression coefficients β_{gm}'s. According to our model, we sample each β_{gm} via a gamma distribution $Ga(1, c\gamma_g)$ with each γ_g following an inverse-Gamma distribution $IGa((\delta + k_g)/2, 2/d)$, where k_g is the number of miRNAs involved in the regulation of each target gene g, according to the $\boldsymbol{R}$ matrix we have just sampled. We choose $\delta = 3$ and $d = 0.2$, values chosen by Stingo et al. [58], values which result in a vague prior. Finally, we need to make sure the regression coefficients correspond to the regulating network, that is why we set to 0 the β_{gm}'s which correspond to the r_{gm}'s equal to 0.

We now have all the quantities we need to compute the gene expression levels $\boldsymbol{Y}$, following Equation (4.2):

$$\boldsymbol{Y} = -\boldsymbol{X}\boldsymbol{\beta}^T + \boldsymbol{\epsilon}.$$

The regulating network and the data have now been sampled and computed, however, we still need to sample few parameters for the MCMC algorithm. First, we sample the simulated target scores, each score s_{gm} being sampled from an uniform distribution $U(0, 1)$. Then, the parameter η is set to $\eta = -1.9$, which result in a prior expected number of regulating miRNAs per gene close to 1. The hyperparameters for τ are set to $a_\tau = 2.5$ and $b_\tau = 1$, so the prior probability of interaction gets higher when we know the target scores, and these values also give a variance of 2.5, thus a high probability to a wide range of τ values.

4.6.2 Covariance matrix assumed to be known

Since the network is now simulated, we will find out how the MCMC procedure performs. We will start from a random configuration, and we want to recreate the initial scenario. We sample a new $\boldsymbol{R}$ interaction table, composed of 20 elements set to 1, one interaction per gene on average. We then apply the estimation procedure in order to predict the true network. To create our inferred network, we compute for each potential interaction P_{gm}, the proportion of MCMC samples for which $r_{gm} = 1$. After deciding on a threshold t_h, we select as functional all the arrows for which $P_{gm} \geq t_h$.

However, in this first analysis, we do not apply the full estimation procedure. In order to study the importance of the correlations between genes in the regulatory network, we set out to compare our method in the best-case scenario (when we know the true correlations), with the method when the target genes are assumed independent. Thus, we can find out what is the effect of correlated genes (which is likely to be the case in real data) for the construction of regulatory networks.

The following graphs and results have been computed with a threshold $t_h = 0.75$, and a number of 4,000 MCMC iterations, including a burn-in of 1,500 iterations.

First of all, we look at the arrows included and selected. In Figure 4.4, we can see on the left panel the number of interactions our model predicts at each iteration. We can see that after the burn-in period, when the number of arrows varies quite a lot with a huge increase, then its stabilizes after about 2,000 iterations, to 24 included functional interactions, a value very close to the number of 25 interactions we set up as functional in the building of our network. Similarly, we can see on the right panel that the number of interactions, which reached the threshold of 75% after the burn-in period, quickly stabilizes at 23 arrows, once again close to the true 25.

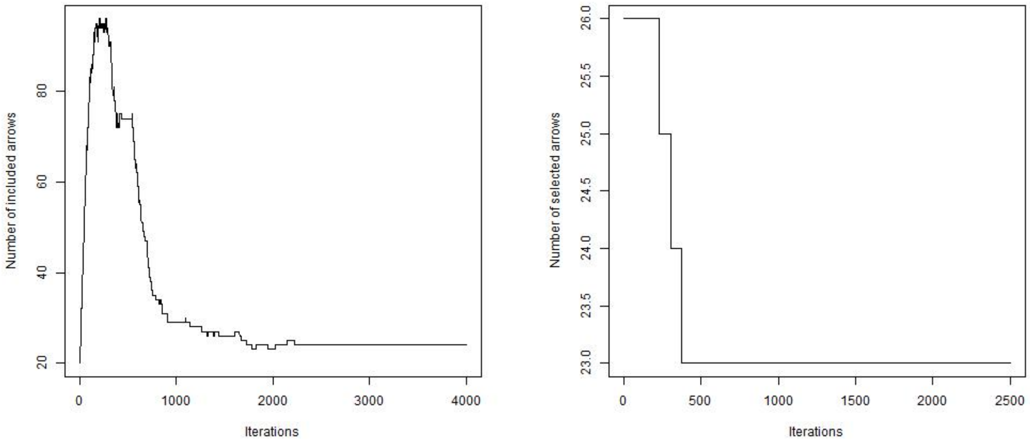

Figure 4.4: Number of included arrows (left) and number of selected arrows (right).

After checking that the 23 inferred interactions were indeed included in the simulated network, without any false positives, we can conclude that our analysis correctly predicted back 23 out of the 25 initial functional arrows. We now turn out attention to the two arrows which were not predicted by our algorithm. The first observation we can make is that the actual regression coefficients of these two arrows are $\beta_{gm} \simeq 0.86$ and $\beta_{gm} \simeq 0.71$ respectively. These two low values of β_{gm}, even if they are not the lowest ones, indicate that the given interactions were quite weak, as if the miRNAs of interest only had a limited regulation over those targets. We can then assume that, for those interactions, in our simulations and computation of $\boldsymbol{Y}$, the white noise $\boldsymbol{\epsilon}$ played a bigger role than the miRNAs themselves. However, we can also check the actual proportions P_{gm} of presence. Then we can notice that one of those missing interactions actually has a proportion very close to the threshold, $P_{gm} \simeq 0.742$, explaining the difference between the 24 included arrows (left panel) and the 23 selected ones (right panel). We can then assume that if we had run the analysis for a bit longer, or if we had actually chosen a lower threshold, that given interaction would have selected as well. The second one though, would not have been selected, having never been included in the model after the burn-in period.

The convergence of our model can be seen in the trace plots of the log-relative probability of the model, log-probability computed as explained in Equation 4.6. In Figure 4.5, we plot the log-probability of the whole model, for all iterations on the left panel, and only after the burn-in period on the right panel. We see that, after that burn-in period, it converges quite quickly and stabilizes to what we can now call as the true distribution. The exact same observations can be made about the likelihood $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$, which was expected in that case. Indeed, in the whole model, the two quantities which have the biggest contributions to the probability of the model are the contribution of the covariance matrix $\pi(\boldsymbol{\Sigma})$, and then the likelihood

$f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$. But since in this analysis, we do not update the true covariance matrix, the shape of the log-relative probability really fits closely the log-likelihood $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$, the probabilities of $\boldsymbol{R}$ and of the γ_g's having a slight impact on the total probability.

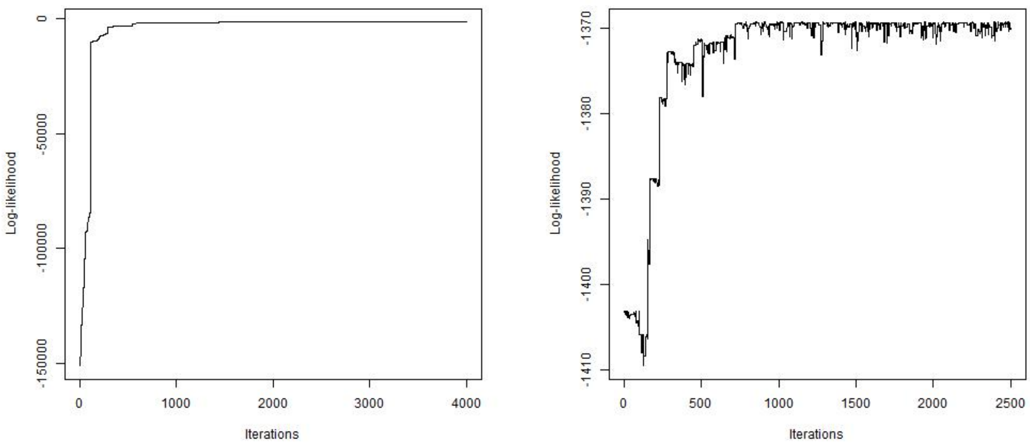

Figure 4.5: Log-relative probabilities of the visited models, for all iterations (left), and for after the burn-in (right).

Finally, we can have a look in Figure 4.7 at the converged chain of the τ parameter, whose the shape fits the one of the number of arrows included, Figure 4.4.

4.6.3 Genes assumed to be independent

After applying our procedure with the true covariance matrix, we apply, to the same data and parameters (when applicable), the procedure from Stingo et al. [58], when the targets are assumed independent upon the miRNAs. We run this chain for 25,000 iterations with a burn-in of 5,000 iterations. We run this one for many more iterations than the last one, in order to see

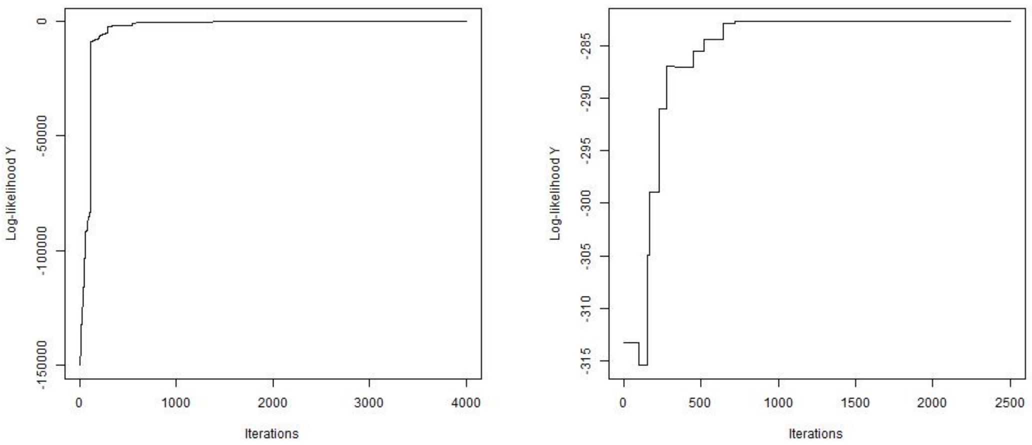

Figure 4.6: Log-likelihood $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$, for all iterations (left), and for after the burn-in (right).

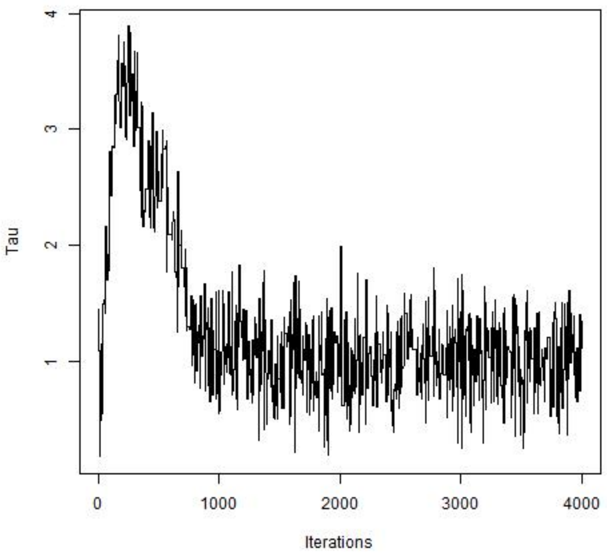

Figure 4.7: Values of τ.

more clearly its convergence. Indeed, as we can see in the graphs later, we observe more additions and deletions or arrows, having then an impact on the probability of the model. That is why more iterations allow us to observe and conclude that the chain always keeps moving a bit.

Also, the big increases of the probability are due to the appropriate additions or deletions of arrows in the inferred network, thus it greatly depends on the number of arrows included at each time. Indeed, several different networks, with approximately the same number of interactions, will have a relatively close log-relative probability. That is why, for this chain, we start from a random configuration of 100 arrows, a number far enough from the actual value of 25, hence allowing us to distinguish the increase of probability when the false interactions are deleted from the inferred network, which would not have been obvious if we had started from a configuration with 20 arrows (graph not provided).

From Figure 4.8, we see that the chain seems to have converged once the number of arrows had been reduced to between 20 and 40 arrows, a number consistent with the prior probabilities of interactions, between 1 and 2 regulating miRNAs per gene on average. However, we also still notice a lot of variations, the number of arrows not really stabilizing around a narrowed range of values.

This is why we need to look at the interactions which reach certain thresholds of presence, interactions which are present most of the time. That would indicate whether the functional interactions are predicted with lower confidence, and/or if we have more false positives. Figure 4.9 show us the number of arrows which are inferred 50% of the time (graph on the left) and 75% of

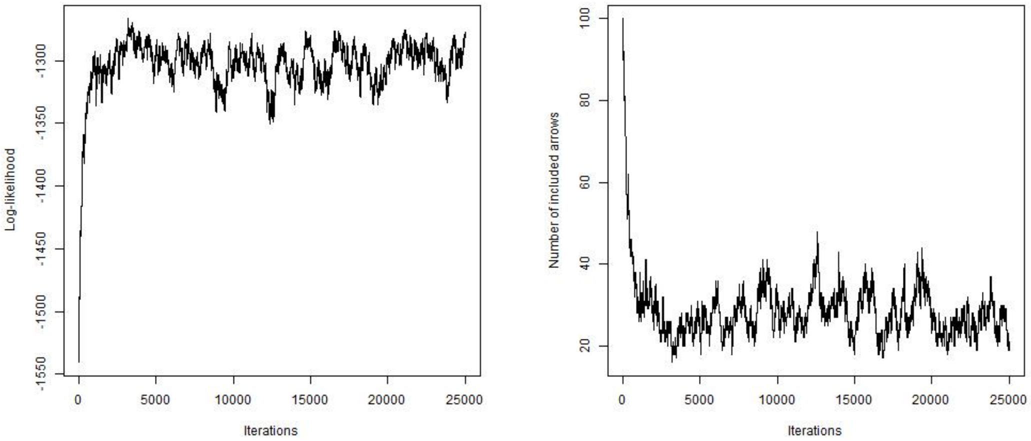

Figure 4.8: Log-probability (left panel), and number of arrows included in the network (right panel), for all iterations.

the time (graph on the right), once the burn-in period is over. We can then realize that, after 15,000 iterations, the number of arrows predicted, at 50% and 75% confidence, converges to only 15 and 11 respectively, two values relatively distant from the original number of 25 functional interactions, and especially the number of 23 interactions predicted by the previous analysis. We need to use a threshold of 25% of presence to obtain 26 predicted arrows, included 4 false positives, resulting in 3 missing functional interactions.

After checking the missing original arrows, we observe that most of them are the weakest interactions, with the lowest β_{gm}'s, most of the strong interactions being predicted. However, we can notice that the second strongest interaction, with a original regression coefficient $\beta_{gm} \simeq 14.3$ is not inferred by the algorithm, not even at a 50% confidence. More generally, the twelfth interaction, with $\beta_{gm} \simeq 4.1$ is the second and third strongest one not to be predicted at 50% and 75% respectively, with the sixth one, $\beta_{gm} \simeq 6.8$

not being predicted at the highest threshold. All the other false negatives interactions are among the weakest half of the original interactions.

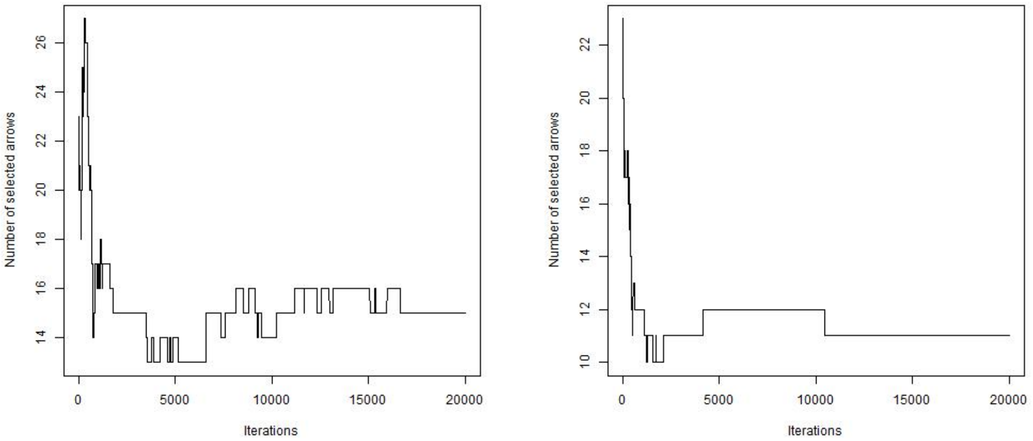

Figure 4.9: Number of arrows selected, with a threshold of 50% (left panel), and 75% (right panel), after the burn-in.

4.6.4 Estimation of the covariance matrix

In this section, we will apply the full procedure presented in Section 4.3.3, estimating the covariance matrix, to recreate the original network. We can then compare its efficiency with the two previous analyses. A priori, we can expect it to work better than the case where the genes are assumed independent given the miRNAs, as the model has been created with correlations between variables. On the other hand, since we do not know the true covariance and try to estimate it, we may not obtain results as precise as in the first scenario.

