



UNIVERSIDAD NACIONAL DE COLOMBIA

An Assessment of Gene Regulatory Network Inference Algorithms

Adrián Guillermo Zuur Pedraza

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Estadística
Bogotá, Colombia
2020

An Assessment of Gene Regulatory Network Inference Algorithms

Adrián Guillermo Zuur Pedraza

Final project submitted for the degree of:
Master of Science – Statistics

Advisor:
Liliana López-Kleine, Ph.D.

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Estadística
Bogotá, Colombia
2020

Acknowledgements

Aunque el resto de este documento esté en inglés, me parece que tiene más sentido escribir esta pequeña parte en español.

En primer lugar, le agradezco a Liliana López Kleine por su guianza a lo largo de la elaboración de este trabajo. Sus sugerencias y observaciones me ayudaron a encauzar las variadas y a veces confusas ideas que tuve. Me alegra haber coincidido con ella en el interés por este tema retador.

Este trabajo supuso un gran esfuerzo emocional para mí, además del intelectual. En ese sentido, doy gracias a mis padres, Ted y Constanza, por apoyarme sostenidamente y de múltiples formas a lo largo de toda mi vida, y por haber cultivado mi cariño por el aprendizaje durante mi crianza. También, le agradezco a Paola Parada por sus palabras de ánimo y muestras de afecto, sin las cuales me habría costado mucho más realizar esta labor. Además de a ellos, les agradezco a todos mis amigos y familiares quienes de una forma u otra me acompañaron en este proceso.

Abstract

A conceptual issue regarding gene regulatory network (GRN) inference algorithms is establishing their validity or correctness. In this study, we argue that for this purpose it is useful to conceive these algorithms as estimators of graph-valued parameters of explicit models for gene expression data. On this basis, we perform an assessment of a selection of influential GRN inference algorithms as estimators for two types of models: (i) causal graphs with associated structural equations models (SEMs), and (ii) differential equations models based on the thermodynamics of gene expression. Our findings corroborate that networks of marginal dependence fail in estimating GRNs, but they also suggest that the strength of statistical association as measured by mutual information may be indicative of GRN structure. Also, in simulations, we find that the GRN inference algorithms GENIE3 and TIGRESS outperform competing algorithms. However, more importantly, we also find that many observed patterns hinge on the GRN topology and the assumed data generating mechanism.

Keywords: gene regulatory network, gene network inference, gene regulation, biological network, relevance network, structural equations model, thermodynamic model.

Resumen

Un problema conceptual con respecto a los algoritmos de inferencia de redes de regulación génica (RRG) es cómo establecer su validez. En este estudio sostenemos que para este objetivo conviene concebir estos algoritmos como estimadores de parámetros de modelos estadísticos explícitos para datos de expresión génica. Sobre esta base, realizamos una evaluación de una selección de algoritmos de inferencia de RRG como estimadores para dos tipos de modelos: (i) modelos de grafos causales asociados a modelos de ecuaciones estructurales (MEE), y (ii) modelos de ecuaciones diferenciales basados en la termodinámica de la expresión génica. Nuestros hallazgos corroboran que las redes de dependencias marginales fallan en la estimación de las RRG, pero también sugieren que la fuerza de la asociación estadística medida por la información mutua puede reflejar en cierto grado la estructura de las RRG. Además, en un estudio de simulaciones, encontramos que los algoritmos de inferencia GENIE3 y TIGRESS son los de mejor desempeño. Sin embargo, crucialmente, también encontramos que muchos patrones observados en las simulaciones dependen de la topología de la RRG y del modelo generador de datos.

Palabras clave: red de regulación génica, inferencia de redes génicas, regulación génica, red biológica, red de relevancia, modelo de ecuaciones estructurales, modelo termodinámico

List of Figures

2-1. Fundamental concepts of genetics	5
2-2. The Central Dogma of Molecular Biology.	6
2-3. Gene regulatory networks as abstract models	8
2-4. Microarray of mouse gene cDNA	11
5-1. Syntax of do-operator in a causal graphical model	45
5-2. Examples of 3-node causal graphs	47
6-1. Undirected 4-chain and 4-cycle motifs	60
6-2. GRNs for simulation study	64
6-3. Induced statistical dependences in non-linear SEM	67
6-4. Sample time series from thermodynamic ODE model of gene expression	69
7-1. Number of unlabelled connected undirected graphs.	72
7-2. Number of labelled acyclic orientations of connected homogeneous graphs.	74
7-3. Number of unlabelled acyclic orientations of connected homogeneous graphs	75
7-4. False positive rate of Relevance Networks on randomly chosen DAGs	77
7-5. Algorithm AUROC on linear Gaussian SEM on GSD network.	86
7-6. Algorithm AUROC on Gaussian Copula SEM with Laplacian marginals on GSD network.	87
7-7. Algorithm AUROC on Gaussian Copula SEM with Pareto marginals on GSD network.	88
7-8. Algorithm AUROC on linear Gaussian SEM on HSC network.	89
7-9. Algorithm AUROC on Gaussian Copula SEM with Laplacian marginals on HSC network.	90
7-10. Algorithm AUROC on Gaussian Copula SEM with Pareto marginals on HSC network.	91
7-11. Algorithm accuracy (default discretization) on linear Gaussian SEM on mCAD network.	92
7-12. Algorithm accuracy (default discretization) on Gaussian Copula SEM with Laplacian marginals on mCAD network.	93
7-13. Algorithm accuracy (default discretization) on Gaussian Copula SEM with Pareto marginals on mCAD network.	94

7-14. Algorithm accuracy (default discretization) on linear Gaussian SEM on HSC network.	95
7-15. Algorithm accuracy (default discretization) on Gaussian Copula SEM with Laplacian marginals on HSC network.	96
7-16. Algorithm accuracy (default discretization) on Gaussian Copula SEM with Pareto marginals on HSC network.	97
7-17. Algorithm AUROC on cosine transformation causal graphical model on GSD network.	98
7-18. Algorithm AUROC on cosine transformation causal graphical model on HSC network.	99
7-19. Algorithm accuracy (default discretization) on cosine transformation causal graphical model on mCAD network.	100
7-20. Algorithm accuracy (default discretization) on cosine transformation causal graphical model on HSC network.	101
7-21. Algorithm AUROC on Thermodynamic ODE Model on VSC network.	102
7-22. Algorithm accuracy (default discretization) on Thermodynamic ODE Model on VSC network.	103
A-1. $m = 3$ case: $S_1 \cap S_2 = \{A, B, C\}$	109
A-2. $m = 2$ case: $S_1 \cap S_2 = \{A, B\}$	109

List of Tables

4-1. Summary of GRN inference method evaluation protocols.	42
6-1. Implementations of Algorithms Tested	62
6-2. Topological features of GRNs for simulation study.	65
7-1. Ranking of GRN inference algorithms by AUROC	78
7-2. Influence of network topology on GRN inference algorithm performance . . .	80
7-3. Ranking of GRN inference algorithms by accuracy with default discretization	81

Contents

List of Figures	viii
List of Tables	x
1. Introduction	1
2. Biological Background: The Concept of a GRN	4
2.1. Genetics and Genomics: Basic Concepts	4
2.2. Gene Expression and its Regulation	6
2.3. Gene Regulatory Networks	7
2.4. Gene Expression Data	10
3. Mathematical and Statistical Background: Graph Theory and Measures of Statistical Dependency	13
3.1. Basic Graph Theory	13
3.2. Dependency measures	19
3.2.1. Copulas	23
4. Gene Regulatory Network Inference Algorithms and their Evaluation	25
4.1. Methods in Bioinformatics for Gene Regulatory Network Inference	25
4.1.1. Methods based on pairwise dependency measures	26
4.1.2. Methods based on regression	31
4.1.3. Methods based on Bayesian Networks	34
4.2. Evaluation of GRN Inference Methods: Literature and Conceptual Issues . .	39
4.2.1. Algorithmic Reconstruction or Statistical Inference?	40
5. Mathematical and Statistical Models of Gene Expression	43
5.1. Causal Graphical Models	43
5.2. Dynamic Models of Gene Expression	52
6. Methods	57
6.1. Statistical Models for Gene Expression Data	57
6.2. Methods: Theoretical Analysis of Relevance Networks	59
6.3. Methods: Simulation Study	60
6.3.1. Inference Algorithms	60

6.3.2.	Models for Simulation: Causal Graphical Models with Gaussian Copula SEMs	62
6.3.3.	Models for Simulation: Causal Graphical Model with Non-Linear SEM	65
6.3.4.	Models for Simulation: Thermodynamic ODE Model of Gene Expression	66
6.3.5.	Evaluation Metrics	68
7.	Results	71
7.1.	Theoretical Observations for Relevance Networks	71
7.2.	Simulation Study	78
7.2.1.	Gaussian Copula SEM Simulations	78
7.2.2.	Cosine transformation SEM and Thermodynamic ODE Simulations .	84
8.	Conclusions	104
A.	Appendix: Proofs	107
A.1.	Proofs of Propositions 5 and 6	107
A.2.	Proof of Proposition 8	109
	Bibliography	112

1. Introduction

Gene regulatory networks (GRNs) are directed graphs that represent the relations of influence among genes in a cell. An accurate GRN is helpful for understanding a cell's behavior, and can inform further pursuits in biological research. Also, practical applications can be derived from GRNs, for example in genetic engineering or pharmaceutical chemistry. Thus, given the usefulness of GRNs, technological progress in measurement of gene expression over the past decades years has propelled the problem of inferring GRNs from data into a pre-eminent issue in the fields of systems biology and bioinformatics.

Inference of GRNs is difficult. Practical challenges include technical obstacles to accurate measurement of gene expression, the large number of variables in a genomic data set, and the difficulty of conducting experimental interventions and consequent reliance on observational data. Importantly, the non-traditional goal of recovering a graph-valued parameter from data places GRN inference outside of the boundaries of routine inferential statistical practice. Altogether, these issues make construction of GRNs a challenging open problem for contemporary statistical research.

Against the backdrop of the above mentioned challenges, the practical relevance of GRNs has spurred a large literature about GRN reconstruction from data. In the past 20 years, many GRN reconstruction algorithms have been proposed in the bioinformatics literature. They use approaches from a variety of perspectives and disciplines including computer science, statistics, and other areas of applied mathematics. These inference methods draw upon disparate statistical and computational techniques to construct graphs from gene expression data that purportedly represent relations of influence among genes.

An important conceptual difficulty of GRN inference is establishment of the validity of any given inference algorithm. A validation procedure that is widely employed is comparison of the output of an inference algorithm applied to a small number of real or simulated data sets to a set of previously established regulatory relations among genes – a 'ground truth' or 'gold standard' GRN. This kind of test offers some empirical evidence for the usefulness of an inference algorithm, but performance in these tests is not necessarily indicative of performance in hypothetical or actual replications. More exhaustive and more theoretically based arguments are desirable for characterizing the limitations or correctness of any algorithm.

To thoroughly assess an algorithm's validity, we contend that it is necessary to link GRNs, which are abstract graphical representations of relations among genes, to concrete models of how gene expression data is generated on the basis of these relations. This setting allows for the formulation of GRN inference as the *inverse problem* of whether, for a given data generating mechanism, a given inference algorithm can succeed in recovering the GRN that underlies it by processing samples from it. On the contrary, in the absence of an explicit model that ties GRNs to data, GRN inference methods can only be appraised as heuristics.

In this study, we address the problem of assessing the performance of several influential GRN inference methods in the literature under assumed data generating processes for gene expression data. We adopt a statistical point of view which considers GRN construction algorithms as estimators of graph-valued parameters of probabilistic models of gene expression data. This allows us to assess the validity of these algorithms not only by their output on a small number of fixed data sets, but also and most importantly through their statistical properties under repeated sampling, which can be studied theoretically or by exhaustive simulations.