However, as we might expect with the dimensions of the problem, convergence can take long, leading to excessive computational cost. Moreover, the complexity of the model can also make convergence difficult to identify, as we will see if we always update the matrix. These are the reasons why we need to slightly alter the procedure, in order to improve its efficiency.

Starting configuration

The first thing we might wonder is the starting configuration: the table of interactions $\boldsymbol{R}$ and the covariance matrix $\boldsymbol{\Sigma}$ in particular. Then, we want to tune the priors, so we can start from a configuration not too far from the true one, to limit the waiting time for convergence, but not too close, as in practice we are not aware of the true distribution. This does not mean that this configuration is close in practice to the real configuration, but that is likely the best prior guess we can have at the start of the approach. For the starting points, a solution is to consider the regression model linear:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta}^l + \boldsymbol{\epsilon}^l,$$

where the errors $\boldsymbol{\epsilon}^l$ are assumed independent, and $\boldsymbol{\beta}^l$ are the ordinary least squares estimated regression coefficients. This approach is possible only if we have enough data, more observations than regressors, which is the case in our study.

After estimating $\boldsymbol{\beta}^l$, we can estimate $\boldsymbol{R}_0$, the starting table of interactions. As it is believed miRNAs down-regulate the gene expressions, we can identify the interactions with the lowest (negative) $\boldsymbol{\beta}^l_{gm}$, and set these interactions up to 1 in R_0. We chose the 25 lowest coefficients, as we know we created a network of 25 functional interactions. However, it is perfectly feasible to choose a different number of arrows for the starting configuration, as long as it is sensible to prior knowledge. For example, since most genes are regulated by one or two miRNAs, it is possible to choose between 20 or 30 arrows. As we might expect, the resulting R_0 is relatively close to the true interaction table, but not identical. Indeed, 18 of the 25 initial arrows are set up to 1, resulting in 7 missing arrows, and thus 7 false positives.

Still under the assumption of linear regression, we compute an estimate of the errors, $\hat{\boldsymbol{\epsilon}} = \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}^{\boldsymbol{l}}$. We can then get a first estimation of $\boldsymbol{\Sigma}$, by computing the among-column variance $\hat{\boldsymbol{S}}$ of $\hat{\boldsymbol{\epsilon}}$. When we compare both matrices, the true $\boldsymbol{\Sigma}$ and $\hat{\boldsymbol{S}}$, we observe that the values lie in the same range, and have a similar mean. However, the likelihood of $\hat{\boldsymbol{S}}$ is much lower than the true one. If we use this estimate as a starting point, it will take a very long time to reach convergence. An alternative is to sample a new matrix $\boldsymbol{S}$ from the inverse-Wishart prior, and rescale it to $\boldsymbol{S}_0 = \boldsymbol{S}/v$, v being a constant such that $\boldsymbol{S}_0$ has a mean value similar to the mean of $\hat{\boldsymbol{S}}$ and the true $\boldsymbol{\Sigma}$.

First chain: burn-in estimation of the covariance matrix without any other update

Since we now have our starting configuration, we can apply the MCMC algorithm. However, still in the aim of reducing the waiting time for convergence, we chose to run two chains. In the first one, that we can call a burn-in estimation or pre-estimation of $\boldsymbol{\Sigma}$, we only update only the covariance matrix $\boldsymbol{S}_0$, all the other parameters $(\boldsymbol{R}, \tau, \gamma_g\text{'s})$ remaining unchanged. By doing so, we allow the chain to move faster to a point closer to the true distribution. This is where $\boldsymbol{R}_0$ is useful. Since it is relatively close to the true $\boldsymbol{R}$, the estimation of $\boldsymbol{\Sigma}$ under this condition is still reasonable. However, we need to be careful though not to run this chain for too long, as the regulatory network defined by $\boldsymbol{R}_0$ is not the true one. Indeed, we do not want to reach the likelihood of the original configuration, just close enough, as we still want to apply the four updates of the whole procedure in the second chain. In this case, we ran this pre-estimation of $\boldsymbol{\Sigma}$ for 1,500 iterations. The resulting matrix, $\boldsymbol{S}_1$ will then be the starting matrix of the second chain, where the full MCMC procedure is applied.

As we briefly said in Section 4.3.3, the degrees of freedom ν_Σ have to be set such that we obtain a suitable acceptance rate. That is the reason why we set $\nu_\Sigma = 10,000$, leading to an acceptance rate of 30% in the first burn-in chain, and 20% in the second chain.

Second chain: parameters updated at each iteration

With all the parameters set up, and the final starting values from the first chain, we can now apply the full MCMC algorithm to infer the original regulatory network. So we first run the chain for a total of 5,000 iterations, with a burn-in of 1,000 iterations, where all the parameters are updated at each iterations, as described in Section 4.3.3.

The first output we can have a look at is the number of selected arrows, at a threshold or confidence level of 75%, meaning these arrows are included in the network at least 75% of the time, visible in Figure (4.10). We notice that 24 arrows are infered at that level. We can mention that there are no false positives, and these 24 selected interactions are exactly the same as the ones predicted in Section 4.6.2, where we ran a chain with the known covari-

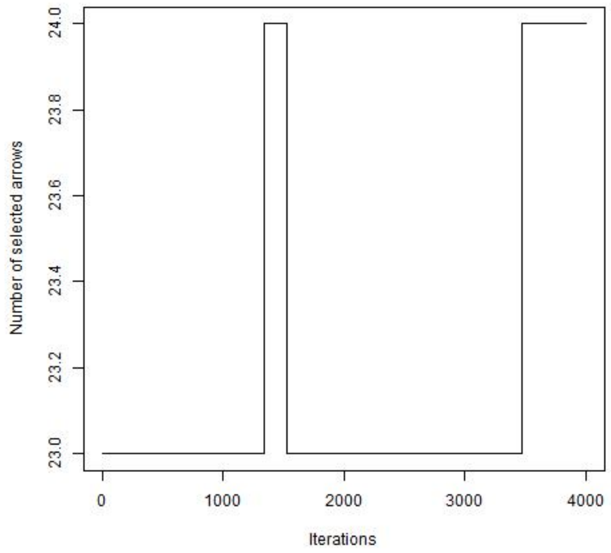

Figure 4.10: Number of selected arrows, at a 75% confidence level.

ance matrix. We can also add that 23 of these 24 arrows were predicted with $t_h = 0.9$. These results then suggest a very good performance of the chain, matching the results of the analysis where we assumed the matrix known. This impression may be reinforced by the plots of the log-probability, which seem to have converged, once the covariance matrix has converged and is no longer updated, as we see in Figure (4.11).

However, we notice a significant difference in the values of the log-probabilities, between this scenario and the one in Section 4.6.2. Indeed, in the first scenario, which is the true one, it converges around -1370, while this time it converges around +800, a much bigger value. The impression of passing convergence and going beyond the true distribution level may be confirmed when we observe the number of arrows included in the network at each iteration

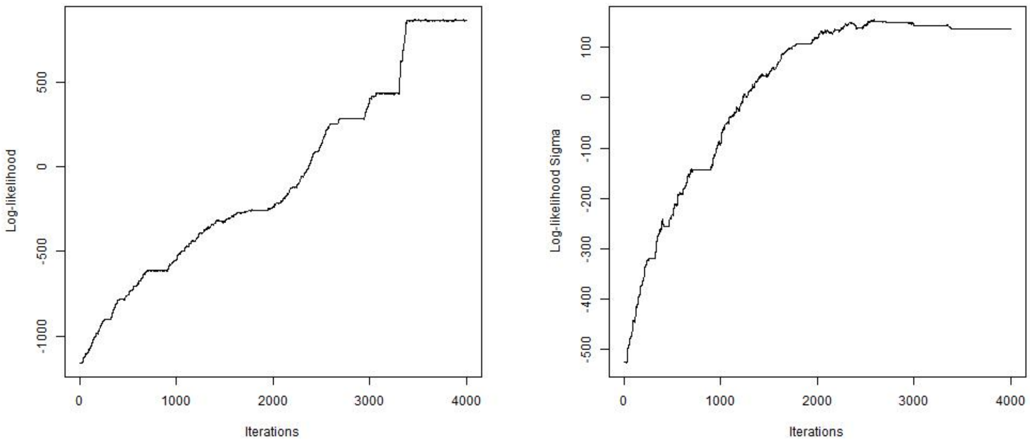

Figure 4.11: Log-probability of the model on the left, and the relative log-likelihood of the covariance matrix on the right. Both graphs represent the log-likelihoods after the burn-in period.

and the number of arrows selected at a 50% threshold, in Figure (4.12).

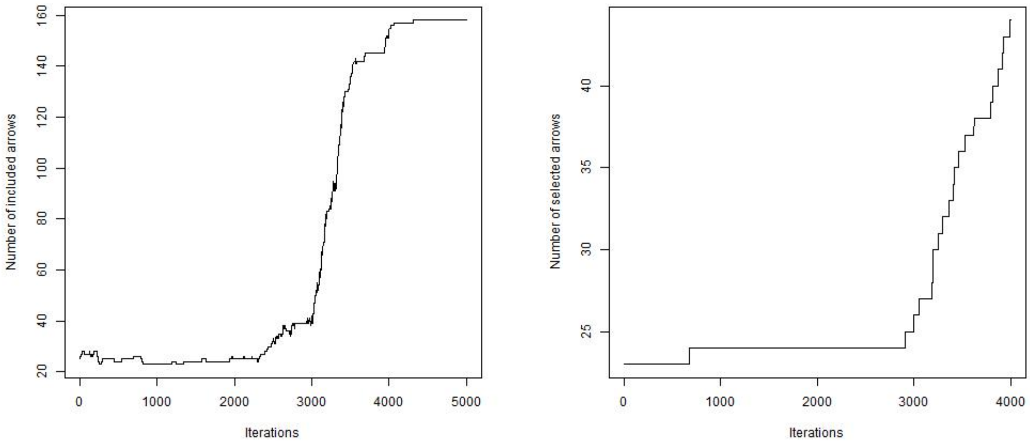

Figure 4.12: Number of arrows included in the model (left panel), and number of arrows selected at a 50% confidence level (right panel).

Moreover, similarly at in Section 4.6.2, the shape of the τ parameter chain

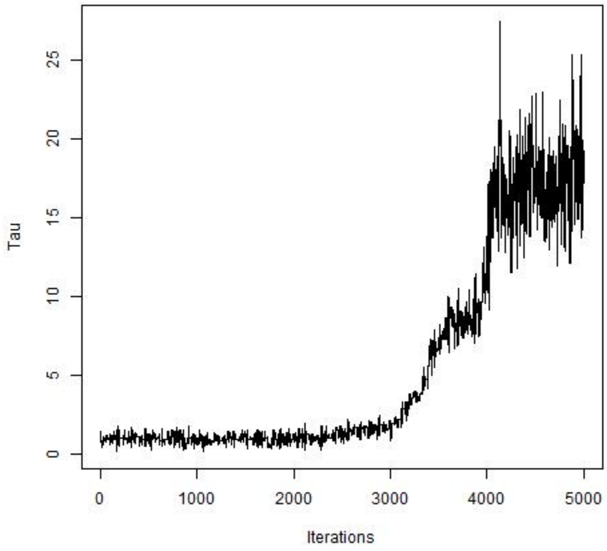

Figure 4.13: Values of τ.

in Figure (4.13) fits the shape of the number of arrows included at each iteration.

These plots suggest that the chain had reached a reasonable configuration, predicting a high proportion of the functional interactions. Indeed, we observe that after about 2,000 iterations, around 25 arrows are included in the regulatory network, with 24 having been for at least 50% of the time since the burn-in. then, the number of arrows selected suddenly and drastically increases, until reaching almost 160, meaning that almost all the gene-miRNA interactions are said to be functional. Consequently, the number of arrows reaching the threshold $t_h = 0.5$ also increases. Similarly, even if it does not appear on the graph (4.10), if we had run the chain for longer, we can expect that the number of arrows reaching the threshold $t_h = 0.75$ would have also much increased, resulting in most of the potential arrows getting predicted.

In this scenario, when we update and estimate the covariance matrix at each iteration, we observe a problem of identifiability. We will discuss it in the next section, and will try and propose few ways to rectify this problem.

Constraints on when to update the covariance matrix

As we have just seen in the last section, the estimation of the covariance matrix leads to some new issues. Indeed, even though it looks like we are about to converge to the true distribution, when the number of included arrows is close to the true number, the chain reaches a point when the number of arrows drastically increases. When we observe the graphs of the number of arrows included (Figure 4.12) and the log-likelihood (Figure 4.11), we notice that this increase takes place after around 2,000 iterations, when the log-likelihood is close to -500. That log-likelihood is higher than the true one, which we can see in Figure 4.5, is around -1370. We then assume that estimating $\boldsymbol{\Sigma}$ without any constraints makes the chain pass beyond the convergence point, because of a problem of identifiability and estimation of the regulatory network, in the sense that the covariance matrix, and its update, overshadow the other parameters of the model, such as $\boldsymbol{R}$, or the γ_g's, as we are about to explain in more details.

Indeed, an update of $\boldsymbol{\Sigma}$ implies many changes of the model. For instance, the update of all the values in the matrix implies changes in most of the quantities involved in the variable selection, such as $\boldsymbol{C}, \boldsymbol{U}, \boldsymbol{Q}$... Hence, updating the covariance matrix too often, at each iteration, might not give enough time to the other parameters of the chain to get adapted to this new value. That is why we decide to update $\boldsymbol{\Sigma}$ only every G iterations. By doing so, we give time to other parameters to actually be affected by a possible new estimated covariance matrix. That means that between two potential updates of $\mathbf{\Sigma}$, each gene will have on average one element of $\boldsymbol{R}$ updated, also resulting in one update on average of the parameter γ_g of each gene g.

The second issue we need to work on is the log-probability level. We want to make sure the chain does not reach a point where the model explodes, all the potential interactions becoming functional. That is the reason why we decide to stop updating $\mathbf{\Sigma}$ after a certain point, the other three steps of the Metropolis-hastings algorithm remaining. Several solutions can be considered for that stopping point. Under the assumptions of the model, all the original and true configurations simulated will have a likelihood in the same range, with respect to the dimensions of the model. Then, we can decide to stop updating $\mathbf{\Sigma}$ once its likelihood reaches a value close to the mean likelihood of the matrices sampled under the assumptions. However, while running the chain, we can notice that $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \mathbf{\Sigma})$ seems to converge to its true level slower than $\pi(\mathbf{\Sigma})$ does. That is why we choose to stop the update of $\mathbf{\Sigma}$ once the sum $\log(f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \mathbf{\Sigma})) + \log(\pi(\mathbf{\Sigma}))$ reaches a suitable value, similar to what we observe by simulating several regulatory networks. That is the option we chose for that simulated case study. However, if we actually have no prior idea, we can decide of a second threshold, after which we keep the current covariance matrix. This second threshold can be set up around 75% of the whole chain, leaving enough iterations to check for convergence after that point.

The following results and graphs have been computed under these modifications: $\boldsymbol{\Sigma}$ is updated only every G iterations, and is stopped being updated once $\log(f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})) + \log(\pi(\boldsymbol{\Sigma})) > -1100$, a likelihood level slightly higher than the true value. We observe in Figure (4.14) that the number of included arrows remains between 23 and 32, a number matching the prior assumptions. That number has some variations, before stabilizing between 25 and 27, once $\boldsymbol{\Sigma}$ is no longer updated, after $\simeq 28,000$ iterations according to Figure (4.15).

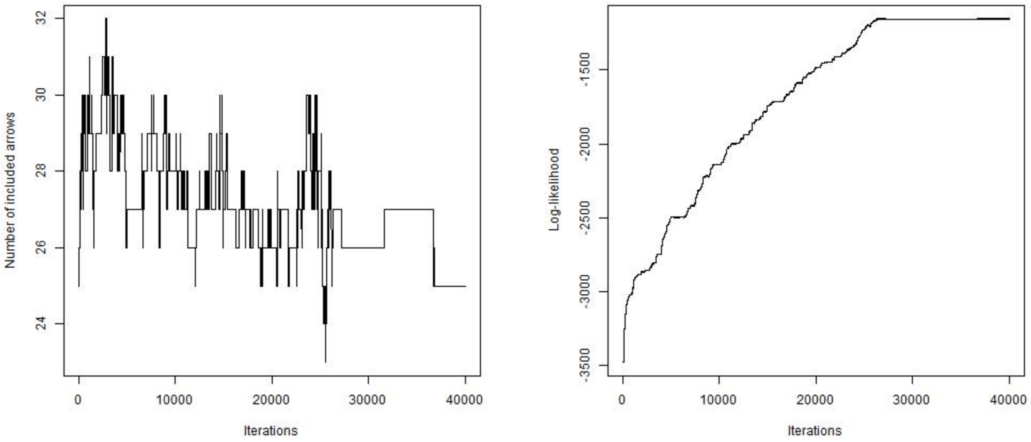

Figure 4.14: Number of arrows included in the network (left panel), and the log-likelihood of the model (right panel).