The rest of this document is organized in eight chapters. In Chapter 2, we present the bare minimum biological background that allows us to define gene regulatory networks as abstract representations of causal relations among genes within cells. We also introduce the notion of gene expression data, which results from measuring the quantities of gene products in living organisms, and which constitutes the most common source of information used to infer GRNs. In Chapter 3, we present a mathematical and statistical background to the literature surrounding GRNs and their inference. To this end, we introduce key definitions from graph theory, and we list definitions of commonly used statistical dependency measures, all of which constitute vocabulary that is frequently encountered in the literature.

In Chapter 4, we provide an overview of a selection of influential GRN inference algorithms, and of how these and other algorithms have been evaluated in the literature. We describe in detail 10 GRN inference algorithms that are based on a variety of approaches, including the computation of traditional pairwise dependency measures, the computation of linear and non-linear regression models, and the estimation of Bayesian networks. Also, we review evaluations of the performance of GRN inference algorithms that are found in the literature. With this context, we argue for the need of rigorous, statistically-grounded, evaluation procedures, which analyze the behavior of GRN inference algorithms against the backdrop of explicit models for gene expression data.

In Chapter 5, we review two approaches for modeling gene expression data as arising from complexes of causal interactions among genes which can be represented by GRNs. First, we describe the general theory of causal graphical models and the associated statistical models known as structural equations models (SEM). We recount well-known existing results on the

conditions which guarantee the identifiability (although not necessarily the estimability) of the graphs underlying these data generating mechanisms. In the second place, we discuss dynamic mathematical models, based on differential or difference equations, of gene expression and regulation. Unlike representations of gene expression through standard causal graphical models, dynamic modelling is well suited to describing the true biophysics of gene expression.

In Chapter 6, we present the methods we use to make our own assessments of the GRN inference algorithms from Chapter 4 as estimation procedures for the GRNs underlying the models from Chapter 5. In Chapter 7 we present the results we obtain, and in Chapter 8 we present a discussion in the way of concluding remarks. In terms of theoretical analysis, we build upon existing results on the relation between the covariance matrices and the adjacency matrices of particular causal graphical models to illustrate the pervasiveness of asymptotic mistakes of “Relevance Networks”, one of the reviewed algorithms from Chapter 4. Furthermore, we carry out a comprehensive simulation exercise which offers some insight into the behavior of our selected algorithms under these models. We find that the GRN inference algorithms GENIE3 and TIGRESS perform best among all considered by AUROC, while the PC algorithm is the most accurate among algorithms that output discretized estimates. However, more importantly, we find many patterns to be dependent on the underlying statistical model and causal graph topology. This underscores the need for making explicit the data generating model under which GRN inference algorithms are to be formulated and evaluated.

2. Biological Background: The Concept of a GRN

In this chapter we present a succinct background of biological concepts that allow us to formulate the notion of a gene regulatory network and consider its inference from data. First, in Section 2.1 we give a refresher on the elementary objects of genetics and genomics. Then, in Section 2.2 we give an overview of the processes of gene expression and regulation, which are the mechanisms that allow genes to determine observable characteristics of living beings. This idea of gene regulation allows us to appreciate that genes typically interact, regulating each other's expressions. These webs of interactions can be portrayed by the abstractions of gene regulatory networks, which we define mathematically in Section 2.3. Finally, we discuss the most prevalent type of data used for the scientific determination of GRNs in practice, known as gene expression data, in Section 2.4.

2.1. Genetics and Genomics: Basic Concepts

The concept of genes, as inheritable discrete units of information that determine characteristics of living beings, was first suggested by Gregor Mendel in the 19th century [1]. This was before there was any clear idea of how genes and their mechanisms might be physically instantiated. Mendel, through experimenting with pea plants over several generations, surmised the existence of a 'particle of heredity' which was passed on from parents to offspring. His work was mostly ignored for nearly forty years, until botanists Hugo de Vries, Carl Correns, and Erich von Tschermak reproduced similar findings and rediscovered Mendel's original contribution. In the years following, the term 'gene' was introduced, and the chromosomal theory of inheritance was developed through a synthesis of experimental results. This theory states that the 'discrete units' of heredity are genes, which are located on large molecules in cells known as chromosomes.

In the present, it has been established that genes are, in fact, located on chromosomes. Specifically, genes are known to be segments of DNA, the molecule which, when tightly packed, constitutes chromosomes (Figure 2-1). DNA is a long chain-like molecule consisting of a sequence of paired nucleotides, or bases. While DNA governs the observed complexity of life forms, its basic structure is simple. The double helix structure of DNA follows a

rigid pairing rule between nucleotides on opposite sides of the chain: adenine (A) pairs with thymine (T), while cytosine (C) pairs with guanine (G). This rule implies that the two coupled strands of nucleotides are mirror images of each other, and therefore contain the same information. In contemporary terms, the sequences of nucleotides in genes can be thought of as codes, and genes can be said to encode information.

A gene determines observable characteristics, or phenotypes, by triggering a specific set of chemical reactions that depends on the information it encodes. The central process that enables this is known as gene expression, and is discussed below. This capability of genes to reliably generate traits of living beings, coupled with their ability to be replicated and inherited, is what makes biological evolution possible.

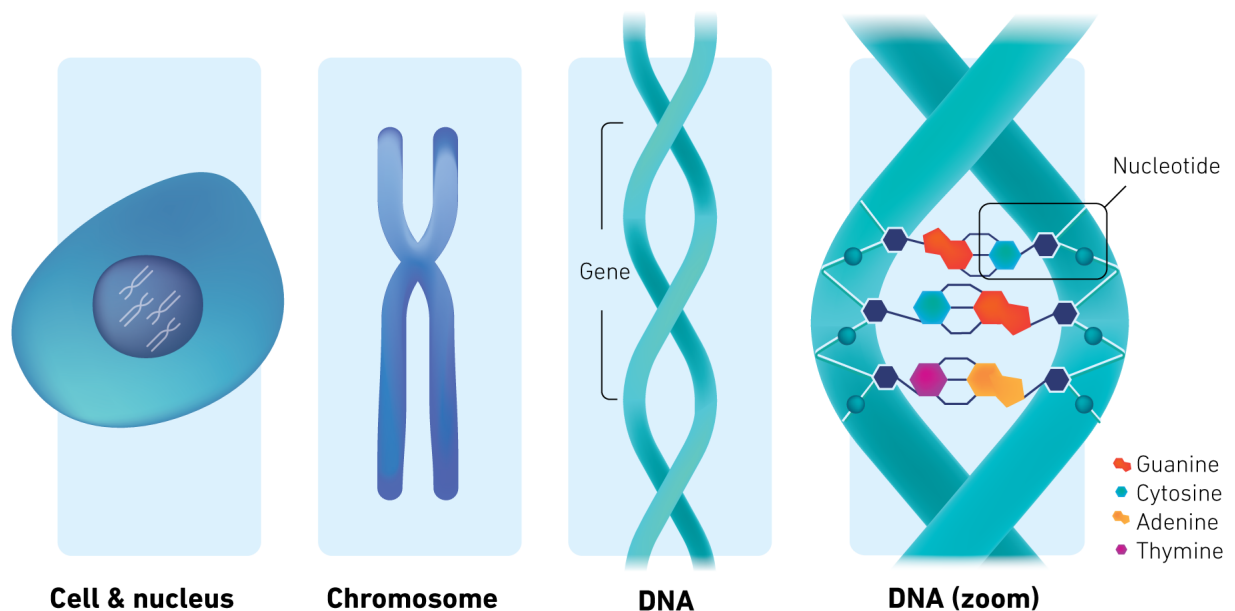


Figure 2-1.: Fundamental concepts of genetics.

The set of all genes in the DNA of an organism is known as its *genome*. The size of a typical genome is in the order of thousands or tens of thousands of genes. Genomes account for a relatively low fraction of the total length of DNA. In human cells it is estimated that genes only cover less than 2% of DNA [2]. A general evolutionary justification for the remaining non-gene DNA is not a settled issue in biology, but many of its functions have been discovered in recent years [3].

2.2. Gene Expression and its Regulation

The general mechanisms by which genes determine traits have been firmly established in the field of biology, to the point of becoming known as the “central dogma of molecular biology”. The central dogma describes a process known as *gene expression*, through which the encoded information in genes may be ‘expressed’ and instantiated in phenotypes. According to the central dogma, gene expression is a two-step procedure whose final output is a protein. In the first stage of gene expression, genes are *transcribed* to a single chain of nucleotides in a process carried out by an RNA-polymerase molecule, as depicted in Figure 2-2. The resulting chain of nucleotides is identical to one of the strands of DNA in the region corresponding to the transcribed gene, except for swapping the nucleotide thymine (T) for uracil (U). Therefore, this molecule contains the same information as the gene, and is thus appropriately known as messenger RNA, or mRNA. In the second stage of gene expression, the information contained in mRNA is *translated* to a chain of aminoacids. This is carried out by a protein known as a ribosome, which performs a mapping of successive triplets of nucleotides to twenty standard aminoacids. The resulting chain of aminoacids is initially chemically unstable, and subsequently folds, forming a protein.

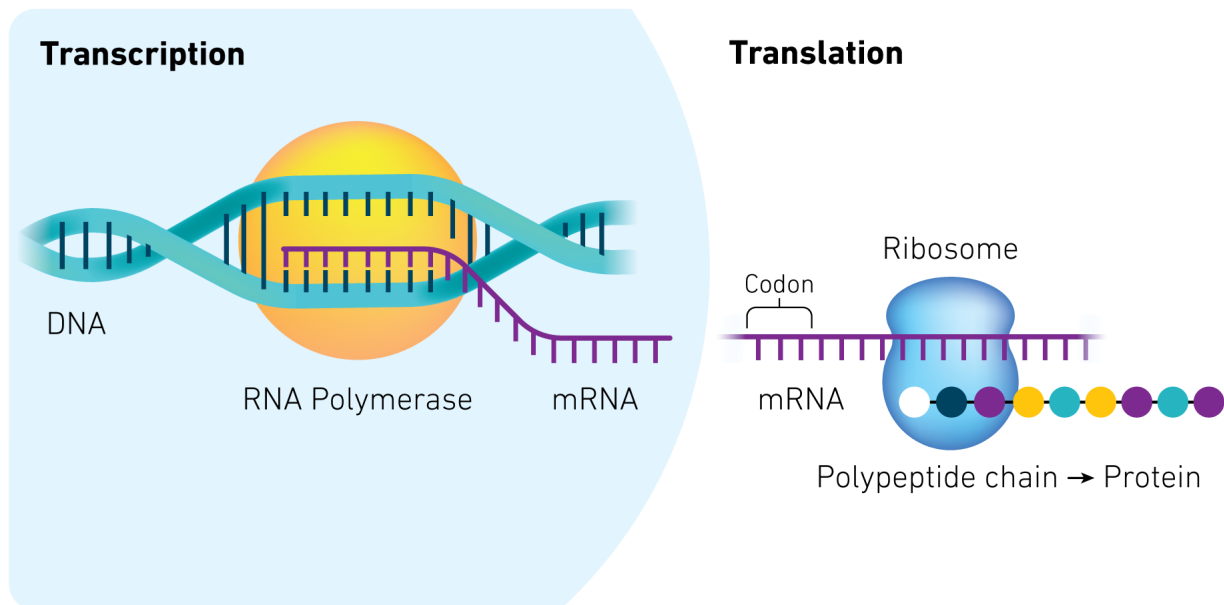


Figure 2-2.: The Central Dogma of Molecular Biology.

The many distinct proteins resulting from gene expression serve a wide range of functions in the maintenance and reproduction of life. In this manner, proteins determine an organism’s phenotype to a large degree, and consequently many biological phenomena can ultimately be traced back to the information encoded in genes, at least in principle if not in practice.

Insofar as gene expression partially determines an organism's behavior, it follows that it must be responsive to conditions in the environment bearing upon the organism's survival. The procurement of energy, defensive mechanisms, and reproduction, for example, all require specific actions at specific times. Since many such actions are rooted in gene expression, there need be mechanisms in place which allow for the expression of a given gene to be triggered or inhibited by environmental pressures. The action of these mechanisms, known as the *regulation* of gene expression, gives rise to varying rates of production of mRNA and proteins across the genome over time.

Gene expression can be regulated at different stages of the overall process, but most regulation occurs during transcription of DNA to RNA. Transcriptional regulation is primarily driven by proteins known as *transcription factors*, which can have positive or negative effects on the rate of transcription, and are therefore classified as *activators* or *repressors*. Commonly, transcription factors work by either recruiting or blocking RNA-polymerase from transcribing a genes. Some transcription factors affect the expression of many genes, in which case they receive the name of *general transcription factors*, while others are specific to a given gene.

2.3. Gene Regulatory Networks

The basic overview of gene regulation provided allows us to formulate a key question. How are regulators of gene expression – transcription factors, in the case of transcriptional regulation – obtained in the first place? The answer is that while some regulators are the natural chemical result of environmental or biological conditions, many others, and in particular transcription factors, *are themselves the protein product of gene expression*. Thus, gene expressions may regulate gene expressions. This means that, at the level of the genome, gene expression is not only a function of exogenous environmental pressures, but of a web of interactions among genes. Consequently, a genome should not be viewed as a set of genes acting in isolation, but instead as a complex system that reacts to external signals in a coordinated manner. The structure of interactions among genes is usually conceptualized as a gene regulatory network, as depicted in Figure 2-3.

A central problem in biology is understanding the functions of genes and the regulatory relationships among them. Scientific knowledge of genes and their functions has increased dramatically in the past decades, but there are many open questions. Although to date the entire genomes of many species have been sequenced, the patterns of nucleotides in a gene do not necessarily provide direct explanations of what their expression accomplishes [5]. Therefore, considerable attention has been paid to understanding gene functions from other kinds of microbiological data in the past two decades. One line of this research program has

Gene regulatory network

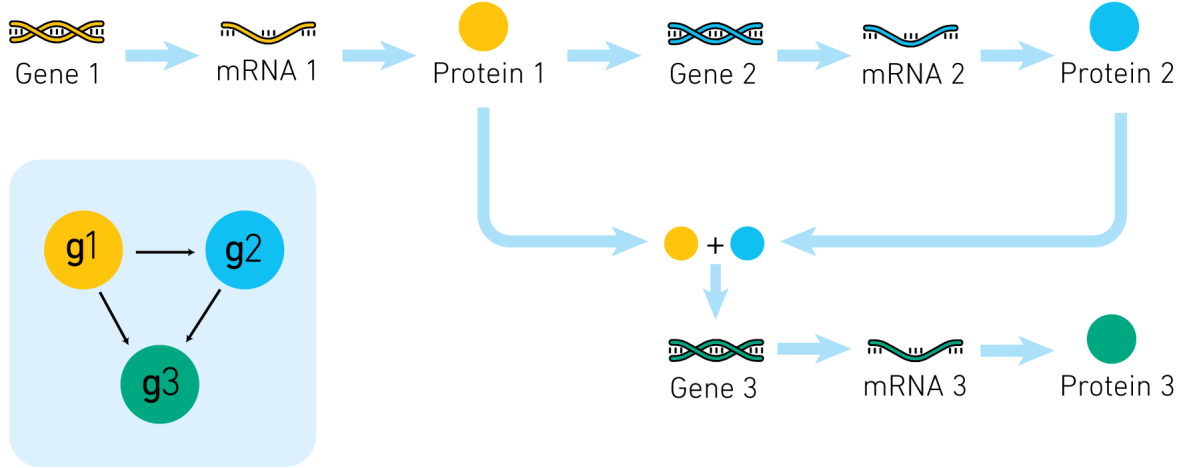


Figure 2-3.: A GRN is a conceptual model that aims to depict the regulatory interactions among gene expressions. This image is our own rendition of Figure 1 in [4].

been attempting to build gene regulatory networks from gene expression data [4]. The focus of this thesis is to assess some of the methods proposed for this task.

To ground the discussion, we introduce a mathematical definition of a gene regulatory network, noting that a natural structure to depict relations of influence is a directed graph.

Definition 1 (Gene regulatory network). A gene regulatory network G is a directed graph (V, E) , where $V = \{v_1, \dots, v_p\}$ is a set of nodes, or vertices, indexed by $\{1, \dots, p\}$, and $E \subseteq V \times V$ is a set of directed edges, or links, between vertices. Directed edges may be thought of as arrows between nodes. We will adopt the convention that a directed edge (v_i, v_j) is oriented from v_i to v_j : its source is v_i and its arrowhead points to its sink, v_j . In the context of gene regulatory networks, each $v_i \in V$ represents a gene, and each edge $(v_i, v_j) \in E$ indicates that the expression of gene v_i regulates the expression of gene v_j . \diamond

Graphs have a very useful representation in adjacency matrices.

Definition 2 (Adjacency matrix). The adjacency matrix A of a directed graph (V, E) is a matrix of dimension $p \times p$, such that

$$A_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{if } (v_i, v_j) \notin E \end{cases}$$

◇

While gene regulatory networks are abstractions, it is important to have a clear idea of their interpretation at the outset. A gene regulatory network is meant to capture relations of influence among genes that obtain from their production of regulators, typically transcription factors. As such, we contend that the links in the network represent causal effects, rather than merely observational coexpression patterns. Without going into a full discussion of causality at this point, what we mean by this is that regulatory relations are stable under experimental intervention. If a gene A regulates a gene B, we expect the levels of expression of gene B to change if a researcher experimentally fixes varying levels of expression of gene A. In contrast, a pattern of coordinated expression that does not reflect a regulatory relation can arise, for example, between two genes that are regulated by a third gene but have no influence upon each other. This distinction between causal and observational associations, explored further in Section 5.1, forms the basis of the notion of spurious causal relations.

A few other remarks are in order regarding the previous definitions. First, we will interpret edges in a gene regulatory network as indicative of “direct” regulatory effects, as opposed to “indirect” effects which are only operative through the mediation of intermediate genes. In second place, although in principle V may be any subset of the genome of an organism, we will take it to be the entire genome unless stated otherwise. Furthermore, the set of edges E , which can be thought of as a binary relation over V , is not restricted in any way at this point, so in principle a gene regulatory network may contain self loops or cycles. Finally, although causal relations represented by a gene regulatory network are presumably directed, in some cases it may be infeasible to infer the directions of edges. To address this case, it is also useful to consider an undirected graph associated to a gene regulatory network known as its *skeleton*.

Definition 3 (Skeleton, Orientation). The skeleton of a directed graph $G = (V, E)$ is an undirected graph $G^* = (V, E^*)$, where E^* is a subset of undirected edges:

$$E^* = \{\{v_i, v_j\} : (v_i, v_j) \in E\}.$$

The adjacency matrix of the undirected graph (V, E^*) is analogously defined as

$$A_{ij}^* = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E \\ 0 & \text{if } \{v_i, v_j\} \notin E \end{cases}$$

Conversely, an *orientation* $G = (V, E)$ of an undirected graph $G^* = (V, E^*)$ is a directed graph such that for each edge $\{v_i, v_j\} \in E^*$, either $(v_i, v_j) \in E$ or $(v_j, v_i) \in E$. ◇

2.4. Gene Expression Data

The key challenge in research on gene regulatory networks is obtaining them in the first place. This constitutes a significant departure from the general field of ‘network science’, where networks are often a given in research questions. In the study of empirical social networks or telecommunications networks, for example, a network is posited and subsequently analyzed. In contrast, most kinds of biological networks cannot be directly observed, but instead are to be inferred from noisy data that depends on the network structure only in an indirect way.

The construction of gene regulatory networks usually relies on *gene expression data*. Gene expression data are direct or indirect measurements of the ‘levels of activity’ of gene expression. Usually, this amounts to measuring the mRNA expressed by each gene in a cell or a sample of cells at a given moment in time. When gene expression data is obtained for an entire genome, as is the usual case in GRN inference, it is understood to be a kind of *genomic data*.

Two of the most popular methods for measuring the intensity of gene expression are microarray experiments and RNA sequencing, or RNA-seq for short. Microarray experiments consist of applying dyed mRNA to a rectangular array of *probes* to which mRNA samples from specific genes bind. The levels of activity of genes can then be indirectly read off, in a continuous scale, from the luminosity or intensity of color of each probe, as shown in Figure 2-4. Microarray experiments have been largely replaced in recent years by RNA-seq methods, which proceed by directly sequencing short segments of mRNA. The sequenced segments are then matched to a reference genome to produce counts of mRNA from each gene at a given point in time. In practice, this procedure must be inspected for overlaps in sequenced segments to avoid double counting.

Inference of gene regulatory networks from gene expression data faces several challenges, involving technical, statistical, and conceptual issues. A central difficulty arises from the observational nature of the data to be used. Ideally, gene regulatory networks can be learned by gene expression data that is experimental in the sense of arising from controlled interventions of gene expressions. In genomics, such experimentation is regularly conducted by means of “knock-out experiments”, in which researchers modify an organism’s DNA to render specific genes unable to be expressed. Although in recent years great strides have been made towards simplifying the implementation of knock-out experiments (in particular, due to CRISPR gene editing technology), conducting the large number of them required to infer a GRN remains a costly and time consuming enterprise. In this situation, currently the construction of a GRN usually involves using at least some observational data, for which exogenous interventions on levels of gene expression are not carried out.

Gene regulatory network inference also is hindered by other particularities of gene expression