We can also verify that after that point, the log-likelihoods, of both $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \boldsymbol{\Sigma})$ and the whole model, no longer vary (they actually do, but slightly). That shows the importance of the covariance matrix in the update of the chain, conditioning all the other parameters, including the table of interactions. That also arises one disadvantage of the approach, which is that in practice, we can not really know when to stop its update. We only know that it has to be done once we reach a region close to the true distribution, but this is quite a vague precision. However, we have seen that this is necessary, to avoid the model to explode. Moreover, providing we do not pass far beyond the true distribution level, several chains run showed that the exact moment of when we stop updating $\mathbf{\Sigma}$ does not have major changes in the regulatory network, the most influential arrows having been predicted for a long time.

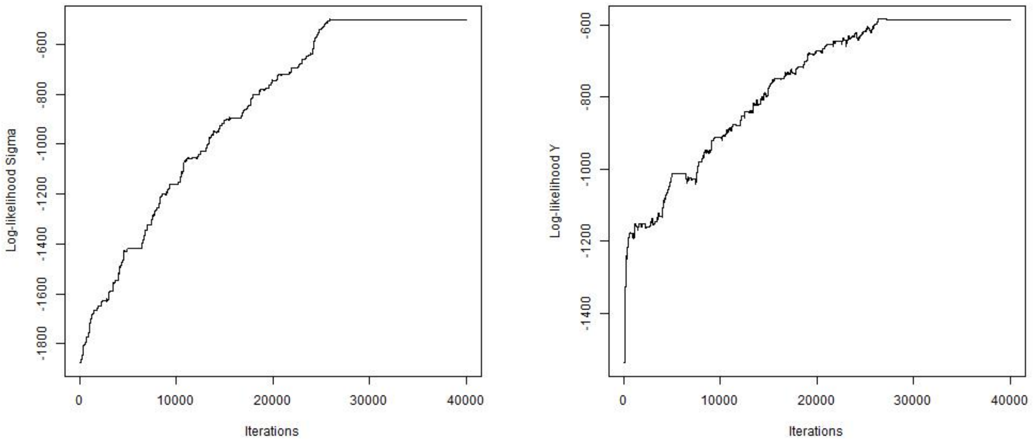

Figure 4.15: Log-relative likelihood of $\mathbf{\Sigma}$ (left panel), and log-likelihood $f(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{R}, \mathbf{\Sigma})$ (right panel), after the burn-in.

Finally, we look at the selected arrows, given their proportion P_{gm} of presence in the model. In Figure (4.16), we observe that 27 and 25 arrows are predicted with respective thresholds $t_h = 0.5$ and $t_h = 0.75$. At $t_h = 0.5$, 24 out of the original 25 arrows are correctly predicted, leading to 3 false positives. The only false negative, with $P_{gm} \simeq 0.05$ is the arrow with $\beta_{gm} \simeq 0.86$, the one which was not predicted even when we ran the chain with the known covariance matrix. At $t_h = 0.75$, 23 out of the 25 predicted interactions were actually in the original network, with 2 false positives. In addition to the previous one, the second missing arrow is the arrow whose coefficient $\beta_{gm} \simeq 0.71$, which is present in the model slightly more than 50% of the time. That was already the arrow with the second lowest P_{gm} in the first analysis with the true $\mathbf{\Sigma}$. Those results, even if not perfect, and not as good as the ones with the true covariance matrix, are still accurate enough, and especially more accurate than when assuming the genes independent.

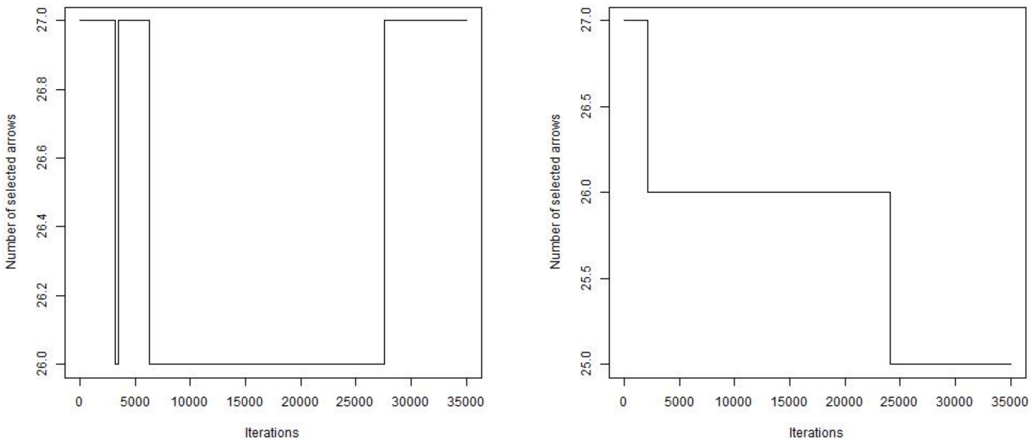

Figure 4.16: Number of arrows selected, with a threshold of 50% (left panel), and 75% (right panel), after the burn-in.

Another approach: multiple-speed chain

An inconvenience of the previous methodology is that, even if the starting $\boldsymbol{R}_0$ is a priori the best guess, we can not be sure of that statement when we are confronted to real data whose we might ignore the distribution. Hence, the start of the second chain is influenced by the first chain, and then depends on that choice of starting configuration. Moreover, it implies that before inferring the regulatory network, we only use the data and the starting choice of parameters to estimate $\mathbf{\Sigma}$, when we then infer the other parameters and $\mathbf{\Sigma}$ in the second chain. A possibility could be to run the first chain, plug-in the output value of the covariance matrix in the second chain, and then infer the regulatory network with that value of $\mathbf{\Sigma}$, without updating it again. However, we would still be unsure of its estimation, as it would depend on the starting choice of $\boldsymbol{R}_0$, which may be different from the real $\boldsymbol{R}$.

Those are the reasons why we may find more suitable to run only one chain, and two with one depending on the other one. However, in order to improve the convergence of all parameters, and especially $\mathbf{\Sigma}$, we use a multiple-speed process, in the sense that the frequency whom $\mathbf{\Sigma}$ is updated and estimated varies:

- in the burn-in, $\mathbf{\Sigma}$ is updated at each iteration. That allows a fast start of the chain to converge to the true distribution.
- after the burn-in, $\mathbf{\Sigma}$ is only updated every G times, now allowing the other parameters to adapt themselves to the new value of $\mathbf{\Sigma}$.
- at the end of the chain, for example the last 20%, we no longer update it, assuming we are now approaching convergence, and we then infer the other parameters, in particular $\boldsymbol{R}$, with the last estimated value of $\mathbf{\Sigma}$.

We then run a chain of 25,000 iterations, with a burn-in of 5,000, and we stop updating the covariance matrix after 20,000 iterations. The log-relative probabilities of the models visited can be seen in Figure 4.17, when the number of arrows, included and predicted with a threshold $t_h = 0.75$ respectively, are in Figure 4.18.

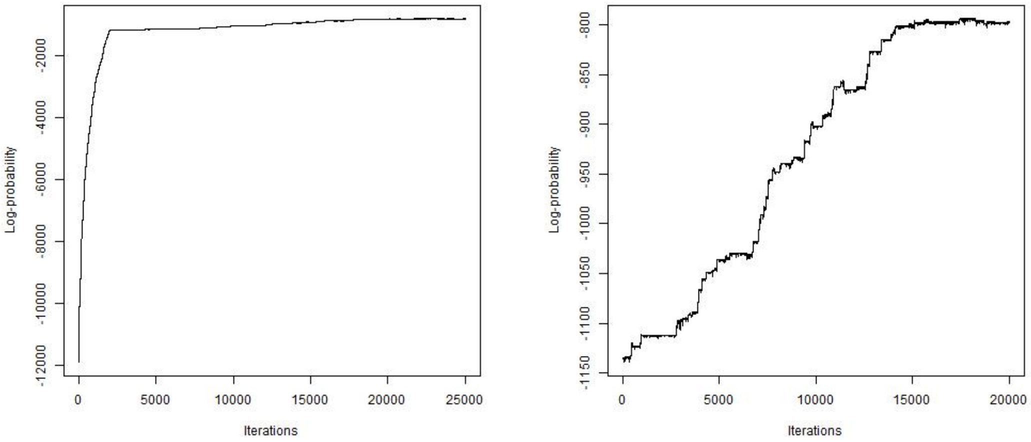

Figure 4.17: Log-relative probabilities of the visited models, through all iterations (left) and after burn-in (right).

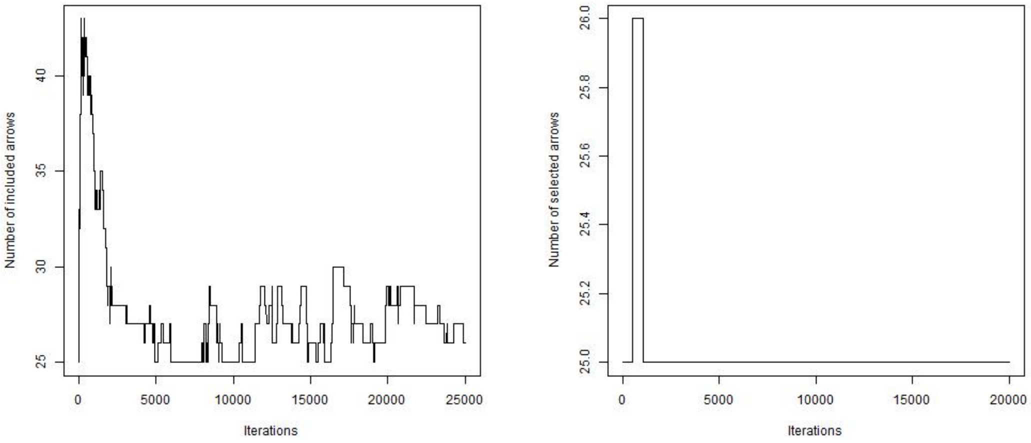

Figure 4.18: Number of arrows included in the network (left panel), and number of arrows selected with $t_h = 75\%$ (right panel), after the burn-in.

The results of this analysis are similar to the ones from the previous approach. From Figure 4.18, we observe that 25 arrows are predicted to be functional with a threshold $t_h = 0.75$. Out of of these 25 arrows, 23 are correctly predicted, resulting then in 2 false positives, as well as 2 false negatives. The 2 false negatives are once again the arrows whose $\beta_{gm} \simeq 0.86$ and 0.71.

Consequently, the results between this multiple-speed approach and the two-chain approach are identical, with the same sensitivity and specificity, at a threshold of selection of 75%. However, the second one does not have the inconvenience of using the data to tune the prior and estimate $\mathbf{\Sigma}$, and does not imply the inference of the network depending on a previous chain. On the contrary, all the parameters are inferred in the same process, the multiple speed of estimating $\mathbf{\Sigma}$ helping the convergence of the chain. That is the reason why we will be using this methodology from now on.

Method	$\mathbf{\Sigma}$ known	Independent	$\mathbf{\Sigma}$ estimated
Sensitivity	0.96	0.44	0.92
Specificity	1	1	0.985

Table 4.4: Comparison of the different chains, with a threshold $t_h = 0.75$

One critic one may have is that the poor sensitivity for the second approach can be expected, as the original regulatory network is built under different conditions and assumptions. Hence these results may be biased by these assumptions. Moreover, as we can see in Section 4.6.1 and Figure 4.3 when we built the network, we deliberately choose a prior for the covariance matrix $\mathbf{\Sigma}$ such that the correlations are positively very strong, which is an extreme situation compared to the assumption of independence. As a consequence, we can admit that it is not really fair to this approach to apply it to a model built under such different assumptions. However, it is likely that some genes are strongly correlated, for example genes involved together in a specific biological function, hence the necessity to assume strong correlations. The inconvenient is that in practice, we do not really know these correlations between genes, we may not even have any prior idea, or a vague one. That is the reason why we specified a vague prior on $\mathbf{\Sigma}$, by choosing $\nu_{\Sigma} = G + 2$, the minimum value such that the expectation of $\mathbf{\Sigma}$ exists. Moreover, that choice of parameters imply that the mode of this distribution, is the identity matrix, a vague prior showing no correlations, as without prior knowledge, we can not know in advance the strength and the sign of the correlations. In practice, if biological knowledge allows us to have a better prior idea of the links between genes included in the dataset, it may be possible and suitable to modify the prior. Similarly, if it is believed that the variables of interest are not correlated, or weakly, then the approach of Stingo et al. would be more suitable, as outcomes should be similar, and the computational cost will be much less expensive.

4.6.5 Monte Carlo study

In the last section, we presented detailed results from one particular case study of simulated data. However, it would be sensible to perform a Monte Carlo study, where we present general details of how our approach performed on certain number of cases. That is what we will discuss in this section. We will describe the details of the analyses, before presenting the performances of the algorithm, performances illustrated by the sensitivity and the false positive rate.

We performed a Monte Carlo study composed of 100 different analyses. Similarly to the particular case study we presented in the previous section, each different network has been created in the same manner, although we allowed the variance of the miRNAs to be higher than the latest, by setting it up to 2 instead of 1.

For each different case study and output, we look at the number of arrows the algorithm predicted, and how many of them are actually correctly predicted. We then compare these two numbers with the known number of functional interactions in the regulatory network. It then allows us to compute the sensitivity of the analysis, as well as the specificity and hence the false positive rate.

Having done for the 100 different cases, we can have a summary of the performances. In Table 4.5, we can read that the average sensitivity is around 91%, with a standard deviation of 6%, while the average false positive rate is around 4% with a standard deviation of 4%.

	Sensitivity	**False positive rate**
Mean	0.91	0.04
Standard deviation	0.06	0.04

Table 4.5: Summary of the Monte Carlo study performed on 100 analyses of simulated data: mean and standard of sensitivity and false positive rate.

These results suggest quite a good performance of the algorithm in general. However, we can also have a look at Figure 4.19, which show the box-plots of both sensitivity and false positive rate to have a better idea of their distributions. On the left panel, we can see that half the sensitivities computed lie between 88% and 96%, with only a very small number of analyses having a sensitivity less than 80%. Similarly, the right panel shows that the 25 and 75% quantiles of the this distribution are respectively 2 and 4%, with only 4 cases which resulted in false positive rate higher than 10%.

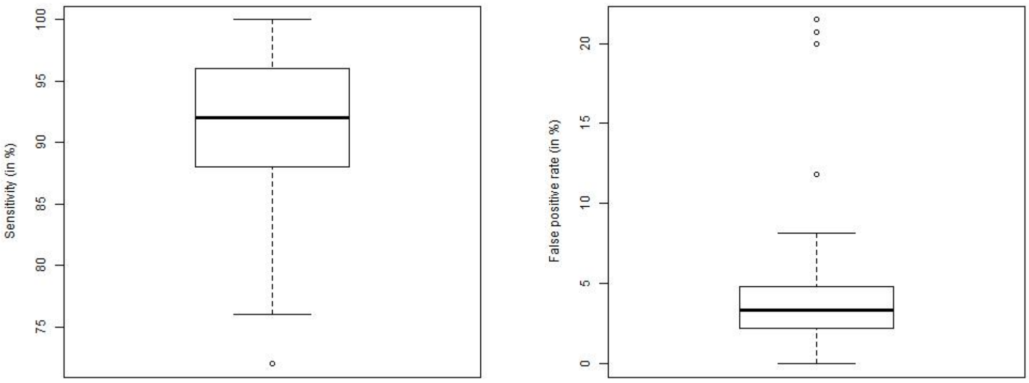

Figure 4.19: Boxplots of the sensitivity (left panel) and false positive rate (right panel) of the Monte Carlo study.

As a small conclusion, we can say that the Monte Carlo study performed in this section provides us evidence that the proposed approach is capable of inferring with good performance, and on a regular basis, a regulatory network assuming correlated variables, suggesting the possibility of applying this procedure to real-life projects.

4.7 Conclusion

In this chapter, we have presented a new approach to infer a miRNA-gene regulatory network, when genes are assumed to be correlated. We then have applied the method to a simulated case study, built under the assumptions of the model. We have applied the method in three different scenarios. The different approaches and regulatory networks can then be compared, as we can see in Table 4.4, by computing the sensitivity and specificity of each chain. First, we ran the chain in the best case scenario, when we know the true covariance matrix. This scenario has the best results, predicting 24 out of the 25 original arrows, if we assume the interaction with $P_{gm} = 0.74$ to be functional. However, this method can obviously not be computed in practice, as it is very unlikely that we know that covariance matrix. In a second scenario, we ran a chain where we assumed the genes independent, by applying the approach from Stingo et al. [58]. In this scenario, the number of arrows included is consistent with the prior knowledge of one or two regulating miRNAs per gene on average. However, that number is subject to many variations, and by computing the proportions of presence P_{gm}, we observe that the number of arrows being selected as functional, with $P_{gm} \geq t_h = 0.75$, is actually quite low, leading to a low sensitivity of 44%. In the third and final scenario, we run the chain with the update and estimation of the covariance matrix $\mathbf{\Sigma}$. As we have seen it in the previous section, we need to introduce some constraints on when to update it, via the multiple-speed process, in order to avoid a problem of identifiability. Once these constraints are applied, the model infers a regulatory network close to the original one, with both sensitivity and specificity higher than 90%. Both these measures are summarized for each method in Table 4.4. In the last section, we also perform a Monte Carlo study on 100 analyses to compute average sensitivity and false positive rate of the proposed methodology, resulting in results that indicate good performance. These results suggest that it is preferable to apply the method with correlated genes, rather than independent genes. Indeed, although the specificities are all equal or close to 1, the sensitivity in the case of independent genes is much lower than in the other two scenarios.

CHAPTER 5

A regulatory network for Acute Coronary Syndromes

In this chapter, the proposed methodology described in Chapter 4 is applied to the Acute Coronary Syndromes data and we draw some regulatory networks, with the genes assumed correlated given the miRNAs, and then independent given the miRNAs, in order to compare the performance of both approaches. However, we will first need to describe the way the original dataset of 897 genes has been divided in several clusters of various sizes. The data consist of genes and miRNAs expression levels, from patients suffering from three different ACS conditions: STEMI, NSTEMI and Unstable Angina. As described in Section 3.2.1, STEMI and NSTEMI refer to myocardial infarction (heart attack) where a coronary artery is blocked by a blood clot, and then the blood flow is stopped. The difference between these two conditions is that the artery which supplies the heart muscle is totally blocked in STEMI, and partially blocked in NSTEMI. Apart from these two severe conditions, the Unstable Angina condition is less serious, in the sense the blood flow is reduced by the clot, but still effective. Then, we will discuss the resulting results and networks, in an attempt to interpret the biological meaning of their similarities and differences. By doing so, we can make suggestions which can benefit the biologists and medical researchers, in order to find a biomarker, source of a disease, and maybe provide a medical diagnostic sooner.

5.1 Clustering of the data

As previously described in Section 3.3.1, several methods have been used to narrow down our dataset of thousands of genes to $G = 897$. Unfortunately, this number of G is still too high for us to perform an efficient analysis of our dataset. Indeed, this number would imply many computational problems, such as the inversion of the $G \times G$ covariance matrix, the computation of the cdf $\Phi_{G*M}(\mathbf{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U})$, and of course a very long running time. This is the reason why we still wish to separate the genes of interest into smaller clusters of correlated genes, whom size will be reasonable to perform our methodology.

We then apply the AutoSOME algorithm, described in Section 3.3.2. We need to adjust some parameters of the algorithm, in particular the p-value threshold, in order to obtain clusters of reasonable sizes. We indeed do not want to obtain one big cluster with most of the variables in there, as the main difficulty of the size of variables will remain. On the other hand, many clusters composed of a very few genes are not desirable either, as this will involve many analyses to run, but it would particularly eliminate many strong correlations of genes which might be in the same cluster.

We finally obtain six clusters of different sizes. We choose to have the number of genes in clusters higher than the number of miRNAs, $M = 13$. This results in 6 clusters of sizes: 36, 36, 32, 29, 19 and 16. All the others will then be assumed independent, and grouped in the same seventh cluster. The approach of Stingo et al. [58] can thus be applied to this dataset. For more discussion about this case, see Section 5.3.1.

5.2 Application to the data in clusters

Having aggregated the genes in several clusters, we can now run our MCMC chains to these different datasets. We then run 18 chains, one for each cluster, and for each condition. The parameters are set up according to the description given in Section 4.5. We remind that the number of patients varies with the condition: we have $N = 16$ for the STEMI condition, $N = 19$ for the NSTEMI condition and $N = 19$ for theUnstable Angina condition. These values of N remain the same for each cluster under the given condition. For computational reasons, we chose to visit only the potential interactions for which the target score s_{gm} is positive, assuming the zero scores predicted by TargetScan imply a biological near impossibility for the miRNA to bind to these target mRNAs. This implies that the number of potential interactions are, for each cluster, respectively: 75, 93, 56, 57, 32 and 30. In our case study on simulated data, we observe that we obtained satisfying results with a total of 25,000 iterations, with a burn-in of 5,000 iterations, and stopping the estimation of $\mathbf{\Sigma}$ after 20,000 iterations, when the number of potential arrows was 160 (no zero scores in S). Having less configurations to visit, we then decided to conserve these numbers of iterations for clusters 1 to 4, even if the dimensions are higher, and so are the covariance matrices. For clusters 5 and 6, which are smaller, we set up a burn-in of 3,000 for a total of 20,000 iterations, stopping the estimation of $\boldsymbol{\Sigma}$ after 12,000 iterations.

The output of these chains will then be used to construct networks of gene-miRNA interactions. In Figures 5.1, 5.2 and 5.3, we can see the number of arrows included in the visited models through the chain, under each of the three conditions: STEMI, NSTEMI and Unstable Angina. In these Figures, each graph corresponds to a different cluster, from cluster 1 (top left) to cluster 6 (bottom-right). If desirable, the log-relative probabilities of the visited models, both through the whole chain and only after the burn-in, are available in the Appendix, Section A.3, Figures A.1 to A.6. We can see that the highest number of arrows appears for the STEMI condition while the UA condition shows the lowest number of arrows. As we will see, it will result in bigger or smaller regulatory networks, depending on the condition. Based on this, we can compute, for each potential arrows between gene g and miRNA m the proportion of MCMC samples P_{gm} for which $r_{gm} = 1$. These proportions will then be used as estimations for the posterior probabilities $\mathbb{P}(r_{gm} = 1|\boldsymbol{Y}, \boldsymbol{X})$ of presence of interactions. The regulatory networks, for each condition, are then built by selecting as functional all the arrows whose probabilities of presence reach the given threshold t_h, $P_{gm} \geq t_h$. That threshold can vary, the higher it is, the higher the confidence of the arrows of interest is. The networks shown in Figures 5.4, 5.5 and 5.6 have been built using $t_h = 0.9$. In these networks, the miRNAs are colored in yellow, while the target genes remained white. The summaries of these networks (number of arrows, genes and miRNAs involved) are given in Table 5.1. We can for example read that the regulatory network created for the NSTEMI condition contains 67 interactions, between 40 different genes and 12 different miRNAs.

Condition	**STEMI**	**NSTEMI**	**UA**
Number of arrows	98	67	57
Number of genes	50	40	32
Number of miRNAs	12	12	12

Table 5.1: Summary of the regulatory networks for each condition. For example, the regulatory network created for the STEMI condition contains 98 interactions, between 50 different genes and 12 different miRNAs.

From this table, and the graphical networks, we observe that the number of arrows, hence also of genes, is higher for the STEMI condition than for the other two. Otherwise, they all involve the same number of miRNAs. However, the interpretation of these networks, and the potential conclusions will be discussed in more detail in Section 5.4.

On the other hand, when we perform the approach from Stingo et al. [58], we observe that very few arrows are predicted, and only if we use a low threshold t_h, less than 0.5. Indeed, after the burn-in, the number of arrows in the visited models never exceed 20, with a lot of variation, suggesting that these arrows are not always the same. Thus, as we want to predict some potential gene-miRNA interactions, and since biology also suggests that genes are correlated, we decide not to apply this approach to the data, including the remaining cluster.

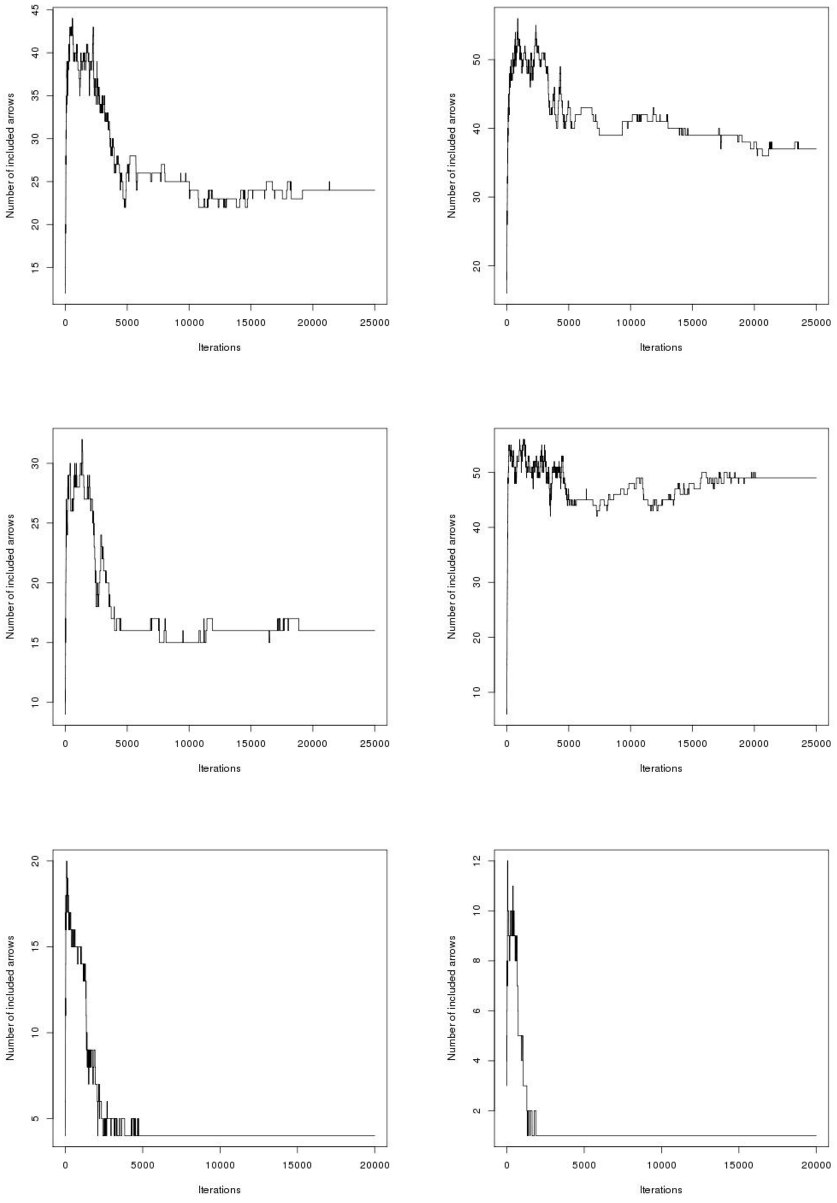

Figure 5.1: Number of arrows included in the visited models, STEMI condition

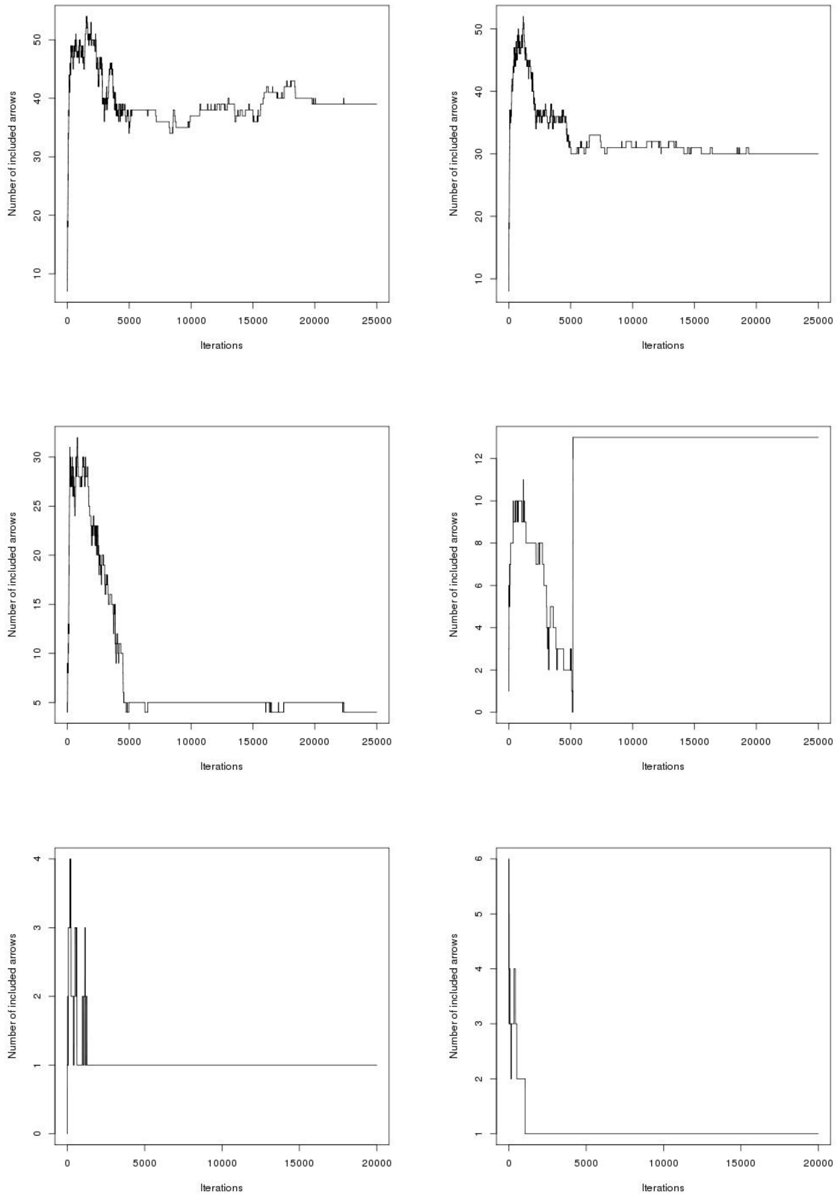

Figure 5.2: Number of arrows included in the visited models, NSTEMI condition

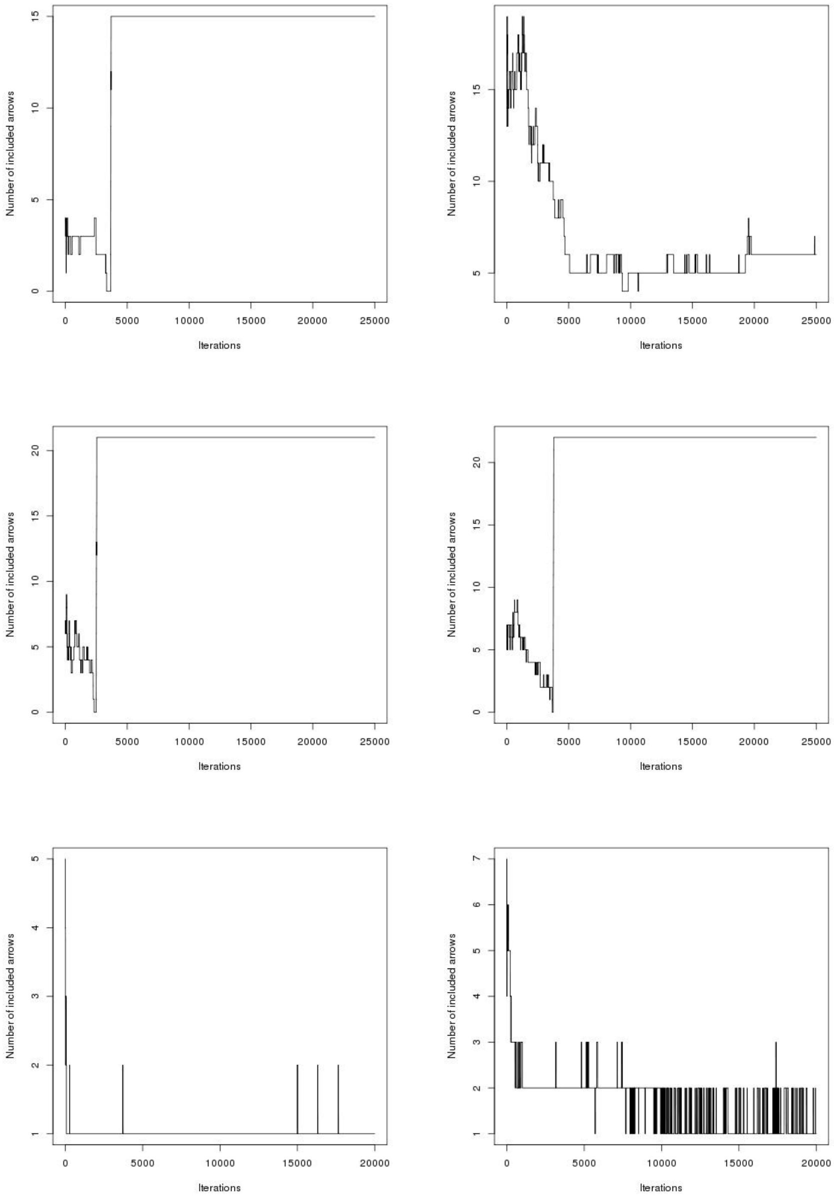

Figure 5.3: Number of arrows included in the visited models, Unstable Angina condition