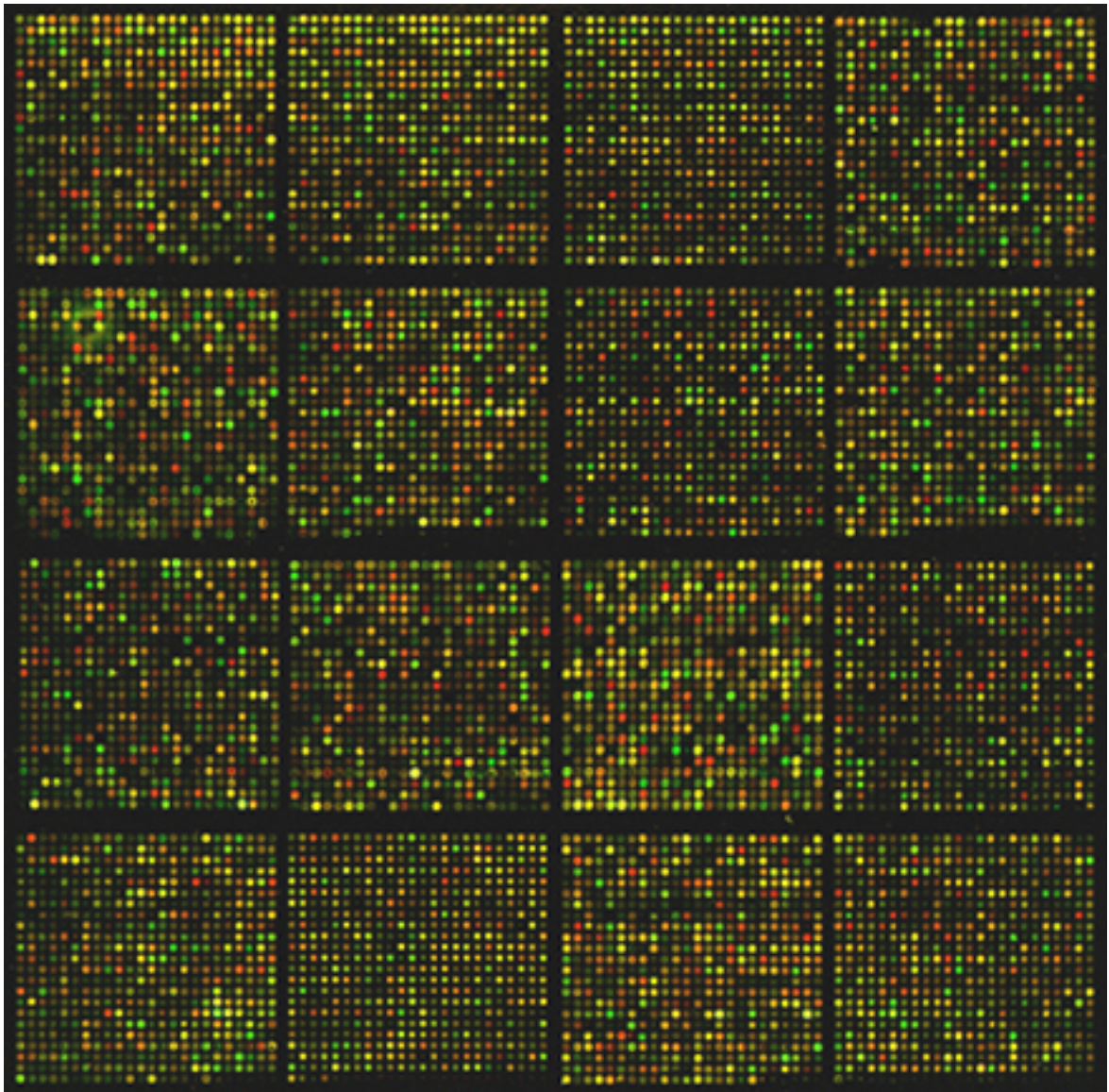


Figure 2-4.: Microarray of mouse gene product, from [6]. Each dot contains a probe of genetic code corresponding to a particular gene, and its coloring represents the corresponding of corresponding mRNA detected in the sample.

data. Gene expression data is frequently noisy and does not conform to standard statistical assumptions. For example, it is common that gene expression data exhibit departures from Gaussian or other frequently encountered distributions [7]. For example, single cell gene expression data commonly suffers from zero-inflation, due to the limitations in sensitivity of measurement technologies. Moreover, given the size of genomes and the cost of measurement, it is common to have less samples than variables in any given data set ($n < p$). Consequently, gene expression data requires analyses that are robust to failures of commonly made assumptions, or that are based on weaker (non-parametric) assumptions. Also, many common statistical procedures requiring, for example, full rank matrices, cannot be carried out.

To discuss gene expression and the construction of gene regulatory networks theoretically, we assume that each observation of gene expression data, indexed by t , is a sample of a random vector X_t of dimension p . In the simplest case, we may take the vectors X_t to be independently and identically distributed, following multivariate distribution P_X . In more realistic settings, measurements in a given gene expression data set may be taken at different times, from different experimental units, or under varying experimental treatments, thus presumably giving rise to temporal or other kinds of dependency structures. We postpone discussion of these cases.

In this context, the general problem of constructing a gene regulatory network can be formulated as follows: Given a sample X_1, \dots, X_n of measurements of expression of the p genes $V = v_1, \dots, v_p$, we seek to specify a preferably directed graph with node set V that represents the causal relations among these genes, that is, a gene regulatory network or its skeleton.

3. Mathematical and Statistical Background: Graph Theory and Measures of Statistical Dependency

In this chapter we provide a background in the mathematical and statistical vocabulary commonly encountered in the literature on GRN inference and its evaluation from a mathematical, statistical, and computational point of view. In Section 3.1 we introduce basic graph theoretical concepts that allow us to examine structural features of gene regulatory networks. Then, in 3.2, we define statistical dependency measures which are commonly used in GRN inference algorithms. This background enables us to review a sample of GRN inference algorithms which have been influential in the field of bioinformatics in the past 20 years, in Chapter 4.

3.1. Basic Graph Theory

Graphs, or networks, are mathematical objects that represent entities and their relations. Definitions for directed and undirected graphs, and for their representations as adjacency matrices, are given in Definitions 1, 2, and 3. With slight modifications, partially directed graphs, which have both directed and undirected edges, can be similarly defined. Then, for a directed, undirected, or partially directed graph $G = (V, E)$, we have the following definitions:

Definition 4 (Basic definitions in graph theory).

- Two nodes v_i, v_j are *adjacent* in G if there is an edge between them: $(v_i, v_j) \in E$ or $(v_j, v_i) \in E$ if G is directed, or $\{v_i, v_j\} \in E$ if G is undirected.
- The adjacency set of a node v_i in G , $adj(G, v_i)$, is the set of nodes adjacent to v_i in G .
- A node v_i is a *parent* of a node v_j , and v_j a *child* of v_i , if there is a directed edge from v_i to v_j , that is, $(v_i, v_j) \in E$.
- The *degree* of a node v_i , $deg(v_i)$, is the number of edges from or to v_i . If G is such that there is at most one edge between any pair of nodes, then $deg(v_i) = |adj(v_i)|$.

- When considering directed edges, the *indegree* of v_i , $\text{deg}^-(v_i)$ is the number of directed edges pointing towards v_i in G , while its *outdegree* $\text{deg}^+(v_i)$ is the number of directed edges whose source is v_i . If G is such that there is at most one edge between any pair of nodes, then $\text{deg}^-(v_i)$ is equal to the number of parents of v_i in G , and $\text{deg}^+(v_i)$ is equal to the number of children of v_i in G .
- A subset of V is called a *clique* in G if every pair of its elements is adjacent in G . If V is a clique, G is said to be a *complete graph*.
- A *path* from a node v_0 to a node v_m is a sequence of adjacent nodes $\hat{V} = (v_0, v_1, \dots, v_m)$ and a sequence of edges $\hat{E} = (e_1, \dots, e_m)$, (\hat{V}, \hat{E}) , such that for $j = 1, \dots, m$, e_j is an edge between nodes v_{j-1} and v_j . A path is *directed* if for $j = 1, \dots, m$, $e_j = (v_{j-1}, v_j)$, and *undirected* otherwise. The *path length* of (\hat{V}, \hat{E}) is $|\hat{E}|$.
- Two nodes v_i, v_j are *connected* if there is a path from v_i to v_j .
- A subset of V is a *connected component* of G if all of its elements are connected to each other. If V is a connected component, G is said to be a *connected graph*.
- A node v_i is an *ancestor* of a node v_j , and v_j a *descendant* of v_i , if there is a directed path from v_i to v_j .
- A *cycle* is a directed path that begins and ends in the same node, that is, a directed path (\hat{V}, \hat{E}) where $v_0 = v_m$.
- A graph G is said to be *cyclic* if it has at least one cycle, and *acyclic* otherwise.
- A *subgraph* G^* of G is a graph whose node set V^* is a subset of V and whose edge set E^* is a subset of E , such that all the nodes that appear in the edges in E^* are in V^* .
- The subgraph *induced by* $V^* \subseteq V$ is the subgraph with node set V^* whose edge set E^* is the set of all edges between nodes in V^* .

◇

When considering the ‘typicality’ of a given trait of a graph – for example, the frequency of a given motif, as defined below –, it is often useful to analyze its prevalence in a set of graphs or a graph formation process which can be considered a “null model”. We introduce the best-known such model, known as the probabilistic *Erdos-Renyi model* of graphs, conventionally associated to the notion of a “random graph”.

Definition 5 (Erdos-Renyi random graph). Given $n \in \mathbb{N} - \{0\}$ and $p \in [0, 1]$, the *Erdos-Renyi random graph* $G(n, p)$ is the undirected graph-valued random variable whose realizations have node set $V = \{1, \dots, n\}$ and for whom each possible edge $\{i, j\}$ is observed with

probability p , independently of other edges. In other words, in the Erdos-Renyi random graph, the indicator variables $\mathbb{1}_{kj}$, for $k, j = 1, \dots, n$, which equal 1 if the edge $\{k, j\}$ is observed and zero otherwise, are independent Bernoulli random variables with parameter p . \diamond

While Definitions 4 allow us to point out rudimentary features of graphs, mostly at a local level, more global and qualitative characteristics of graphs, such as the extent to which its nodes are connected to each other, or the degree to which its nodes are clustered, can also be defined quantitatively and studied. These notions are generally known as ‘topological’ features of graphs. In the following, we recount a series of graph-topological notion which are often encountered in the literature of GRNs. Initially, we consider the distribution of node degrees.

Definition 6 (Empirical and theoretical degree distribution). In an undirected graph $G = (V, E)$, the relative frequency of the degree $n \in \mathbb{N}$ is

$$FD(n) = \frac{|\{v_i \in V : deg(v_i) = n\}|}{|V|}$$

The sequence $(FD(n))_{n=0,1,2,\dots}$ is the *empirical degree distribution* of G . The relative frequency of degree n is the probability that a node chosen from a uniform distribution over V has degree n .

When considering a graph-valued random variable, we may also consider the probability that a node chosen at random (from a uniform distribution) has degree n , which we will write as $P(deg(v) = n)$. Thus, in the context of a theoretical random graph model, we will call the probability distribution of $deg(v)$ its *theoretical degree distribution*, to distinguish it from the realized degree distribution of a given graph. \diamond

Degree distributions of graphs are of interest as they provide some rudimentary information of the structure of a graph. In the *Erdos-Renyi* random graph model, where every pair of nodes has an undirected edge between them with probability $0 < p < 1$, independently from the rest of pairs, the theoretical degree distribution is binomial. In contrast, it has been suggested that many real world networks exhibit empirical degree distributions following *power laws* [8], where $P(deg(v) = n) \propto n^{-\gamma}$, $\gamma > 0$, although this is matter of ongoing debate in the literature [9]. In either case, right-skewed degree distributions such as those with power laws suggest the presence of few *hub nodes* that are adjacent to many others, and in turn many nodes that are adjacent to few nodes.

In the case of gene regulatory networks, there is mixed evidence of power law degree distributions [10]. However, some authors have advanced theoretical explanations for why hub

nodes, right skewness of degree, and, ultimately, power laws in degree distributions could arise in GRNs. Well known arguments are rooted in the evolution of organisms through gene duplication [11] and the efficiency of sparse network structures [12]. From a biological point of view, hub nodes in GRNs may correspond to *master regulators*, which interpret signals from the environment and set in motion complex responses by regulating the activity of several other 'downstream' genes simultaneously. Master regulators, and hubs in general, can therefore be thought of as 'central' in a network.

In graph theory, hub nodes can be identified to be nodes with high degree centrality. Degree centrality is one basic measure to capture the influence of a node in a network structure, while betweenness centrality and eigenvector centrality are two popular alternatives. Betweenness centrality defines the importance of a node by its frequency in the shortest paths across the graph. Meanwhile, eigenvector centrality is defined through the eigenvectors of the adjacency matrix, and can be interpreted as an compromise between degree and betweenness centrality, combining the influence of a node pertaining to the its degree and the influence that it derives from the degree of its neighboring nodes.