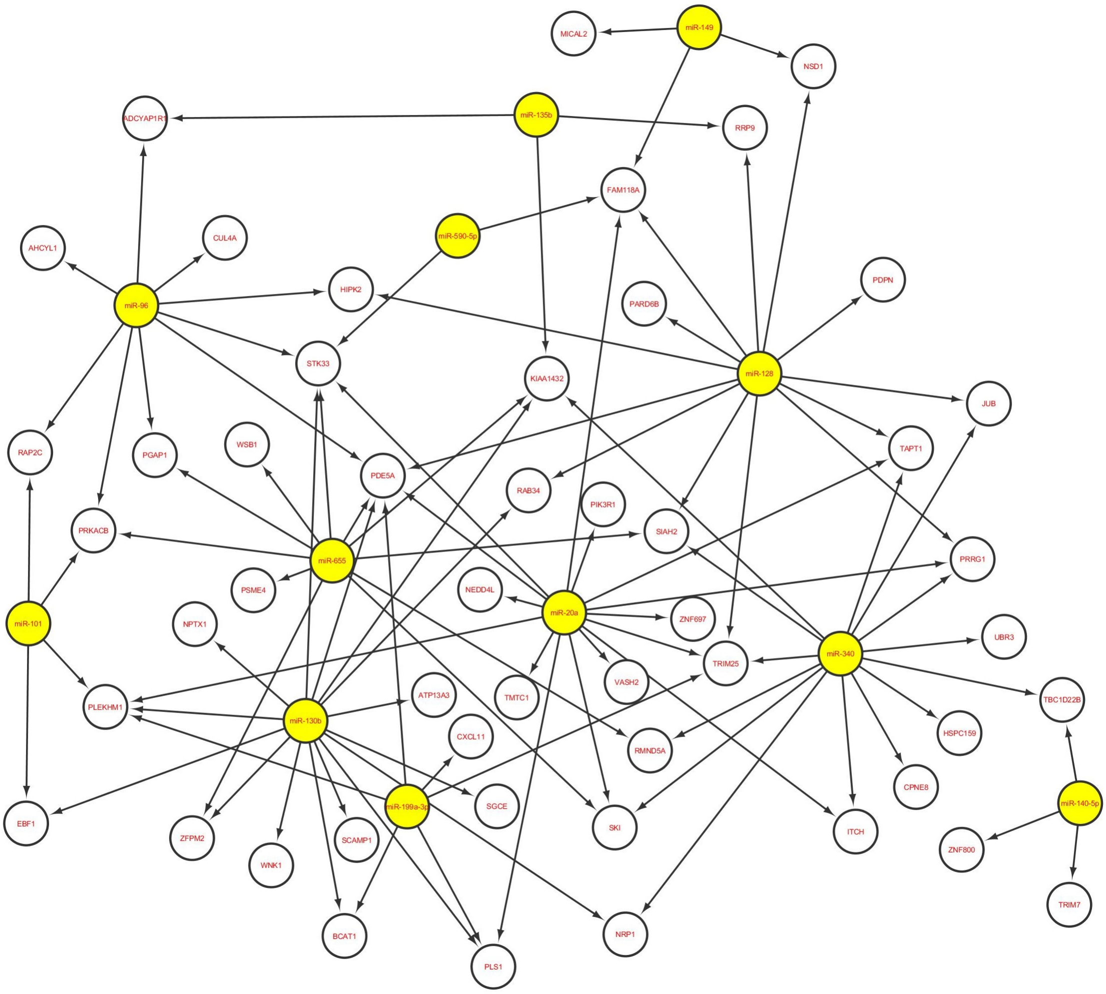

Figure 5.4: Network for the STEMI condition: 98 arrows involving 50 genes and 12 miRNAs.

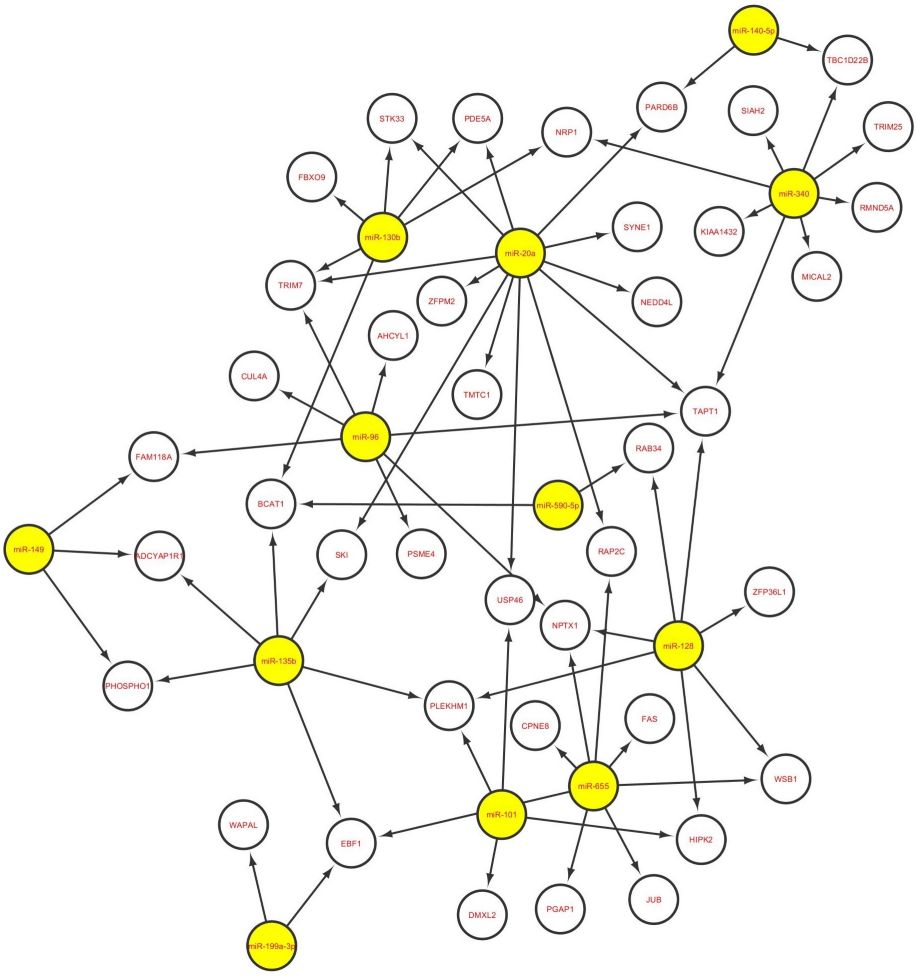

Figure 5.5: Network for the NSTEMI condition: 67 arrows involving 40 genes and 12 miRNAs.

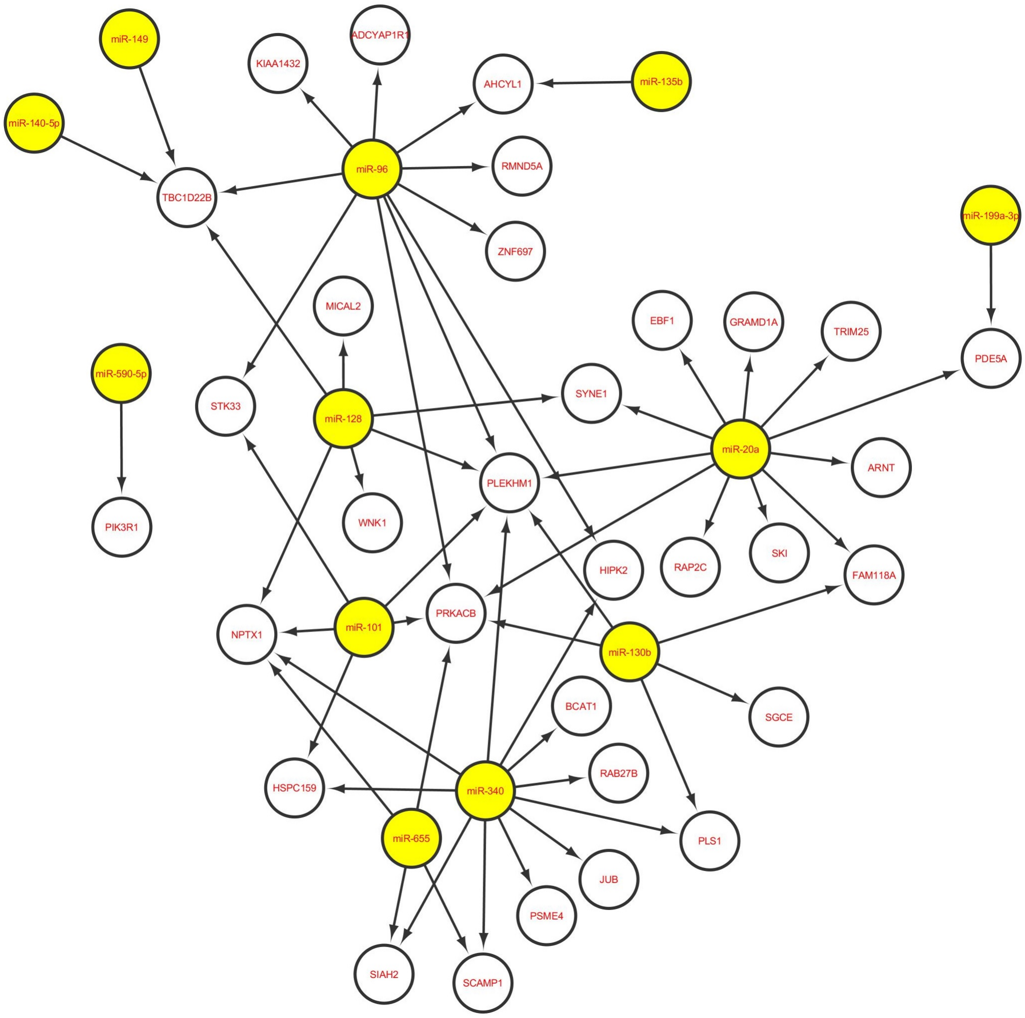

Figure 5.6: Network for the UA condition: 57 arrows involving 32 genes and 12 miRNAs.

5.3 Discussion

5.3.1 Special case: cluster of independent genes

Before discussing the advantages and inconveniences of the proposed methodology, in particular about the estimation of $\mathbf{\Sigma}$, we briefly discuss how to deal with the remaining cluster of independent genes upon the miRNAs.

The proposed methodology does not appear to be directly applied in this case. Indeed, the estimation of the covariance matrix and the sample of $\mathbf{\Sigma}_{new}$ would imply a non-diagonal covariance matrix. Hence we need to consider another approach. The intuitive possibility would of course be to apply the method of Stingo et al. [58]. However, as we mentioned in Section 5.2, this method only predicted with low confidence a small number of arrows for the genes in clusters, we decide not to apply this approach to our dataset. A second possibility could be to apply our approach, with $\mathbf{\Sigma}$ diagonal, with the variances updated and estimated one after the other, rather than estimating the matrix. However, the eventual high number K of functional arrows may imply computational problems when we need to compute the K-dimensional cdf $\Phi_{G*M}(\mathbf{0}, -\boldsymbol{U}\boldsymbol{V}^T, \boldsymbol{U})$.

5.3.2 Estimation of $\mathbf{\Sigma}$

The biggest challenge of the approach is how to deal with a large covariance matrix, hence the difficulty to estimate it. The computational cost and the long running time would obviously be the first ones coming to mind. Although $\mathbf{\Sigma}$ has an influence on the regulatory associations and thus needs to be estimated, we shall keep in mind that the priority is the estimation of the interaction matrix $\boldsymbol{R}$, and not of $\boldsymbol{\Sigma}$. This is the reason why we stop updating it after some time. We assume that the covariance matrix at that time is now reasonably estimated and can be used as the true one, in order to then focus on the estimation of $\boldsymbol{R}$. Moreover, updating it involves an update of all the variances and covariances, hence all the other parameters of the model, like $\boldsymbol{R}$ and $\boldsymbol{\gamma}$, need time to adapt to the new value of $\boldsymbol{\Sigma}$. As a result, after the burn-in, the update now takes place only G times, such that on average, each γ_g had the possibility to get updated between two updates of $\boldsymbol{\Sigma}$. However, this number G can be modified, as we may wish to estimate $\boldsymbol{\Sigma}$ every 100 iterations for example. That would give more time to the chain to learn more from the data and to be adapted to the new covariance matrix. The main disadvantage of this possibility is that it would imply an even longer running time, hence it is more expensive computationally. This is why we opted to estimate it at each G iterations. We also have to point out that G has to be manageable. Indeed, if we had bigger clusters, with for example 200 or 300 genes, computational issues could emerge, and we might not be able to perform an efficient analysis of the data.

A difficulty is also observed on the diagnostic of convergence. Indeed, with these dimensions, the relative probability of the visited models may be affected by the impact of $\boldsymbol{\Sigma}$. Moreover, we specified a vague prior for $\boldsymbol{\Sigma}$, with the mode of the Inverse-Wishart being the identity matrix. If we had more biological knowledge about the eventual relationships between our variables, we would be able to specify a more informative prior, with covariances different than 0, and then the chain may converge quickly to the true distribution. For example, if we believe that our variables are positively correlated, we may choose to set the covariances for the prior to a positive value, according to how strongly we believe the variables are correlated.

5.4 Interpretation of the different networks

5.4.1 Precautions to take into account

In this section, we will discuss some results we can observe from the three different networks built, which are shown in Figures 5.4, 5.5 and 5.6. We will also attempt to give a biological interpretation of these results, which could be used to devise a medical diagnostic, or to improve our knowledge of these conditions. The purpose of this section is not to draw final conclusions, but only to make suggestions. Indeed, one first reason is that due to the complexity of the model, we can not claim to present concrete results. As we have previously discussed, there are still some aspects we are not able to fully understand and explain, for example we can not have a certain prior knowledge of the correlations between genes, or even how the miRNAs regulate the genes. In our model, we assume that they down-regulate the genes. However, as it is believed that is the case when the regulation is direct, other studies make the assumption that the regression coefficients can be positive, especially when the regulation is indirect, when the miRNAs control the genes through one or more entities. That is the case of Ročková [52], where a miRNA regulatory network is infered via a factor augmented multivariate regression, and where the variable selection is performed by an EM algorithm.

Furthermore, we shall point out the small number of samples, a maximum of $N = 19$ for the NSTEMI and UA conditions. That small number of measurements or patients might imply a lack of power of our methodology. Also, we made the choice to explore only the potential interactions predicted by the TargetScan algorithm. However, as we mentioned, there exists other algorithms, which can be combined in the prior model, as in equation (4.5). By using other sources, we would have increased the number of potential arrows, and then our graphs may also have been larger.

Our comparison and interpretation of the networks will consist of noticing their main similarities and differences: which arrows are mostly present, what does it mean if an arrow is present in a network of one condition but not in the others, and so forth. As it is very difficult, if not impossible, to create biology out of these, our suggestions can be used in a preventive aim. If a patient is admitted to the hospital after a heart attack, and if their blood samples show relationships we can observe in our graphs, we may be able to diagnostic their condition, and possibly to assess the risk to aggravate that condition. For a better understanding about comparison between graphs, statistical methods can be found in [54] and [55].

Another difficulty about drawing conclusions from these results is that we do not have available data from healthy patients, only expression levels from patients suffering from one of the three conditions. That absence of control group then does not allow us to have a regulatory network under a healthy condition, thus we can not say if what we observe is specific to ACS, or if these arrows are also observed in healthy people. Hence the necessity to be measured in our suggestive conclusions. Moreover, some of the genes in our dataset, even if they got through our several filters, may have already been identified as genes involved in complete different biological processes, a function which is not related to ACS.

Finally, we point out that the length of the arrows does not imply a stronger of weaker interaction, as the graphs have been designed for reasonably clear viewing. All the arrows present in these graphs are arrows which have been included in the chains more than 90% of the time, $P_{gm} \geq 0.9$.

5.4.2 Main observations from the graphs

The first observation we made in Table 5.1 is that the number of miRNAs involved in the networks is always 12. Actually, we can also notice that these are always the same set of 12, the last miRNA, miRNA-134, never regulating any gene under any condition. That may suggest that this miRNA does not have a role in ACS. Also, the number of arrows is quite consistent with the prior suggestion that most genes are regulated by one or few miRNAs, the average number of regulating miRNAs per gene in our networks being between 1.7 (NSTEMI) ans 2 (STEMI).

The main similarity we then need to point out is if there are arrows present under all three conditions. We can indeed observe six arrows always functional, which are presented in Table 5.2: Also, if we compare the two

miRNA	**Gene**
miR-20a	SKI
miR-20a	PDE5A
miR-96	AHCYL1
miR-340	SIAH2
miR-140-5p	TBC1D22B
miR-101	PLEKHM1

Table 5.2: Interactions present under the 3 conditions

more severe conditions, STEMI and NSTEMI together, with thes less severe one Unstable Angina, we find 25 mutual arrows between these two groups. Then, without any control group, it is difficult to affirm with confidence if these interactions are characteristic of ACS, and reflect a possible genetic source of the disease, or if they are interactions which might also be functional in healthy people.

Another point to highlight is the number of common arrows between the two severe conditions, STEMI and NSTEMI. We can count 21 of them, and 17 out of these 21 are shared by only 4 miRNAs: miR-20a, miR130b and miR-340 and miR-128. We can then assume that these genes, and especially these miRNAs may play an important role in ACS. Indeed, these interactions are present in both severe conditions, but are not in the less severe one. Then, if a patient has a heart attack or chest pain, and if we notice in his blood samples some relationship between the genes and miRNAs involved in those 21 arrows, (for example high levels of these miRNAs and small levels of genes) we must have in mind that this patient may develop one of these two serious forms of Coronary Syndromes. Once again, that would be a guideline, a suggestion for further test and samples, as a more advanced investigation may show a different reason for the heart attack or chest pain, a reason which may not even be genetic. Also, when we perform this first diagnostic, we need to take into account the moment of sampling, how long it was after the acute chest pain or heart attack. Indeed, the pain may have decreased significantly, so has a potential evidence of gene-miRNA interaction if the patient is sampled a long time after their incident. That is why further investigation will still need to be performed.

A last observation we may point out is that in all three graphs, some miRNAs, and sometimes some genes, play a central role in the graph. For example, for all three networks, between 72 and 82 % of the arrows involve a subset of only six miRNAs: miR-20, miR-340, miR-130b, miR-128, miR-96 and miR-655. We can notice that 4 of these were mentioned in the source of the STEMI and NSTEMI mutual arrows. Similarly, some genes may be regulated by many miRNAs under a given condition: PDE5A (6 arrows) and STK33 (5) in STEMI, TRIM7 and BCAT1 (3 arrows) in NSTEMI, PRKACB (5 arrows) in UA, and so forth. We can also name PLEKHM1 which is relatively well important in all conditions, with respectively 4, 3 and 6 arrows(STEMI/NSTEMI/UA). This may as well give us some directions, hints to look further, in order to target a potential genetic source causing ACS, subject to further medical investigation.

5.5 Conclusion

In this chapter, we described some results we have obtained from applying our methodology to the Acute Coronary Syndromes data. It resulted in the construction of three different regulatory networks, one for each ACS condition. We highlighted the main similarities and differences between them, in particular the genes and miRNAs which seem to play an important role in ACS, due to the number of arrows they are involved in. We have also proposed some biological interpretation of these results, which may help to diagnostic the source of a possible disease. However, as we have pointed out, these suggestions will need further investigation to be verified, as our results can not be taken as definite conclusions, due to several reasons: complexity of the model, absence of a control group to compare with healthy patients, lack of knowledge about the gene-miRNA interactions, etc.

Even if the final results depend on the clustering method and the dimensionality reduction used (other methods would have probably selecting a different subset of data), the methodology does not rely on them. Indeed, it is possible to use other filtering methods one may find more suitable, and then perform the proposed methodology.

Similarly, the model can be applied to other types of data, biological or from another field. An example of another type of application in genomics would be data obtained from RNA sequencing, a technique that uses the capabilities of next-generation sequencing. As described in Section 2.1.6, more details about RNA sequencing and next-generation are available in [7], [17] and [41].

CHAPTER 6

Conclusion and Discussion

6.1 Conclusions

In this thesis, we have proposed a Bayesian graphical model that infers a gene-miRNA regulatory network by integrating expression levels of both miRNAs and their potential gene targets, and that also integrates sequence and structure information via the prior probability model. The regulatory network is inferred using a stochastic search variable selection, via an MCMC procedure, where the different parameters of the model are estimated and updated. In this approach, genes are assumed correlated given the miRNAs, while they are often assumed independent given the miRNAs, sometimes for practical and computational reasons. These correlations need to be taken into account, as in real life, in the cell, genes (or at least some groups of genes) are believed to be correlated. However, also for practical and computational reasons, the large original dataset needs to be filtered, and then separated in several clusters. While variables within a same cluster are assumed correlated, the different clusters however are assumed independent of each other given the miRNAs.

We considered a case study on simulated data to evaluate the performance of our approach, and to compare with the approach of Stingo et al., where the genes are independent upon the miRNAs. Then, we considered an experimental study on Acute Coronary Syndromes, consisting of three different conditions. The analysis involved 13 miRNAs and 168 potential target gens, split in 6 clusters of various size. The aim was to identify a small set of potential miRNA-gene interactions under each condition, and compare the resulting networks. The main similarities and differences, for instance an edge present in all networks or only in one, can give future ideas and guidelines for further medical investigation, in order to identify biomarkers related to a disease. However, these results and comments need to be taken with caution, as the small number of samples and the absence of a control group of healthy patients do not allow us to draw definite conclusions, but only suggestions.

The proposed modeling approach is general and can easily be applied to other types of data (next-generation sequencing [17]) and network inference, like gene-gene networks [59]. The prior model also allows to integrate and combine different sources of prior information.

A limitation of such a study in genetics is that our knowledge of biological processes remain incomplete, and can therefore imply errors. For example, the prior information can vary from the source we use. Some potential arrows predicted by TargetScan might not have been predicted by another algorithm and then would not have been included in our study, and vice versa. It is also challenging to establish an accurate prior about the covariance matrix, as we can not know how all the genes are linked with each other. Under our methodology and assumptions of correlated genes, this covariance matrix plays an important role, and therefore deserves further investigation, especially related to its optimization and estimation.

6.2 Extensions and future work

Extensions of the proposed approach are possible. In our study, we mostly aimed to identify potential arrows, however it may be desirable to estimate the strength of these arrows. As we mentioned it in Section 4.3.2, one possibility would be to estimate the regression coefficients $\boldsymbol{B}_R$. The more negative a coefficient is, the stronger the corresponding arrow is assumed.

Moreover, we may be interested in studying the impact of some arrows in different networks. That is the reason a comparison between graphs, obtained from different conditions, may be desired in order to identify the structural similarities and differences between the graphs. This may also help us to identify the source of the disease. In [54], Ruan compare multiple graphs via a procedure based on Generalized Hamming Distance, and also provide a review of statistical methods for comparing graphs; see also Ruan, Young and Montana [55].

We also need to keep in mind that our knowledge of biological processes in the cell, including the gene and miRNA regulation is still unclear and incomplete. It is possible that miRNAs may also regulate each other, and can regulate a target gene indirectly, via another gene or miRNA. In our approach, we assumed the miRNAs independent, and the gene-miRNA interactions to be direct and directed from the miRNA to the target gene, without passing via another node. As it is also believed that miRNAs down-regulate the genes, we integrated this prior knowledge by imposing negative regression coefficients.

However, some approaches infer regulatory networks under different assumptions or methodologies. Ročková in [52], and Ročková and George in [53], infer the regulatory network via a factor augmented multivariate regression, and perform an EM algorithm for the variable selection. In [24], Huang, Morris and Frey proposed a variational learning method using Kullback-Leibler divergence and EM algorithm to obtain a set of functional miRNA-gene interactions. In [59], Stingo and Vanucci proposed a variable selection with a Markov random field prior to classify subjects according to phenotypes via gene expression data, where they consider gene-gene networks as undirected graphs. Another different approach is proposed by Johnson, Welcker and Bass [28], where they develop a Dynamic Linear Model to identify candidate miRNAs regulating their target genes. Comparison with these methods could be very interesting, in particular to evaluate the performance of each methodology, both in terms of network estimation and computational performance. For example, in [59], the authors consider some similar priors as our approach, such as the Inverse-Wishart prior for the covariance matrix of genes in the same group, but use a different one (Markov random field) to map the connections between nodes. On the other hand, Expectation-Maximization Variable Selection and variational learning approaches may show better computational performances than our intensive MCMC procedure. However, we may need to modify prior distributions to make the EM steps tractable. Nonetheless, extending those methods to graphical modeling is a challenging and interesting path to investigate.

Finally, the increasing amount of biological knowledge is essential to perform accurate and efficient studies and to identify biomarkers. It is therefore important to integrate into the different models as much accurate prior biological information as possible, as considered in [60]. Additionally, the improvement of computational technology can only help us to conduct such high-dimensional analyses.

APPENDIX A

Appendix

A.1 Kronecker product and vec operator

This section discusses some topics in matrix algebra, namely the Kronecker product of two matrices and the vec operator.

Definition A.1. *Let $\boldsymbol{A}$ an $n \times p$ matrix and $\boldsymbol{B}$ an $m \times q$ matrix. Then the Kronecker product of $\boldsymbol{A}$ and $\boldsymbol{B}$, denoted by $\boldsymbol{A} \otimes \boldsymbol{B}$, is the $mn \times pq$ matrix*

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} a_{11}\boldsymbol{B} & a_{12}\boldsymbol{B} & \dots & a_{1p}\boldsymbol{B} \\ a_{21}\boldsymbol{B} & a_{22}\boldsymbol{B} & \dots & a_{2p}\boldsymbol{B} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1}\boldsymbol{B} & a_{n2}\boldsymbol{B} & \dots & a_{np}\boldsymbol{B} \end{pmatrix}$$

Properties:

- Associativity: $\boldsymbol{A} \otimes (\boldsymbol{B} \otimes \boldsymbol{C}) = (\boldsymbol{A} \otimes \boldsymbol{B}) \otimes \boldsymbol{C}$,
- Distributivity: $\boldsymbol{A} \otimes (\boldsymbol{B} + \boldsymbol{C}) = (\boldsymbol{A} \otimes \boldsymbol{B}) + (\boldsymbol{A} \otimes \boldsymbol{C})$, $\quad (\boldsymbol{A} + \boldsymbol{B}) \otimes \boldsymbol{C} = (\boldsymbol{A} \otimes \boldsymbol{C}) + (\boldsymbol{B} \otimes \boldsymbol{C})$,
- For some scalars a and b: $a\boldsymbol{A} \otimes b\boldsymbol{B} = ab\boldsymbol{A} \otimes \boldsymbol{B}$,
- For some matrices with right dimensions: $(\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{C} \otimes \boldsymbol{D}) = \boldsymbol{AC} \otimes \boldsymbol{BD}$,
- Transposition: $(\boldsymbol{A} \otimes \boldsymbol{B})^T = \boldsymbol{A}^T \otimes \boldsymbol{B}^T$,
- Trace: $\mathrm{tr}(\boldsymbol{A} \otimes \boldsymbol{B}) = \mathrm{tr}(\boldsymbol{A})\mathrm{tr}(\boldsymbol{B})$,
- Rank: $\mathrm{rank}(\boldsymbol{A} \otimes \boldsymbol{B}) = \mathrm{rank}(\boldsymbol{A})\mathrm{rank}(\boldsymbol{B})$,
- Determinant: $\det(\boldsymbol{A} \otimes \boldsymbol{B}) = \det(\boldsymbol{A})^n \det(\boldsymbol{B})^m$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are respectively $m \times m$ and $n \times n$ matrices,
- Inverse: $(\boldsymbol{A} \otimes \boldsymbol{B})^{-1} = \boldsymbol{A}^{-1} \otimes \boldsymbol{B}^{-1}$.

Definition A.2. *The vec operator is an operator which creates a column vector from a matrix $\boldsymbol{A}$ by stacking the column vectors of $\boldsymbol{A} = [\boldsymbol{a}_1\boldsymbol{a}_2...\boldsymbol{a}_n]$ below one another :*

$$vec(\boldsymbol{A}) = \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{a}_n \end{pmatrix}$$

Example : If $A = \begin{pmatrix} 2 & -7 \\ 14 & 6 \end{pmatrix}$, then $\text{vec}(A) = \begin{pmatrix} 2 \\ 14 \\ -7 \\ 6 \end{pmatrix}$.

Theorem A.1. *Let $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{X}$ be 3 matrices of conforming sizes. Then*

$$vec(\boldsymbol{AXB}) = (\boldsymbol{B}^T \otimes \boldsymbol{A})vec(\boldsymbol{X})$$

Proof. Let $\boldsymbol{B} = [\boldsymbol{b}_1 ... \boldsymbol{b}_n], \boldsymbol{X} = [\boldsymbol{x}_1 ... \boldsymbol{x}_m]$. The k-th column of $\boldsymbol{AXB}$ is

$$\begin{aligned} (\boldsymbol{AXB})_{..k} &= \boldsymbol{AXb}_k = \boldsymbol{A} \textstyle\sum_{i=1}^m \boldsymbol{x}_i b_{ik} \\ &= [b_{1k}\boldsymbol{A} ... b_{mk}\boldsymbol{A}] \begin{pmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_m \end{pmatrix} \\ &= ([b_{1k} ... b_{mk}] \otimes \boldsymbol{A})\text{vec}(\boldsymbol{X}) = \boldsymbol{b}_k^T \otimes \boldsymbol{A})\text{vec}(\boldsymbol{X}) \end{aligned}$$

Then, stacking the colums below one another

$$\begin{aligned} \text{vec}(\boldsymbol{AXB}) &= \begin{pmatrix} \boldsymbol{AXB}_{..1} \\ \vdots \\ \boldsymbol{AXB}_{..n} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b}_1^T \otimes \boldsymbol{A} \\ \vdots \\ \boldsymbol{b}_n^T \otimes \boldsymbol{A} \end{pmatrix} \text{vec}(\boldsymbol{X}) \\ &= (\boldsymbol{B}^T \otimes \boldsymbol{A})\text{vec}(\boldsymbol{X}) \end{aligned}$$

□

Corollary A.1.

$$\begin{aligned} vec(\boldsymbol{AB}) &= (\boldsymbol{B}^T \otimes \boldsymbol{A})vec(\boldsymbol{I}) \\ &= (\boldsymbol{B}^T \otimes \boldsymbol{I})vec(\boldsymbol{A}) \\ &= (\boldsymbol{I} \otimes \boldsymbol{A})vec(\boldsymbol{B}) \end{aligned}$$

Property:

$$\mathrm{tr}(\boldsymbol{AB}) = \mathrm{vec}(\boldsymbol{A}^T)^T \mathrm{vec}(\boldsymbol{B})$$

The proof is immediate writing the formula of the trace, using the expression of the matrices coefficients.

A.2 Distribution theory

This section discusses some common distributions and their propoerties.

Normal distribution

A random variable X follows a normal distribution with mean μ and standard deviation σ, denoted by $N(\mu, \sigma^2)$, if and only if its density $f(x)$ is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

The normal distribution with mean $\mu = 0$ and standard deviation $\sigma = 1$ is called the standard normal distribution.

Truncated normal distribution

The truncated normal distribution is the probability distribution of a normally distributed random variable, whose values are restricted to lie between two values a and b in the case of a two-tailed truncation, or higher than a or lower than b in the case of an one-tailed truncation. If X follows a truncated normal distribution $N(\mu, \sigma^2)$ between a and b, its density function is:

$$\frac{\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})}{\Phi(\frac{b-\mu}{\sigma}) - \Phi(\frac{a-\mu}{\sigma})},$$

for $x \in [a, b]$, where ϕ and Φ respectively denote the probability density function and cumulative distribution function of the standard normal distribution. In the case of an one-tailed truncation $x \geq a$, we can write $b = \infty$ and $\Phi(\frac{b-\mu}{\sigma}) = 1$, then the density function becomes:

$$\frac{\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})}{1 - \Phi(\frac{a-\mu}{\sigma})} = \frac{\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})}{1 - \Phi(\alpha)},$$

with $\alpha = \frac{x-\mu}{\sigma}$. The expected value and value are then:

$$\begin{aligned} \mathbb{E}(X|x \geq a) &= \mu + \sigma\lambda(\alpha) \\ \mathbb{V}(X| \geq a) &= \sigma^2[1 - \delta(\alpha)], \end{aligned}$$

with

$$\begin{aligned} \lambda(\alpha) &= \phi(\alpha)/[1 - \Phi(\alpha)] \\ \delta(\alpha) &= \lambda(\alpha)[\lambda(\alpha) - \alpha]. \end{aligned}$$

These results and other details can be found in Greene [18].