Definition 7 (Centrality measures). Let $G = (V, E)$ be an undirected graph. The *degree centrality* of node $v_i \in V$ is defined to be

$$C_D(v_i) = \text{deg}(v_i).$$

The *betweenness centrality* of v_i is defined as

$$C_B(v_i) = \sum_{v_j, v_k \in V, v_i \neq v_j \neq v_k} \frac{\sigma_{jk}(v_i)}{\sigma_{jk}},$$

where σ_{jk} is the number of shortest paths between nodes v_j and v_k , and $\sigma_{jk}(v_i)$ is the number of such shortest paths that pass through node v_i . Finally, the *eigenvector centrality* of v_i is

$$C_E(v_i) = (e)_i,$$

where e is the eigenvector corresponding to the leading eigenvalue of the adjacency matrix of G , A (assuming it has algebraic multiplicity equal to 1).¹ Clearly, since $Ae = \lambda e$, $C_E(v_i)$ is proportional to the sum of the eigenvector centralities of the nodes adjacent to v_i . Because eigenvectors are defined up to multiplication by scalars, eigenvector centralities can in general only be interpreted by their ratios, and by their magnitudes only when $|e|$ is somehow fixed.

◇

¹Alternatively, eigenvector centrality can be defined with the leading eigenvalue and corresponding eigenvector of AD^{-1} , where D is the diagonal matrix whose diagonal holds node degrees ($D = \text{diag}(A\mathbb{1})$). In this case, eigenvector centralities capture the long-run probability that a random walk over the graph finds itself in each node.

Other topological features of graphs involve patterns of connectivity, taking into account not only the degree of individual nodes but the subgraphs they form together. To define and analyze these patterns it is useful to consider a network's *motifs*.

Definition 8 (Graph Isomorphisms and Automorphisms). Two undirected graphs $G^* = (V^*, E^*)$ and $G_m = (V_m, E_m)$ are *isomorphic* if there exists an isomorphism between them. A graph *isomorphism* is a bijection $f : V^* \mapsto V_m$ such that $\{v_i, v_j\} \in E^*$ if and only if $\{f(v_i), f(v_j)\} \in E_m$.

A special case of graph isomorphism is graph *automorphism*. For an undirected graph $G = (V, E)$, an automorphism is a bijection $f : V \mapsto V$ such that $E = \{\{f(v_i), f(v_j)\} : \{v_i, v_j\} \in E\}$. The set of automorphisms of G is known as its *automorphism group*.

For directed and partially directed graphs entirely analogous definitions hold, swapping undirected edges for directed edges.

◇

Definition 9 (Motif). Let $G = (V, E)$ and $G_m = (V_m, E_m)$ be two graphs such that $|V_m| \leq |V|$. If there is at least one subset $V^* \subseteq V$ whose induced subgraph G^* in G is isomorphic to G_m , G_m is called a *motif* in G , and G^* an *instance* of G_m in G . The number of instances of a motif in G is its *frequency* in G .

◇

Motifs are of interest because their frequencies can arguably point to underlying network formation processes. For example, when analyzing social relations, it is often suggested that people with friends in common tend to become friends. This suggests that in network of friendships it is common to observe 'triangle motifs'. This hypothesis can be investigated by assessing the frequency of the motif $G_m = (V_m, E_m)$ with $V_m = \{a, b, c\}$ and $E_m = \{\{a, b\}, \{b, c\}, \{c, a\}\}$ in the graph. In the study of GRNs, it has been suggested that certain network motifs provide stability to gene expression under perturbations from the environment [13]. More generally, the frequency of motifs can give information about structural properties of a graph. For example, triangle motifs are used to define the degree to which a graph exhibits *clustering*, as per the Definition 10 below.

Definition 10 (Clustering coefficients). Let $G = (V, E)$ be an undirected graph and H be the 'triangle motif' undirected graph (V^*, E^*) with $|V^*| = 3$ and $E^* = 2^{V^*} - \{\emptyset, V^*\}$. The *global clustering coefficient* of G is

$$GCC(G) = \frac{Fr_G(H)}{|V|(|V| - 1)},$$

where $Fr_G(H)$ is the frequency of motif H in G . The *local clustering coefficient* of node $v_i \in V$ in G is given by

$$LCC_G(v_i) = \frac{Fr_{v_i}(H)}{|adj(G, v_i)| (|adj(G, v_i)| - 1)},$$

where $Fr_{v_i}(H)$ is the frequency of motif H in the subgraph of G induced by $\{v_i\} \cup adj(G, v_i)$. Finally, the *average clustering coefficient* of G is

$$ACC(G) = \frac{1}{|V|} \sum_{v_i \in V} LCC_G(v_i).$$

◇

The notion of cluster captured by clustering coefficients is that of edge density, in particular that which results in triangle motifs. High global and average clustering coefficients are defining features of *small world* networks. Small world networks are of interest in network science due to their robustness to node deletion and other kinds of perturbations [14, 15]. In small world networks most nodes are not adjacent to each other, but nodes adjacent to a given node are likely to be adjacent to each other, and there tend to be short paths between most pairs of nodes. This kind of arrangement can be achieved through the connections provided by hub nodes in a network: in fact, small world and scale free network structures have been shown to overlap [16].

While clustering coefficients measure the overall presence of densely connected regions in a graph, actually specifying the distinct subsets of nodes that induce such subgraphs requires clustering algorithms analogous to those used for data in Euclidean spaces. Many such algorithms for 'graph clustering' have been advanced in the literature. Central to this line of work are the Laplacian and modularity matrices of a graph, which can be used to define criteria to judge the quality of clusterings, and whose spectra and eigenvectors provide approximations to the optimal clustering under those criteria.

Definition 11 (Laplacian matrix). Let $G = (V, E)$ be an undirected graph with adjacency matrix A . The (*unnormalized*) *Laplacian matrix* of G is

$$L = D - A,$$

where D is the diagonal matrix whose diagonal's i -th component is the degree of node i ($D = \text{diag}(A\mathbb{1})$). As a convention, we denote the n not necessarily distinct eigenvalues of L as $\lambda_1 \leq \dots \leq \lambda_n$.

◇

The Laplacian matrix of a graph is a discrete analogue of the Laplace operator in Euclidean spaces, governing diffusion processes over its nodes. Importantly, the quadratic form associated to this matrix measures the 'ruggedness' of a real-valued function over the node set,

as $x^\top Lx = \sum_{\{v_i, v_j\} \in E} (x_i - x_j)^2$. Therefore, minimizing this quadratic form provides natural clusterings and embeddings of nodes in Euclidean spaces. This problem is intimately related to the spectra of L : the eigenvector associated to λ_i , e_i , minimizes the Rayleigh quotient $\frac{x^\top Lx}{x^\top x}$ subject to the restriction that $x^\top e_j = 0$ for $j < i$. Among other applications of this fact, it can be shown that the signs of the components of the eigenvector associated to λ_2 offer an approximation to the optimal clustering of V into two subsets according to the RatioCut objective function, which consists of a weighted count of the edges between nodes in different clusters [17].

Definition 12 (Modularity matrix). Let $G = (V, E)$ be an undirected graph with adjacency matrix A . The *modularity matrix* of G is

$$M = A - \frac{1}{2|E|} dd^\top,$$

where $d = A\mathbb{1}$. The *modularity of a partition* $\mathbb{A} = \{A_1, \dots, A_k\}$ of V is then given by

$$Q(\mathbb{A}) = \frac{1}{2m} \text{Trace}(S^\top MS),$$

where $S_{ij} = 1$ if $v_i \in A_j$ and $S_{ij} = 0$ otherwise. ◇

Each component of the modularity matrix of a graph compares the presence or absence of an edge with the approximate probability of that edge being present in a random rewiring of the graph that preserves node degrees. Thus, M can be used to assess deviations between the graph and a comparable random graph. Specifically, given $A \subseteq V$ and the membership indicator vector s such that $s_i = 1$ if $v_i \in A$ and $s_i = 0$ otherwise, the value $s^\top Ms$ approximately measures the difference between the number of edges within the subgraph induced by A and the expected number of edges within the subgraph induced by A in a null random model that preserves the original node degrees. A partition of the node set with high modularity can therefore be thought of as specifying regions of the graph that have an unexpectedly high level of interconnection, and thus reflect the 'community structure' of the graph. As a consequence, community detection algorithms in graph theory typically attempt to maximize the modularity of a clustering using the modularity matrix, either by spectral techniques or other heuristics [18, 19].

3.2. Dependency measures

In this section we present the definitions of standard statistical dependency and association measures, and related concepts, such as the concept of a copula. These are frequently employed in analyses of gene expression data, and are often used by algorithms to construct gene regulatory networks, as can be seen in Chapter 4

Definition 13 (Association measures). Let be (X, Y) a real-valued random vector and $(X_1, Y_1), \dots, (X_n, Y_n)$ an independently and identically distributed sample of (X, Y) . The *sample and population Pearson correlation coefficients of X, Y* are, respectively,

$$\widehat{\rho}_{X,Y}^P = \frac{\widehat{\sigma}_{XY}}{\widehat{\sigma}_X \widehat{\sigma}_Y} \quad \text{and} \quad \rho_{X,Y}^P = \frac{\sigma_{XY}}{\sigma_X \sigma_Y}$$

where the σ terms denote the usual population and sample covariance and standard deviations.²

The *sample Kendall τ coefficient* is

$$\widehat{\tau}_{X,Y} = \frac{1}{\binom{n}{k}} \sum_{i=1}^n \sum_{j>i} \text{sign}[(X_i - X_j)(Y_i - Y_j)],$$

The *population Kendall τ coefficient* is

$$\tau_{X,Y} = P\left((X_1 - X_2)(Y_1 - Y_2) > 0\right) - P\left((X_1 - X_2)(Y_1 - Y_2) < 0\right)$$

This is the difference between the probabilities of concordance and discordance of two randomly sampled points from (X, Y) .

The *sample Spearman ρ coefficient* is the sample Pearson correlation of the ranks of the sample points, denoted by $(S_1, R_1), \dots, (S_n, R_n)$,

$$\widehat{\rho}_{X,Y}^S = \frac{\widehat{\sigma}_{R^{[n]}, S^{[n]}}}{\widehat{\sigma}_{R^{[n]}} \widehat{\sigma}_{S^{[n]}}},$$

where the superscripts $[n]$ remind that ranks are formed from a sample of size n .³

The *population Spearman ρ coefficient* is

$$\rho_{X,Y}^S = P\left((X_1 - X_2)(Y_1 - Y_3) > 0\right) - P\left((X_1 - X_2)(Y_1 - Y_3) < 0\right).$$

This coefficient is equal to the difference of probabilities of concordance and discordance between a point sampled from (X, Y) — (X_1, Y_1) — and a point sampled from a bivariate

²Unless explicitly noted, we will adopt as conventional the unbiased sample variance and covariance estimators, with $n - 1$ in the denominator.

³Assuming that X and Y are absolutely continuous, there should be no ties in the sample, in which case $\rho_{X,Y}^S$ simplifies to

$$\rho_{X,Y}^S = 1 - 6 \sum_{i=1}^n (R_i - S_i)^2 / (n(n^2 - 1)).$$

distribution whose marginals are equal to those of (X, Y) but independent — (X_2, Y_3) — [20].

◇

The dependency measures introduced above vary in their sensitivity to the shape of the association and to features of the marginal distributions. Pearson's ρ measures the strength of linear association, whereas Spearman's ρ and Kendall's τ are more general measures of concordance which reflect general monotonic associations between variables. Also, Pearson's ρ is highly dependent on marginal distributions and is sensitive to outliers. In contrast, Spearman's ρ and Kendall's τ are clearly invariant to monotonic transformations of the variables, and are more stable in the presence of outliers.

A further measure of dependency, which we focus on in later chapters, known as mutual information, captures not only concordance or linear association, but general, arbitrary statistical dependencies.

Definition 14 (Entropy and Mutual Information). The *entropy* of a random variable X with (discrete or continuous) probability density function p_X is

$$H(X) = -E(\log p_X(X)).$$

Let (X, Y) be a random vector with joint probability density function $p_{X,Y}$. The *joint entropy* of X and Y is

$$H(X, Y) = -E(\log p_{X,Y}(X, Y)).$$

The *mutual information* of X and Y is

$$MI(X, Y) = H(X) + H(Y) - H(X, Y)$$

◇

Mutual information is of interest because it is invariant to monotonic transformations of X and Y , and, importantly, because $MI(X, Y) \geq 0$, with equality holding if and only if X and Y are independent. This means that mutual information detects any existing pattern of statistical dependency between two variables, including non-linear and non-monotonic relationships. The drawback of this generality is that, barring parametric assumptions, estimating mutual information is difficult. If (X, Y) are assumed to be multivariate Gaussian, then $MI(X, Y)$ is a monotonous function of $|\rho_{X,Y}^P|$, and can be consistently estimated by $\widehat{\rho_{X,Y}^P}$. However, in more general cases, estimating $MI(X, Y)$ is often attempted by way of estimating marginal and joint entropies of (X, Y) .

Definition 15 (Plug-in Estimators of H and MI). Let (X, Y) be a real-valued random vector and $(X_1, Y_1), \dots, (X_n, Y_n)$ independently and identically distributed samples of (X, Y) . *Plug-in estimators of $H(X)$, $H(Y)$, and $H(X, Y)$* are statistics of the form

$$\begin{aligned}\widehat{H}^{p-i}(X) &= -\int_A \log \widehat{p}_X(x) \, d\widehat{p}_X(x), \\ \widehat{H}^{p-i}(Y) &= -\int_B \log \widehat{p}_Y(y) \, d\widehat{p}_Y(y), \\ \widehat{H}^{p-i}(X, Y) &= -\int_C \log \widehat{p}_{X,Y}(x, y) \, d\widehat{p}_{X,Y}(x, y),\end{aligned}$$

where \widehat{p} are estimates of the marginal and joint density functions of X and Y obtained from the random sample, and A , B , and C are measurable subsets of \mathbb{R} , \mathbb{R} , and \mathbb{R}^2 , respectively. In the same manner, a *plug-in estimator of $MI(X, Y)$* is a statistic of the form

$$\widehat{MI}^{p-i}(X, Y) = \widehat{H}^{p-i}(X) + \widehat{H}^{p-i}(Y) - \widehat{H}^{p-i}(X, Y)$$

In particular, suppose \mathbb{U} and \mathbb{V} are collections of disjoint intervals of \mathbb{R} . For example, suppose \mathbb{U} and \mathbb{V} are the sets of bins used to form histograms of the sample. Then, denoting by $\lambda(\cdot)$ the Lebesgue measure of a set in \mathbb{R}^n , a particular case of plug-in estimators of entropy and mutual information are the *maximum likelihood estimators* \widehat{H}^{ML} and \widehat{MI}^{ML} , in which⁴

$$\begin{aligned}\widehat{p}_X(x) &= \sum_{U \in \mathbb{U}} \mathbf{1}_U(x) \lambda(U)^{-1} [\sum_{i=1}^n n^{-1} \mathbf{1}_U(X_i)], \\ \widehat{p}_Y(y) &= \sum_{V \in \mathbb{V}} \mathbf{1}_V(y) \lambda(V)^{-1} [\sum_{i=1}^n n^{-1} \mathbf{1}_V(Y_i)], \\ \widehat{p}_{X,Y}(x, y) &= \sum_{U \in \mathbb{U}} \sum_{V \in \mathbb{V}} \mathbf{1}_U(x) \mathbf{1}_V(y) \lambda(U \times V)^{-1} [\sum_{i=1}^n n^{-1} \mathbf{1}_U(X_i) \mathbf{1}_V(Y_i)].\end{aligned}$$

Other common choices for \widehat{p} are kernel density estimates. It is worth to note that plug-in estimators are generally biased due to the concavity of \log . \diamond

In some situations it is useful to measure the strength of association of two variables once one has conditioned on a set of other variables — a measure of *conditional* association —. Conditional mutual information can serve this purpose.

Definition 16 (Conditional Mutual Information). Let (X, Y) and S be two random vectors defined on the same probability space. The *conditional mutual information of X and Y given S* is

$$MI(X, Y|S) = H(X, S) + H(Y, S) - H(X, Y, S) - H(S)$$

An equivalent expression for $MI(X, Y|S)$ is

⁴In this case, if the norms of the partitions \mathbb{U} and \mathbb{V} shrink towards zero as $n \rightarrow \infty$, then the expressions \widehat{p} provide increasingly accurate and precise estimators of f and its marginals. More precisely, assuming bins of equal width that shrink as $n^{-1/3}$, the mean integrated squared error $E[\int_{\mathbb{R}} (\widehat{p}(x) - p(x)) \, dx]$ converges to zero as $n^{-2/3}$.

$$MI(X, Y|S) = -E_S \left[E_X \left(\log p_{X|S}(X|S) \right) + E_Y \left(\log p_{Y|S}(Y|S) \right) - E_{XY} \left(\log p_{X,Y|S}(X, Y|S) \right) \right]$$

◇

Conditional mutual information, similarly to unconditional mutual information, satisfies $MI(X, Y|S) \geq 0$, with equality holding if and only if X and Y are conditionally independent given S (denoted as $X \perp Y|S$). As such, estimators of conditional mutual information are often central to constraint-based causal graph structure learning algorithms, as discussed in chapters 4 and 5.

In a linear, Gaussian setting, conditional mutual information is a transformation of partial correlation, which can be obtained from the inverse of the covariance matrix [21]. In this case, accurately estimating partial correlations in this case can be achieved by linear regression. However, in general, estimating conditional mutual information is a considerably more challenging task than estimating mutual information. While discrete conditioning variables that take finite values can be relatively amenable to plug-in estimators of conditional mutual information (as in Definition 15), in the case of continuous variables such estimators suffer severely from the curse of dimensionality. It has been shown that, for example, density estimates based on discretized data or kernels deteriorate rapidly with as the dimension of the conditioning set increases. As discussed in 4.1.3 and 5.1, recent research on conditional independence testing suggests that, barring substantive parametric assumptions, conditional mutual information estimation is difficult [22].

3.2.1. Copulas

Several of the above measures of dependence can be shown to be independent of the marginal distributions of variables involved. Rather, they capture properties of the “core” of multivariate distributions that determines dependency, irrespective of the specific forms of the marginal distributions. This concept of “core” is formalized by the notion of *copula* and Sklar’s Theorem.

Definition 17 (Copula, Gaussian Copula). A *copula* $C(u_1, \dots, u_p)$ is a multivariate distribution function for a random vector over $[0, 1]^p$ with uniform marginals. For example, the *Gaussian copula* is given by $\Phi_{\mathbf{R}}(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_n))$, where $\Phi_{\mathbf{R}}$ is the distribution function of a mean zero multivariate normal vector with covariance and correlation matrix both equal to R , and Φ^{-1} is the univariate standard normal quantile function.

◇

Sklar's Theorem [20] states that multivariate distributions of random vectors can be in some sense split between a copula and its marginal distributions.

Theorem 1 (Sklar's Theorem). *Suppose F is the distribution function of a random vector $(X_1, \dots, X_p)^\top$ with marginal distribution functions F_i . Then, there exists a p -dimensional copula C such that F can be expressed as follows:*

$$F(x_1, \dots, x_n) = C(F_1(x_1), \dots, F_n(x_n)).$$

Furthermore, it can be shown that when considering a random vector $(X_1, \dots, X_p)^\top$, its copula C is invariant to monotone transformations of X_i , and it is unique if all variables X_i are continuous [20].

As mentioned, copulas are of interest because they capture patterns of statistical dependence among variables independently of their marginal distributions. Well known dependence measures such as Spearman's ρ , Kendall's τ , mutual information, and conditional mutual information, depend exclusively on the copula that describes the multivariate distribution of variables, and not on their marginals [20]. In evaluating GRN inference algorithms, we consider the Gaussian Copula to make simple extensions to linear Gaussian models that nonetheless admit arbitrary marginal distributions.

4. Gene Regulatory Network Inference Algorithms and their Evaluation

With the background presented in chapters 2 and 3, in this chapter we engage with the literature on GRN inference algorithms and their evaluation. Initially, in Section 4, we review a sample of GRN inference algorithms which have been influential in the field of bioinformatics in the past 20 years. This overview leads us to consider how these methods are usually assessed and validated in practice, which we discuss in Section 4.2. In light of this summary, we argue in 4.2.1 that there should be greater efforts devoted to understanding how many GRN inference algorithms perform as statistical estimators of graph-valued parameters in probabilistic models.

4.1. Methods in Bioinformatics for Gene Regulatory Network Inference

Many methods to build gene regulatory networks have been proposed in the bioinformatics literature in the past 20 years. The variety of methods is a result of different approaches to balancing several concerns in the construction of the network, such as (i) correctly reconstructing a causal structure by excluding spurious or indirect links, (ii) computational efficiency, which determines the feasibility of the algorithm with real data sets, (iii) taking into account temporal or other dependency structures in data, in particular with time series data. To limit the scope of this work, we focus on methods which treat the data as exchangeable in the sense of not explicitly accounting for temporal or spatial dependency, nor measurement of several experimental conditions.

In the following, we discuss a few inference algorithms that have had a large impact in the field of bioinformatics. We organize them in three categories: (i) methods based on pairwise dependency measures, (ii) methods based on regressions, (iii) methods based on Bayesian networks. Although these distinctions are not clear cut, as regression models, dependency measures, and Bayesian networks are conceptually and computationally related, we use this classification as a useful guide to navigate the literature.

4.1.1. Methods based on pairwise dependency measures

A first class of GRN inference methods are based on a similarity matrix among genes obtained by estimating pairwise statistical dependency measures from gene expression profiles, such as linear correlation coefficients, non-parametric measures of association or concordance such as Spearman’s ρ or Kendall’s τ , or estimates of mutual information. In the simplest approach, known as a “Relevance Network”, this similarity matrix is itself taken to be a proxy to the adjacency matrix of the skeleton of the GRN. The remaining methods in this section perform additional steps to modify this similarity matrix before using it to define the adjacencies of a GRN.

Relevance Networks

Relevance networks are the simplest approach to building GRNs. Well known expositions of this approach are found in [23, 24]. A Relevance Network is an undirected network whose edges join pairs of genes for whom some estimated pairwise dependency measure – typically, mutual information – is above some threshold. When the dependency measure used is mutual information and the threshold is chosen on the basis of a statistical test for marginal independence, then Relevance Networks simply aim to reflect the non-null elements of a matrix of pairwise mutual information values.

Alternatively, edges in Relevance Networks may be characterized as attempting to join pairs of genes whose expression levels have non-null values for other dependency measures, such as Kendall’s τ or Spearman’s ρ . Furthermore, under the assumption that the gene expression profiles follow a multivariate Gaussian distribution, non-null mutual information is equivalent non-null covariance or Pearson correlation, in which case a Relevance Network can be specified from appropriate estimates of the latter. In the following, we discuss Relevance Networks specified through estimators of mutual information, specifically.

One issue in building Relevance Networks is appropriately selecting a threshold to discretize the mutual information matrix. One possibility is to use a unique threshold for all entries of the matrix. Such a threshold can be defined on the basis of a significance test for the dependency measure estimated. Alternatively, in [23] the authors propose to select a threshold that makes the output graph’s topology as close as possible to that of a scale-free network. On the other hand, a set of thresholds specific to each estimated dependency score can be specified by, for example, permutations tests.

Clearly, Relevance Networks cannot in general be considered adequate approximations to GRNs, as statistical dependency does not equal causation. For the specific case of a multivariate Gaussian model, the authors of [25] give necessary and sufficient conditions under

which the zero-pattern of the covariance matrix is equal to that of the adjacency matrix of a causal graph. These conditions fail for an overwhelming majority of graph structures, as we show in Chapter 7. Thus, in general, Relevance Networks estimates generally cannot be counted on to distinguish edges that reflect genuine direct causal influence from spurious edges due to common causes or indirect causal influence.