Exponential distribution

A random variable $X > 0$ follows an exponential distribution with rate parameter $\lambda > 0$, denoted by $X \sim Exp(\lambda)$, if and only if its density $f(x)$

is:

$$f(x) = \lambda \exp(-\lambda x).$$

Gamma distribution

A random variable $\phi > 0$ has a gamma distribution with shape parameter $a > 0$ and scale parameter $b > 0$, denoted by $\phi \sim Ga(a, b)$ if and only if its density $f(\phi)$ is:

$$f(\phi) = \frac{1}{\Gamma(a) b^a} \phi^{a-1} \exp(-\phi/b),$$

where $\Gamma(.)$ is the gamma function.

Properties:

- Expected value: $\mathbb{E}(\phi) = ab$.
- Variance: $\mathbb{V}(\phi) = ab^2$.
- If $a = 1$, then ϕ has an exponential distribution with parameter $1/b$.

Inverse-gamma distribution

A random variable $\psi > 0$ has an inverse-gamma distribution with parameters $\alpha > 0, \beta > 0$, denoted by $\psi \sim IGa(\alpha, \beta)$, if and only if its density $f(\psi)$ is:

$$f(\psi) = \frac{\beta^\alpha}{\Gamma(\alpha)} (1/\psi)^{\alpha+1} \exp(-\beta/\psi).$$

Properties:

- Expected value: $\mathbb{E}(\psi) = \frac{\beta}{\alpha-1}$.
- Variance: $\mathbb{V}(\psi) = \frac{\beta^2}{(\alpha-1)(\alpha-2)}$.

- If $\psi \sim IG(\alpha, \beta)$, then $\phi = 1/\Psi \sim G(\alpha, 1/\beta)$.

Multivariate normal distribution

A random vector $\boldsymbol{X}$ of size p is said to have a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, written as $\boldsymbol{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ or $\boldsymbol{X} \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, when its density function is:

$$(2\pi)^{-\frac{p}{2}}|\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{X}-\boldsymbol{\mu})\Big),$$

where $|\boldsymbol{\Sigma}|$ denotes the determinant of $\boldsymbol{\Sigma}$.

Matrix-variate normal distribution

A $n \times p$ random matrix is said to have a matrix variate normal distribution with mean matrix $\boldsymbol{M}$, $n \times n$ among-row covariance matrix $\boldsymbol{U}$, $p \times p$ among-column covariance matrix $\boldsymbol{V}$, written as $\boldsymbol{X} \sim N(\boldsymbol{M}, \boldsymbol{U}, \boldsymbol{V})$, or $\boldsymbol{X} \sim N_{n,p}(\boldsymbol{M}, \boldsymbol{U}, \boldsymbol{V})$, if its density is:

$$\frac{\exp\Big(-\frac{1}{2}\mathrm{tr}\big[\boldsymbol{V}^{-1}(\boldsymbol{X}-\boldsymbol{M})^T\boldsymbol{U}^{-1}(\boldsymbol{X}-\boldsymbol{M})\big]\Big)}{(2\pi)^{\frac{np}{2}}|\boldsymbol{V}|^{\frac{n}{2}}|\boldsymbol{U}|^{\frac{p}{2}}}.$$

The matrix variate normal distribution is related to the the multivariate normal distribution by the following equivalence:

$$\boldsymbol{X} \sim N_{n,p}(\boldsymbol{M}, \boldsymbol{U}, \boldsymbol{V}) \quad \Leftrightarrow \quad vec(\boldsymbol{X}) \sim N_{np}(\boldsymbol{M}, \boldsymbol{V} \otimes \boldsymbol{U}).$$

This equivalence can be proved by using properties of the trace, *textrmvec* operator and kronecker product; details can be found in [20].

Property: If $\boldsymbol{X} \sim N_{n,p}(\boldsymbol{M}, \boldsymbol{U}, \boldsymbol{V})$, then, assuming matrices $\boldsymbol{D}$ and $\boldsymbol{C}$ of appropriate dimensions and of full rank:

$$\boldsymbol{DXC} \sim N_{n,p}(\boldsymbol{DMC}, \boldsymbol{DUD}^T, \boldsymbol{C}^T\boldsymbol{VC}).$$

A proof of that property is available in [20].

Wishart distribution

A $p \times p$ random symmetric positive definite matrix $\boldsymbol{V}$ is said to have a Wishart distribution with parameters ν degrees od freedom, and scale matrix $\boldsymbol{S}$, written as $\boldsymbol{V} \sim W_p(\nu, \boldsymbol{S})$, if its density is:

$$\frac{1}{2^{\frac{\nu p}{2}}|\boldsymbol{S}|^{\frac{\nu}{2}}\Gamma_p(\frac{\nu}{2})}|\boldsymbol{V}|^{\frac{\nu-p-1}{2}} \exp\Big(- \frac{\operatorname{tr}(\boldsymbol{S}^{-1}\boldsymbol{V})}{2}\Big),$$

where the scale matrix $\boldsymbol{S}$ is a $p \times p$ positive definite matrix and Γ_p is the multivariate gamma function.

Properties:

- Expected value: $\mathbb{E}(\boldsymbol{V}) = \nu\boldsymbol{S}$.
- Mode: $\text{Mode}(\boldsymbol{V}) = (\nu - p - 1)\boldsymbol{S}$.
- Variance: $\mathbb{V}(V_{ij}) = \nu(s_{ij}^2 + s_{ii}s_{jj})$.
- In Bayesian statistics, the Wishart distribution is the conjugate prior to the precision matrix $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$, where $\boldsymbol{\Sigma}$ is the covariance matrix.

Inverse-Wishart distribution

A $p \times p$ symmetric positive definite matrix $\boldsymbol{X}$ is said to have an inverse-Wishart distribution, with ν degrees of freedom and positive definite scale matrix $\boldsymbol{\Psi}$, written as $\boldsymbol{X} \sim IW_p(\nu, \boldsymbol{\Psi})$ if its density is:

$$\frac{|\boldsymbol{\Psi}|^{\frac{\nu}{2}}}{2^{\frac{\nu p}{2}}\Gamma_p(\frac{\nu}{2})}|\boldsymbol{X}|^{-\frac{\nu+p+1}{2}} \exp\Big(-\frac{\mathrm{tr}(\boldsymbol{\Psi}\boldsymbol{X}^{-1})}{2}\Big).$$

Properties:

- Expected value: $\mathbb{E}(\boldsymbol{X}) = \frac{\boldsymbol{\Psi}}{\nu-p-1}$.
- Mode: $\mathrm{Mode}(\boldsymbol{X}) = \frac{\boldsymbol{\Psi}}{\nu+p+1}$.
- Variance: $\mathbb{V}(X_{ij}) = \frac{(\nu-p+1)\psi_{ij}+(\nu-p-1)\psi_{ii}\psi_{jj}}{(\nu-p)(\nu-p-1)^2(\nu-p-3)}$.
- If $\boldsymbol{X} \sim IW_p(\nu, \boldsymbol{\Psi})$, then $\boldsymbol{X}^{-1} \sim W_p(\nu, \boldsymbol{\Psi}^{-1})$

MCMC procedure: Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm is a MCMC method (Markov Chain Monte Carlo) which is generally used to approximate a probability distribution which it is difficult to sample from directly. It is generally computed for multi-dimensions distributional distributions, especially when the dimensions are high, the reason why it is difficult to directly sample from.

Let $X_1, X_2, \ldots, X_n$ be a Markov chain, generated from the stationary distribution $f_X(x)$, also called the target density. If we wish to estimate the following points $X_{n+1}, X_{n+2,\ldots}$, we may compute the Metropolis-Hastings procedure as follows:

- let X_c be the current value of our Markov chain, and X_{new} the next point we wish to estimate;
- a candidate point X_p is sampled from a proposal distribution $q(X_p|X_c)$; This proposal distribution is defined and adapted according to the situation of interest. We can choose a normal distribution with mean X_c, a gamma distribution if the random variable needs to be positive...
- we then compute the probability of acceptance $\alpha(X_c, X_p)$:

$$\alpha(X_c, X_p) = \min\Big(\frac{f_X(X_p)q(X_c|X_p)}{f_X(X_c)q(X_p|X_c)}, 1\Big); \tag{A.1}$$

  If the proposal density is symmetric (normal distribution for example), this probability is:

$$\alpha(X_c, X_p) = \min\Big(\frac{f_X(X_p)}{f_X(X_c)}, 1\Big);$$

- we sample U from the $U(0,1)$ distribution;
- if $U \geq \alpha(X_c, X_p)$, we accept the proposed value and we set $X_{new} = X_p$. On the contrary, if $U \leq \alpha(X_c, X_p)$, we do not accept the proposed value and do not update the chain, we set $X_{new} = X_c$.
- and so on for the following steps, until we reach convergence to the target distribution.

One difficulty lies in the choice of the proposal distribution $q(X_p|X_c)$. For better performance of the algorithm and convergence of the chain, it is preferable to choice a density which matches the shape of the target density $f_X(x)$. For example, we choose a normal distribution if the variable is believed to follow a distribution close to the normal. If we want the variable to lie in a certain range of values, it may be suitable to use a truncated normal. Also, if we assume the variable to be positive, we may want to use a gamma or exponential distribution to satisfy this constraint. A good convergence also requires a suitable acceptance rate, the proportion proposed samples which are actually accepted. This acceptance rate depends on the variance of the proposal density. This is why we need to tune the parameters of $q(X_p|X_c)$ during the burn-in period such that we obtain a suitable acceptance rate, ideally around 25% for a multidimensional normal distribution [50]. If the variance $q(X_p|X_c)$ is too high, the proposed samples will lie in regions with very low likelihoods, then the acceptance rate will be too small and the chain will converge slowly. On the other hand, if the variance is too small, the proposed samples will have high likelihoods, leading to a high acceptance rate, but will remain very close to the current value, also resulting in very slow convergence to the true distribution $f_X(x)$.

In practice, the main difficulty in implementing MCMC is to identify whether or not the Markov chain has converged or not to its stationary distribution $f_X(x)$. A practical way to check convergence is to have a look at plots of the chain. If we observe that the chain converges and stabilizes towards a given value, if the likelihood of the model also stays in the same region, we may conclude the chain has reached convergence. A convergence diagnostic, often used as a formal way of checking for convergence, is the Brooks, Gelman and Rubin (BGR) diagnostic [15, 6]. More detailed discussion about MCMC can be found in either [14] or [49].

A.3 Graphs from the different chains run in Chapter 5

The Figures in this section give the trace plots of the log-relative probabilities of the visited models, under each condition, in the order STEMI, NSTEMI, Unstable Angina. Each graph corresponds to a chain run to build the different networks presented in Section 5.2. Figures A.1 to A.3 show the log-relative probabilities for the whole chains, when Figures A.4 to A.6 only show those probabilities from the end of the burn-in onwards. For each Figure, the different graphs represent the different clusters, from cluster 1 (top left) to cluster 6 (bottom right).

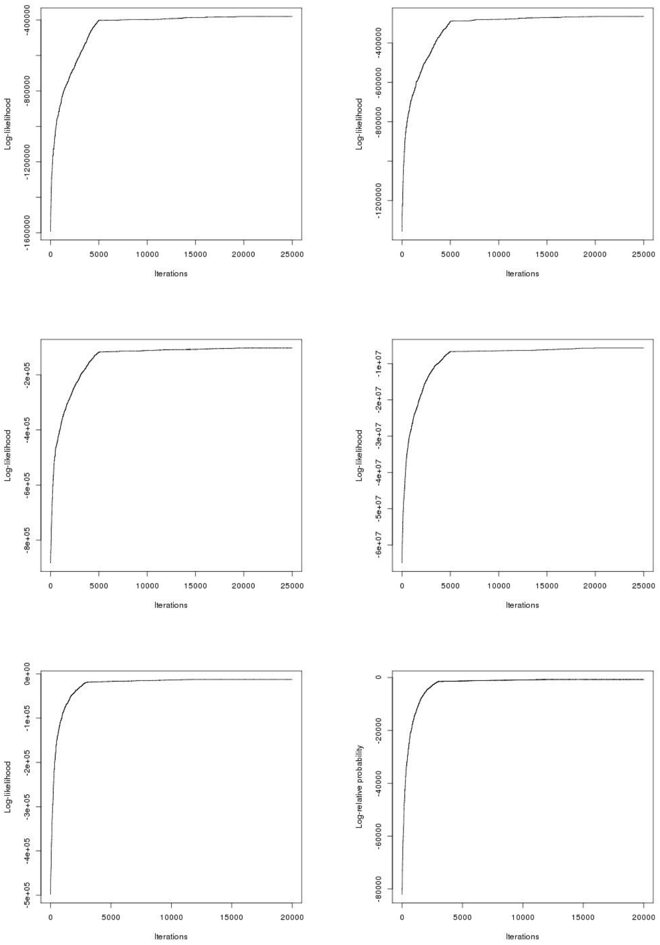

Figure A.1: Log-relative probabilities of the visited models, STEMI condition

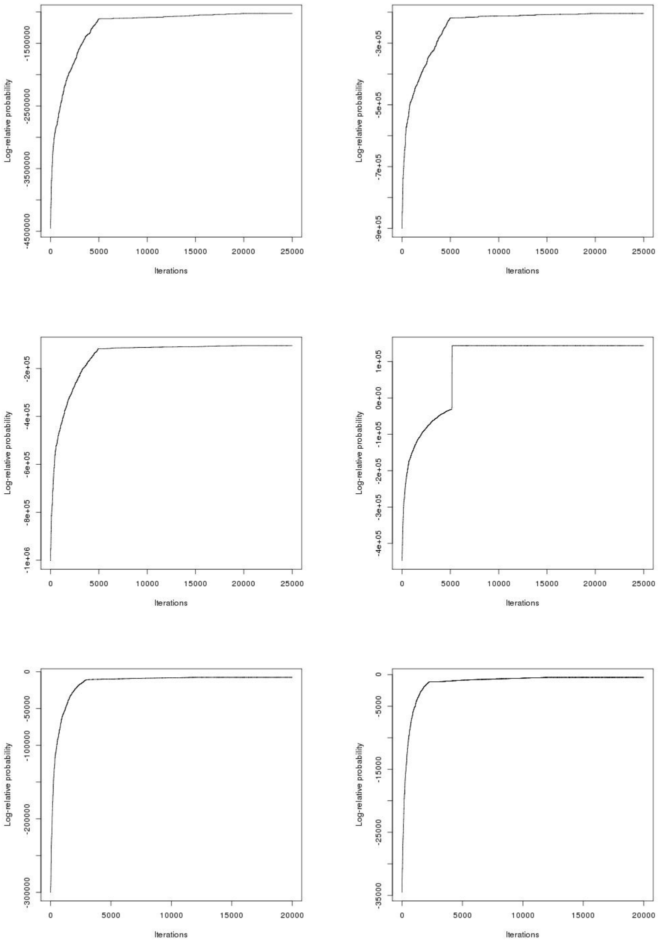

Figure A.2: Log-relative probabilities of the visited models, NSTEMI condition

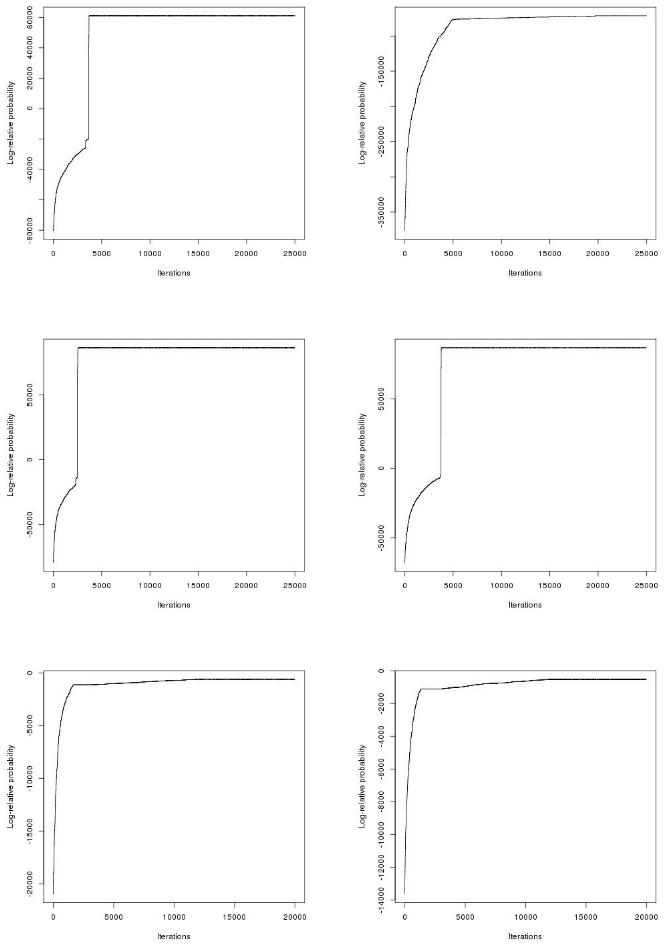

Figure A.3: Log-relative probabilities of the visited models, Unstable Angina condition