For the reasons given above, in the literature, Relevance Networks usually are (correctly) not interpreted as regulatory networks, but as merely indicative of coordinated gene expression. Despite this caveat, Relevance Networks are influential in the overall literature of biological networks, and are conceptually related to more sophisticated approaches for GRN inference. We thus include them in this study for purposes of benchmarking and comparison.

Algorithm 1: Relevance Network

Input: X , $n \times p$ data matrix.
Parameter: τ , a threshold. Optionally, an exponent $\alpha \geq 1$ to perform 'soft thresholding' (regular thresholding corresponds to $\alpha = 1$).
Output: Skeleton of a GRN, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
- 2 Compute $p \times p$ similarity matrix \hat{S} of gene expression profiles using estimates of mutual information, correlation, or concordance measures.
- 3 **for** all pairs of genes V_i, V_j of genes in dataset **do**
- 4 **if** $\hat{S}_{ij}^\alpha < \tau$ **then**
- 5 Set $E := E - \{\{V_i, V_j\}\}$.
- 6 **end**
- 7 **end**
- 8 **return** G

ARACNe

ARACNe (*Algorithm for the Reconstruction of Accurate Cellular Networks*) was proposed to learn a graph that characterizes statistical dependency of gene expressions in a steady state, and that excludes edges for dependencies arising from indirect interactions [26]. In this approach, the joint probability density of a gene expression profile in steady state is assumed to be given by:

$$f(x_1, \dots, x_p) \propto \exp \left\{ \sum_{i=1}^p \phi_i(x_i) + \sum_{i=1}^p \sum_{j>i}^p \phi_{ij}(x_i, x_j) \right\}$$

This expression for f reflects the assumption that dependency among gene expressions is limited to pairwise interactions. In this scenario, ARACNe seeks to output a graph with edges between genes i and j such that $\phi_{ij} \neq 0$. Beginning with a complete graph, ARACNe proceeds by, first, forming a Relevance Network based on mutual information. In second place, each instance of the triangle motif in the resulting graph is examined, and the edge corresponding to the lowest mutual information value is discarded. This step is intended to remove edges between genes that interact indirectly, only through an intermediate gene, making use of the so-called “data processing inequality”. The authors show that this procedure correctly reconstructs the desired graph structure if it is a tree and if mutual information is estimated without error or, alternatively, asymptotically, given a consistent estimator of mutual information.

While ARACNe refines the approach of Relevance Networks, the statistical graphical model employed is limited. In this model, it is unclear if an edge necessarily represents relations of direct causal influence, as opposed to statistical dependencies with alternative interpretation. Furthermore, the correctness of ARACNe is guaranteed only for very simple graph topologies, and under the strong distributional assumption regarding the pairwise nature of interactions.

Algorithm 2: ARACNe

Input: X , $n \times p$ data matrix.

Parameter: τ , a threshold.

Output: Skeleton of a GRN, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
 - 2 Compute $p \times p$ similarity matrix \hat{S} of gene expression profiles using estimates of mutual information or correlation.
 - 3 **for** all pairs of genes V_i, V_j of genes in dataset **do**
 - 4 **if** $S_{ij} < \tau$ **then**
 - 5 Set $E := E - \{\{V_i, V_j\}\}$
 - 6 **end**
 - 7 **end**
 - 8 **for** all triplets $T = \{V_i, V_j, V_k\}$ of genes whose induced subgraph in G is a triangle **do**
 - 9 Find $e = \operatorname{argmin}_{\{a,b\} \subseteq T} \hat{S}_{a,b}$.
 - 10 Set $E := E - \{e\}$.
 - 11 **end**
 - 12 **return** G
-

CLR networks

CLR (*Context-likelihood of relatedness*) [24] is proposed as another enhancement of Relevance Networks. Instead of directly applying a threshold to an estimated mutual information matrix to obtain a GRN estimate, the authors propose to first rescale each estimated mutual information value. This is done by standardizing elements of the estimated mutual information matrix both row-wise and column-wise, and averaging these surrogate values. The authors justify this transformation by noting that if a particular gene's expression is measured with high error, its observed statistical dependencies to other gene expressions may be weak. This may make edges incident to particular genes more difficult to detect than those incident to other genes. Thus, their proposal of rescaling can be interpreted as seeking to strike a different balance between Type I and Type II errors for different pairs of genes, in such a way to lessen the probability that the resulting graph exhibits disconnected nodes.

The procedure implemented by CLR does not solve the foundational issues of Relevance Networks regarding the lack of causal interpretation of the statistical dependencies found. Furthermore, the transformation of the similarity matrix employed is not obviously based on a sound statistical justification. A simple and statistically valid alternative to account for particularities in the marginal distributions of gene expressions would be to dispense with a unique threshold altogether, and instead perform permutation tests for mutual information. This would result in data-driven thresholds for assessing mutual information that are specific to each pair of genes, without recourse to parametric assumptions, and without arbitrary and intractable modifications to the balance between Type I and Type II errors.

MRNET

MRNET [27] uses estimated mutual information scores to find, for each gene, a set of candidate regulators that predict its expression. This set of predictor variables, which forms the set of parents of the gene in the estimated GRN, is built using a principle of “maximum relevance, minimum redundancy”. This notion is implemented in MRNET as a stepwise variable selection procedure, akin to other similar algorithms in applied statistics. MRNET sequentially includes, and subsequently removes, genes from the predictor set according to a measure of their contributions to predictive power and a measure of their redundancy. Both measures are based on pairwise mutual information estimates. This is expected to exclude indirect causes from the final edge set, since indirect regulators will be presumably highly correlated to the direct regulators of a gene, and therefore will make small contributions to predictive value and at the same time will be deemed highly redundant.

While MRNET presumably produces sets of relevant predictor genes for each gene expression, it is unclear how this goal relates to inferring causal structure. Thus, MRNET raises the

Algorithm 3: Context-likelihood of relatedness (CLR)**Input:** X , $n \times p$ data matrix.**Parameter:** τ , a threshold.**Output:** Skeleton of a GRN, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
- 2 Compute $p \times p$ similarity matrix \hat{S} of gene expression profiles using estimates of mutual information or correlation.
- 3 **for** $i = 1, \dots, p$ **do**
- 4 Compute $\mu_{i,\cdot} = \frac{\sum_j \hat{S}_{i,j}}{|V|}$ and $\sigma_{i,\cdot} = \sqrt{\frac{\sum_j (\hat{S}_{i,j} - \mu_{i,\cdot})^2}{|V|-1}}$.
- 5 **end**
- 6 **for** $j = 1, \dots, p$ **do**
- 7 Compute $\mu_{\cdot,j} = \frac{\sum_i \hat{S}_{i,j}}{|V|}$ and $\sigma_{\cdot,j} = \sqrt{\frac{\sum_i (\hat{S}_{i,j} - \mu_{\cdot,j})^2}{|V|-1}}$.
- 8 **end**
- 9 Calculate matrices R and C as $R_{i,j} = \frac{S_{i,j} - \mu_{i,\cdot}}{\sigma_{i,\cdot}}$ and $C_{i,j} = \frac{S_{i,j} - \mu_{\cdot,j}}{\sigma_{\cdot,j}}$.
- 10 Calculate surrogate similarity matrix \hat{M} as $\hat{M}_{i,j} = \sqrt{R_{i,j}^2 + C_{i,j}^2}$.
- 11 Set $E := \left\{ \{a, b\} : \hat{M}_{a,b} > \tau, a, b \in V \right\}$.
- 12 **return** G

same concerns as previously discussed pairwise dependency-based inference algorithms in terms of accurately reconstructing a GRN.

Algorithm 4: Maximum-relevance minimum-redundancy network (MRNet)

Input: X , $n \times p$ data matrix.
Parameter: τ , a threshold.
Output: Skeleton of a GRN, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
- 2 Compute $p \times p$ similarity matrix \hat{S} of gene expression profiles using estimates of mutual information.
- 3 Initialize surrogate (non-symmetric) score matrix as $\hat{M} = (0)_{|V| \times |V|}$.
- 4 **for** $i = 1, \dots, p$ **do**
- 5 Find gene V_j that maximizes $\hat{S}_{i,j}$.
- 6 Set $\hat{M}_{i,j} := \hat{S}_{i,j}$ and $Q := \{V_j\}$.
- 7 **while** $|Q| < |V| - 1$ **do**
- 8 Find gene V_j that maximizes “maximum relevance, minimum redundancy”
 score $f_{i,j} = \hat{S}_{i,j} - \frac{1}{|Q|} \sum_{k:V_k \in Q} S_{j,k}$.
- 9 Set $\hat{M}_{i,j} := f_{i,j}$ and $Q := Q \cup \{V_j\}$.
- 10 **end**
- 11 **end**
- 12 Set $E := \left\{ \{a, b\} : \max \left(\hat{M}_{a,b}, \hat{M}_{b,a} \right) > \tau, a, b \in V \right\}$.
- 13 **return** G

4.1.2. Methods based on regression

A second class of GRN inference methods estimate regressions to assess the presence and absence of edges. Regressions with linear predictors provide, in their coefficients, a natural way to assess the existence and strength of a statistical dependency. More flexible, non-parametric, regression algorithms can also be used for this purpose. Estimating regression models instead of directly computing pairwise dependency measures can serve several purposes, including (i) implementing shrunk estimators via penalization to control variance, which can be especially useful in $n \ll p$ settings, and (ii) effectively detecting non-linear, and possibly non-monotonic, statistical dependencies. Typically, the cost of these improvements is an increase in computational complexity.

In the following, we review three influential regression-based GRN inference algorithms.

NARROMI

NARROMI (*Noise and Redundancy Reduction using Recursive Optimization and Mutual Information*) [28] combines mutual information Relevance Networks with the estimation of least absolute deviations regressions with $L1$ penalty (LAD-LASSO regressions) to find sets of candidate regulators for each gene. Since LAD-LASSO regressions in general converge on corner solutions with regression coefficients set to zero, these are performed recursively until stable subsets of predictive genes are found for each target gene. The estimate of a GRN is then obtained by averaging an estimated mutual information matrix and the matrix of estimated regression coefficients.

The authors of NARROMI justify the estimation of a linear regression for GRN inference through an analysis of a deterministic ordinary differential equations model for gene expression and regulation. They note that, after a logarithm transformation, the steady state values of gene expression values in this model are linear functions of their regulators' expression values. Although this model does not include randomness in gene expression, it nonetheless provides a theoretical rationale for the inference algorithm. Moreover, in this context, least absolute deviations loss with penalization is adopted as a strategy to improve estimation with outliers and with data sets with $n \ll p$.

Some results in the literature suggest that penalized linear regressions can conceivably be used to accurately infer graphical models from data. For example, in [29] the authors show that LASSO regressions provide consistent estimates of the moral graph associated to a Gaussian causal graphical model (see Section 5.1), assuming well tuned sequences of penalty weights. This suggests possibility that LAD-LASSO regressions, and hence NARROMI, can be used successfully for the same purpose. Nonetheless, to our knowledge, there is no such analogous result in the literature that gives statistical support to the procedure implemented in NARROMI.

TIGRESS

TIGRESS (*Trustful Inference of Gene REgulation using Stability Selection*) [30] specifies a GRN estimate based on a regression algorithm for high-dimensional data known as least angle regression (LARS). LARS approximates the observation of the response variable by a sequence of linear predictors in which predictor variables are included in a stepwise manner, in a procedure similar to forward stepwise regression [31]. LARS regression proceeds by following a path of linear predictors that bisect the angles between currently active predictor variables, and by including new predictor variables when their correlation to the current residual vector equals that of the current linear predictor. This procedure can be shown to produce a shrunk linear regression estimator similar to that of LASSO regression.