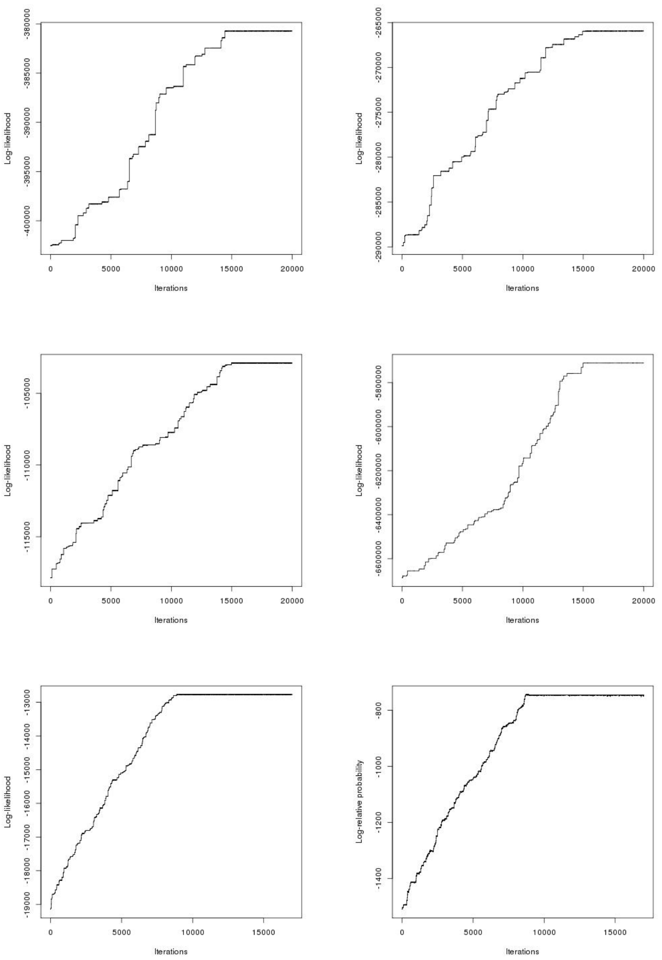

Figure A.4: Log-relative probabilities of the visited models after burn-in, STEMI condition

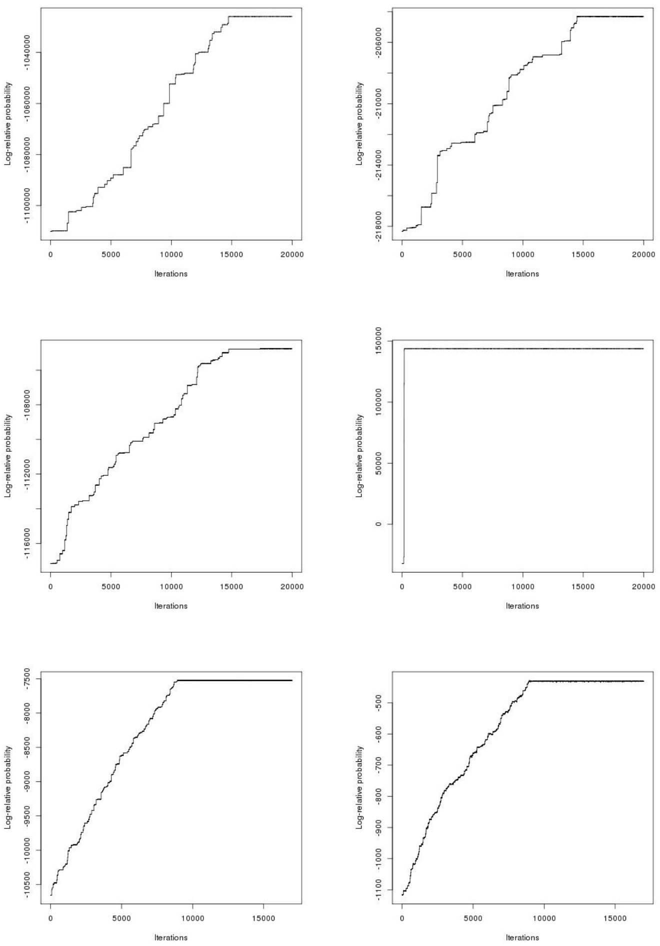

Figure A.5: Log-relative probabilities of the visited models after burn-in, NSTEMI condition

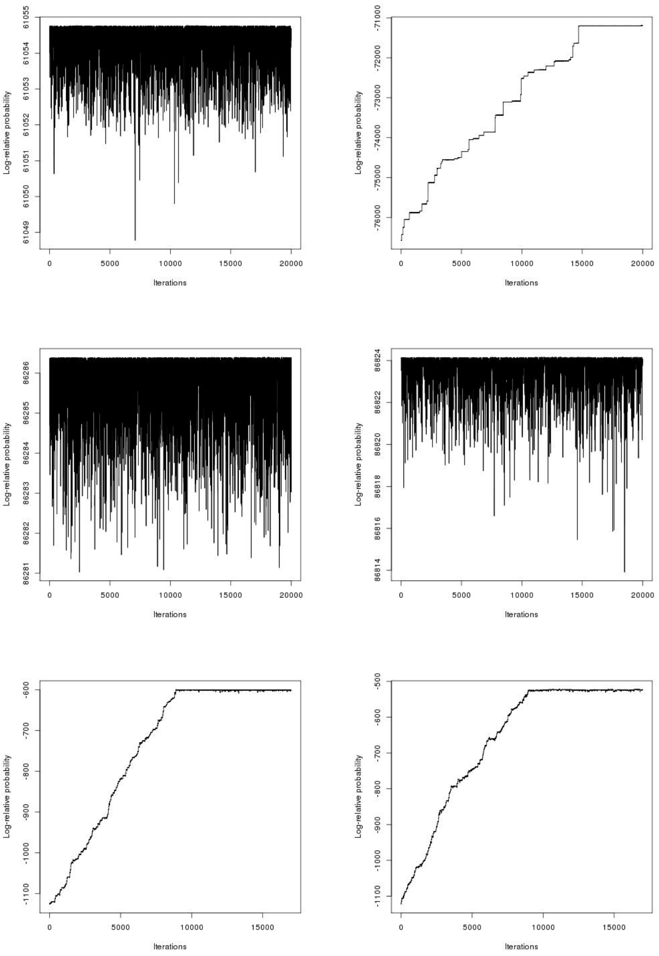

Figure A.6: Log-relative probabilities of the visited models after burn-in, Unstable Angina condition

Bibliography

[1] Alberts B, Johnson A, Lewis J, Raff M, Roberts K, and Walter P. *Molecular Biology of the Cell.* Garland Science, New York, 4th edition, 2002.

[2] Alberts B, Johnson A, Lewis J, Raff M, Roberts K, and Walter P. *Molecular Biology of the Cell.* Garland Science, New York, 5th edition, 2010.

[3] Barabási AL and Oltvai ZN. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[4] Bartel DP. MicroRNAs: Target recognition and regulatory functions. *Cell*, 136(2):215:33, 2009.

[5] Bollobás B. *Random graphs, Second edition.* Cambridge University Press, 2001.

[6] Brooks SP and Gelman A. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, 1998.

[7] Chu Y and Corey DR. Rna sequencing: Platform selection, experimental design, and data interpretation. *Nucleic Acid Therapeutics*, 22(4):271–274, 2012.

[8] Dawid AP and Lauritzen SL. Hyper Markov laws in the statistical analysis of decomposable graphical models. *Annals of Statistics*, 21(3):1272–1317, 1993.

[9] Enright AJ, John Band Gaul U, Tuschl T, Sander C, and Marks DS. MicroRNA targets in Drosophila. *Genome Biology*, 5(1):R1, 2003.

[10] Erdos P and Renyi A. On random graphs. *Publicaciones Mathematicae*, 6:290–297, 1959.

[11] Farh KKH, Grimson A, Jan C, Lewis BP, Johnston WK, Lim LP, Burge CB, and Bartel DP. The widespread impact of mammalian microRNAs on mrna repression and evolutions. *Science*, 310(5755):1817–1821, 2005.

[12] Frey BJ and Dueck D. Clustering by passing messages between data points. *Science*, 315(5814):972–16, 2007.

[13] Friedman RC, Farh KK, Burge CB, and Bartel DP. Most mammalian mRNAs are conserved targets of microRNAs. *Genome Research*, 19(1):92–105, 2008.

[14] Gamerman D and Lopes HF. *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, Second Edition.* Chapman and Hall/CRC Press Texts in Statistical Science, 2006.

[15] Gelman A and Rubin B. Inferences from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472, 1992.

[16] Giancarlo R, Scaturro D, and Utro F. Computational cluster validation for microarray data analysis: experimental assessment of clest, consensus clustering, figure of merit, gap statistics and model explorer. *BMC Bioinformatics*, 9(462), 2008.

[17] Grada A and Weinbrechy K. Next-generation sequencing: Methodology and application. *Journal of Investigative Dermatology*, 133(8):e11, 2013.

[18] Green WH. *Econometric Analysis.* Prentice Hall, 5th edition, 2002.

[19] Grimson A, Farh KKH, Johnston WK, Garrett-Engele P, Lim LP, and Bartel DP. MicroRNA targeting specificity in mammals: Determinants beyond seed pairing. *Molecular Cell*, 27(1):91–105, 2007.

[20] Gupta AK and Nagar DK. *Matrix Variate Distributions.* CRC Press, 1999.

[21] Handl J, Knowles J, and Kell DB. Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–12, 2005.

[22] Hofacker IL, Fontana W, Stadler PF, Bonhoeffer SL, Tacker M, and Schuster P. Fast folding and comparison of RNA secondary structures. *Chemical Monthly*, 125(2):167–188, 1994.

[23] Huang JC, Frey BJ, and Morris QD. Comparing sequence and expression for predicting microRNA targets using genmir3. *Pacific Symposium for Biocomputing (PSB)*, 13:52–63, 2008.

[24] Huang JC, Morris QD, and Frey BJ. Detecting microRNA targets by linking sequence, microRNA and gene expression data. *Lecture Notes in Computer Science*, 3909:114–129, 2006.

[25] Huang JC, Morris QD, and Frey BJ. Bayesian inference of microRNA targets from sequence and expression data. *Journal of Computational Biology*, 14(5):550–563, 2007.

[26] Jensen F. *Introduction to Bayesian Networks*. Springer-Verlag, New York, 1996.

[27] John B, Enright AJ, Aravin A, Tusch T, Sander C, and Marks DS. Human microRNA targets. *Plos Biology*, 2(11):e363, 2004.

[28] Johnson WE, Welker NC, and Bass BL. Dynamic Linear Model for the identification of miRNAs in next-generation sequencing data. *Biometrics*, 67(4):1206–1214, 2011.

[29] Kertesz M, Iovino N, Unnerstall U, Gaul U, and Segal E. The role of site accessibility in microRNA target recognition. *Nature Genetics*, 39:1278–1284, 2007.

[30] Kiriakidou M, Nelson P T, Kouranov A, Fitziev P, Bouyioukos C, Mourelatos Z, and Hatzigeorgiou A. A combined computational-experimental approach predicts human microRNA targets. *Genes and Development*, 18(10):1165–1178, 2004.

[31] Kohonen T. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–80, 1990.

[32] Krek A, Grun D, Poy MN, Wolf R, Rosenberg L, Epstein EJ, Macmenamin P, da Piedade I, Gunsalus KC, Stoffel M, and RajewskyN. Combinatorial microRNA target predictions. *Nature Genetics*, 37(5):495–500, 2005.

[33] Lagos-Quintana M, Rauhut R, Lendeckel W, and Tuschl T. Identification of novel genes coding for small expressed RNAs. *Science*, 294(5543):853–858, 2001.

[34] Lau NC, Lim LP, Weinstein EG, and Bartel DP. An abundant class of tiny RNAs with probable regulatory roles in caenorhabditis elegans. *Science*, 294(5543):858–862, 2001.

[35] Lauritzen. *Graphical Models.* Oxford University Press, Oxford, 1996.

[36] Lee RC, Feinbaum RL, and Ambros V. The *C.elegans* heterochronic gene *lin-4* encodes small RNAs with antisense complementarity to *lin-14*. *Cell*, 75(5):843–854, 1993.

[37] Lee RC and Ambros V. An extensive class of small RNAs in caenorhabditis elegans. *Science*, 294(5543):862–864, 2001.

[38] Leitner A. MicroRNA target prediction. Master's thesis, Graz University of Technology.

[39] Lewis BP, Burge CB, and Bartel DP. Conserved seed-pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. *Cell*, 125(1):15–20, 2005.

[40] Lewis BP, Shih IH, Jones-Rhoades MW, Bartel DP, and Burge CB. Prediction of mammalian microRNA targets. *Cell*, 115(7):787–798, 2003.

[41] Mardis ER. Next-generation DNA sequencing methods. *Annual Review of Genomics and Human Genetics*, 9:387–402, 2008.

[42] Newman AM and Cooper JB. Autosome: a clustering method for identifying gene expression modules without prior knowledge of cluster number. *BMC Bioinformatics*, 11:117, 2010.

[43] Pearson R, Liu X, Sanguinetti G, Milo M, Lawrence N, and al. puma: a bioconductor package for propagating uncertainty in microarray analysis. *BioMed Central*, 10(211), 2009.

[44] Porekar J. Random networks, 2002. Unpublished paper available at www-f1.ijs.si/r̃udi/sola/Random_Networks.pdf.

[45] Rajewski N. MicroRNA target predictions in mammals. *Nature Genetics*, 38(Supplement):S8–13, 2006.

[46] Rattray M, Liu X, Sanguinetti G, Milo M, and Lawrence N. Propagating uncertainty in microarray analysis. *Brief Bioinformatics*, 7(1):37–47, 2006.

[47] Reinhart BJ, Slack FJ, Basson M, Pasquinelli AE, Bettinger JC, Rougvie AE, Horvitz HR, and Ruvkun G. The 21-nucleotide let-7 RNA regulates developmental timing in Caenorhabditis elegans. *Nature*, 403(6772):901–906, 2000.

[48] Rhoades MW, Reinhart BJ, Lim LP, Burge CB, Bartel B, and Bartel DP. Prediction of plant microRNA targets. *Cell*, 110(4):513–520, 2002.

[49] Robert C. *The Bayesian Choice: from Decision-Theoretic Motivations to Computational Implementation.* Springer-Verlag, New York, 2001.

[50] Roberts GO, Gelman A, and Gilks WR. Weak convergence and optimal scaling of random walk metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120, 1997.

[51] Rothman AMK, Flaherty LM, Morton AC, Alecu A, Coca D, Crossman DC, and Chico TJA ans Milo M. Whole blood microrna profiles of individual patients characterise myocardial infarction and unstable angina. preprint.

[52] Ročková V. *Bayesian Variable Selection in High-Dimensional Applications.* PhD thesis, Erasmus Universiteit Rotterdam, 2013.

[53] Ročková V and George E. Emvs: The EM approach to bayesian variable selection. 109(506):828–846, 2014.

[54] Ruan D. *Statistical Methods for Comparing Labelled Graphs.* PhD thesis, Department of Mathematics, Imperial College London, 2014.

[55] Ruan D, Young A, and Montana G. Differential amalysis of biological networks. Submitted for publication, 2014.

[56] Selbach M, Schwanhäusser B, Thierfelder N, Fang Z, Khanin R, and Rajewski N. Widespread changes in protein synthesis induced by microRNAs. *Nature*, 455(7209):58–63, 2008.

[57] Shannon P, Markiel A, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, and Ideker T. Cytoscape: a software environment for in-

tegrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.

[58] Stingo FC, Chen YA, Vanucci M, Barrier M, and Mirkes PE. A Bayesian graphical modeling approach to microRNA regulatory network inference. *Annals of Applied Statistics*, 4(4):2024–2048, 2010.

[59] Stingo FC and Vanucci M. Variable selection for discriminant analysis with Markov random field priors for the analysis of microarray data. *Bioinformatics*, 27(4):495–501, 2011.

[60] Stingo FC, Chen YA, Tadesse MG, and Vanucci M. Incorporating biological information into linear models: A Bayesian approach to the selection of pathways and genes. *Annals of Applied Statistics*, 5(3):1978–2002, 2011.

[61] Wang J, Delabie J, Aasheim HC, Smeland F, and Myklebost O.

[62] Yoon S and Micheli GD. Computational identification of microRNA and their targets. *Birth Defects Research (Part C) 78:118-128*, 78:118–128, 2006.

[63] Zweig MH and Campbell G. Receiver-Operating Characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry*, 39(4):561–577, 1993.