Algorithm 5: NARROMI

- Input:** X , $n \times p$ data matrix.
- Parameter:** λ , L1-penalty weight. γ threshold for coefficients in iterative LAD-LASSO estimation. ϕ , weight of regression-based score in adjacency matrix. τ , threshold to discretize edges.
- Output:** Weighted, signed, adjacency matrix of GRN A . Discrete directed or undirected GRN, G .
- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
 - 2 Compute $p \times p$ similarity matrix \hat{S} of gene expression profiles using estimates of mutual information.
 - 3 Initialize regression-based score matrix as $\hat{M} = (0)_{|V| \times |V|}$.
 - 4 **for** $i = 1, \dots, p$ **do**
 - 5 Initialize set Q as $\{1, \dots, p\} - \{i\}$, and set Q^* as \emptyset .
 - 6 **while** $Q \neq Q^*$ **do**
 - 7 Set $Q^* := Q$.
 - 8 Calculate coefficients $\hat{\beta}_{ij}$ for $j \in Q^*$ by minimizing loss function $\sum_{k=1}^n |X_{ik} - \sum_{j \in Q^*} \beta_{ij} X_{jk}| + \lambda |\sum_{j \in Q^*} \beta_{ij}|$ (LAD-Lasso regression).
 - 9 Set $Q := \{j : j \in Q^* \wedge \hat{\beta}_{ij} > \gamma\}$.
 - 10 **end**
 - 11 Set $\hat{M}_{j,i} := \hat{\beta}_{ij}$ for $j \in Q$.
 - 12 **end**
 - 13 Compute weighted, signed, non-symmetric adjacency matrix A as $A_{i,j} = \text{sign}(M_{i,j}) (\phi * |M_{i,j}| + (1 - \phi) * S_{i,j})$ for i, j such that $M_{i,j} \neq 0$, and $A_{i,j} = 0$ otherwise.
 - 14 (Optional: discretize as undirected network). Set $E := \{\{a, b\} : \max(\hat{A}_{a,b}, \hat{A}_{b,a}) > \tau, a, b \in V\}$.
 - 15 (Optional: discretize as directed network). Set $E := \{(a, b) : \hat{A}_{a,b} > \tau, a, b \in V\}$.
 - 16 **return** A, G
-

TIGRESS builds a GRN estimate by exploiting the order in which variables are included in the predictor. This order is a natural measure of predictive relevance, and can be used for feature selection in the general case. In TIGRESS, edge weights are determined by a score based on estimated probabilities that a gene expression X_i is included in the first k -steps of the LARS regression for gene expression X_j . These probabilities are estimated through a resampling scheme akin to bootstrapping, dubbed “stability selection”.

The use of LARS regression and the “stability selection” score in TIGRESS are meant to provide estimates of GRN structure that are robust to outliers and are stable in $n \ll p$ settings. While these features are appealing in terms of feature selection for prediction, it is unclear how they bear upon accurately reconstructing a causal graph structure.

GENIE3

GENIE3 (*GEne Network Inference with Ensemble of trees*) [32] builds an estimated GRN based on non-parametric regressions. In particular, GENIE3 uses ensembles of regression trees (for example, random forests) as its regression algorithm. The flexibility of regression trees is meant to allow GENIE3 to capture statistical dependencies that are characterized by complex functional forms other than linear relationships.

GENIE3 fits an ensemble of regression trees for each gene expression on remaining gene expression. Then, given each estimated regression tree for a gene expression X_i , GENIE3 measures the predictive relevance for each predictor variable X_j by the reduction in the X_i 's observed variance that obtains from splitting the sample at tree nodes corresponding to X_j . In Chapter 7 we show that this strategy can be successful in asymptotically estimating (a subset of) the moral graph of a causal graphical model (see Section 5.1).

GENIE3 was the best performing algorithm in the DREAM4 challenge (see Section 4.2). If employing random forests as the tree ensemble algorithm, the run time of GENIE3 is $O(pn \log nTK)$, where K is the number of candidate variables to be considered for splitting at each tree node in a random forest, and T is the number of trees in each ensemble.

4.1.3. Methods based on Bayesian Networks

A third class of methods in the literature are formulated within the theoretical framework of Bayesian networks. In this theory, discussed further in 5.1, absent edges represent *conditional* independence relationships among variables. This implies that two variables whose statistical relation is entirely mediated by other variables will *not* have an edge linking them, despite them being statistically dependent. In GRN inference, this feature may be desirable as it

Algorithm 6: TIGRESS

Input: X , $n \times p$ data matrix.

Parameter: L , number of predictor genes to include in LARS estimates. τ , threshold to discretize edges. L , number of LARS steps. R , number of stability selection replicates. α , parameter for random noise in stability selection.

Output: Weighted, signed, adjacency matrix of GRN, A . Discrete directed or undirected GRN estimate, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
- 2 Initialize regression-based weighted adjacency matrix as $A = (0)_{|V| \times |V|}$.
- 3 **for** genes $i = 1, \dots, p$ **do**
- 4 **for** replicates $r = 1, \dots, R$ **do**
- 5 Generate a random partition $\{W_1, W_2\}$ of $\{1, \dots, n\}$ with $|W_1| = |W_2|$ (if n is odd, fix $|W_1| + 1 = |W_2|$). Obtain subsamples of dataset given by this partition: two subsamples of expression vector of gene i , $X_{w_1, i}, X_{w_2, i}$, and two subsamples of the data matrix excluding gene i , $X_{w_1, -i}, X_{w_2, -i}$.
- 6 Multiply columns c of $X_{w_1, -i}$ and $X_{w_2, -i}$ by $2(p-1)$ i.i.d. random scalars $\alpha_c \sim U(0, 1)$ to rescale variables.
- 7 Run LARS regression up to the inclusion of the L -th variable on both partitions of the data set, taking expression vectors of gene i as the predicted variables and rescaled gene expression vectors as the predictors.
- 8 Save order of inclusion of predictors in both LARS regressions performed.
- 9 **end**
- 10 **for** each predictor variable $j \neq i$ **do**
- 11 **for** each step $1, \dots, L$ **do**
- 12 Calculate $F(j, l)$, the proportion of the $2R$ least angle regressions in which variable j is included in the top l predictor variables.
- 13 **end**
- 14 Compute score $S_j = \frac{1}{L} \sum_{l=1}^L F(j, l)$
- 15 **end**
- 16 Set edge weights in adjacency matrix as $A_{ji} := S_j$ for $j \neq i$.
- 17 **end**
- 18 (Optional: discretize as undirected network). Set $E := \left\{ \{a, b\} : \max(\hat{A}_{a,b}, \hat{A}_{b,a}) > \tau, a, b \in V \right\}$.
- 19 (Optional: discretize as directed network). Set $E := \left\{ (a, b) : \hat{A}_{a,b} > \tau, a, b \in V \right\}$.
- 20 **return** A, G

Algorithm 7: GENIE3

Input: X , $n \times p$ data matrix.

Parameter: Auxiliary parameters to estimate regression trees. Auxiliary parameters to combine regression trees in an ensemble.

Output: Weighted adjacency matrix of GRN, A . Discrete directed or undirected GRN estimate, G .

- 1 Set $G = (V, E)$ as the complete undirected graph over genes V .
- 2 Standardize columns of data matrix X to z-scores.
- 3 Initialize regression-based weighted adjacency matrix as $A = (0)_{|V| \times |V|}$.
- 4 **for** genes $i = 1, \dots, p$ **do**
- 5 Using predefined auxiliary parameters, fit tree ensemble regression (e.g. random forest) of gene expression i with other gene expression profiles as predictor variables. Denote T as the set of regression trees in the ensemble.
- 6 **for** each tree $t \in T$ **do**
- 7 **for** each non-terminal node w in t **do**
- 8 Calculate influence score of w as

$$I(w) = |X_{w,i}| \text{Var}(X_{w,i}) - |X_{w_T,i}| \text{Var}(X_{w_T,i}) - |X_{w_F,i}| \text{Var}(X_{w_F,i}),$$
 where $X_{w,i}$ are the samples of gene expression i that reach node w , and $X_{w_T,i}$, $X_{w_F,i}$ are the samples that pass and fail the splitting condition at node w , respectively.
- 9 **end**
- 10 **for** predictor variables $j \neq i$ **do**
- 11 Calculate score of predictor j in tree t , $S_{j,t}$, as the sum of node scores $I(w)$ corresponding to nodes where the splitting condition is given on variable j . If variable j is not used to split at any node, set $S_{j,t}$ to 0.
- 12 **end**
- 13 **end**
- 14 **for** predictor variables $j \neq i$ **do**
- 15 Calculate ensemble-wide score of predictor j , S_j , as the average of scores $S_{j,t}$.
- 16 **end**
- 17 Set edge weights in adjacency matrix as $A_{ji} := S_j$ for $j \neq i$.
- 18 **end**
- 19 (Optional: discretize as undirected network). Set

$$E := \left\{ \{a, b\} : \max \left(\hat{A}_{a,b}, \hat{A}_{b,a} \right) > \tau, a, b \in V \right\}.$$
- 20 (Optional: discretize as directed network). Set $E := \left\{ (a, b) : \hat{A}_{a,b} > \tau, a, b \in V \right\}$.
- 21 **return** A, G

can be interpreted as excluding edges that reflect only indirect regulation. More generally, as will be seen in 5.1, Bayesian networks are proposed as a abstract structure for representing general causal processes. Here, we present one of the the most influential inference algorithms for Bayesian networks in the literature, known as the PC algorithm. Another influential approach based on Bayesian networks, not discussed here, assumes that variables follow a multivariate Gaussian distribution, in which case estimating their Bayesian network is reduced to estimating their precision matrix; this is known as the Gaussian Graphical Model [33, 34].

PC Algorithm

The PC algorithm aims to reconstruct a Bayesian network that characterizes the joint distribution of a set of variables. Originally formulated in [35] as a method for general causal graphical models, it has since been implemented for GRN inference in, for example, [36]. The PC algorithm proceeds by removing edges from a complete graph according to conditional independence relations found between variables, considering increasingly large conditioning variable sets. Intuitively, this approach removes edges that correspond to statistical dependencies that arise from arbitrarily indirect, “high-order”, causal relations, leaving only edges that represent direct causal relations in the output graph.

The authors of [35] prove that the PC algorithm provides a pointwise consistent estimator of the skeleton of a Bayesian network, given assumptions detailed in 5.1 and, crucially, the availability of a pointwise consistent test of conditional independence [37]. Under the assumption of multivariate Gaussian data, such a test can be implemented by estimating and testing for the nullity of partial correlations. However, more broadly, the existence of a suitable conditional independence test is not guaranteed, and is a field of current research.

Strictly speaking, the PC algorithm outputs a partially directed graph. In Algorithm 8, we present only a subset of steps of the PC algorithm that output an estimate of the undirected skeleton of the directed Bayesian network. Additional steps, not shown here, exploit the statistical properties of ‘collider motifs’, discussed in 5.1, to orient a subset of the edges. Also, we present these steps as formulated for the PC-stable algorithm proposed in [38], which guarantees that the output is invariant to the order in which variables are selected to test for conditional independence.

Algorithm 8: PC-Stable Algorithm (First Step)

Input: X , $n \times p$ data matrix.

Parameter: A (statistical) test for conditional independence of pairs of variables $\{X, Y\}$, $X \perp Y | S$, that admits conditioning sets of variables S of arbitrary size. An arbitrary ordering $\mathbf{O}(V)$ over variables/nodes.

Output: Discrete undirected GRN estimate, G .

```

1 Set  $G = (V, E)$  as the complete undirected graph over genes  $V$ .
2 Initialize  $l := -1$ 
3 repeat
4   Set  $l := l + 1$ .
5   for all nodes  $v_i \in V$  do
6     | Set  $a(X_i) := \text{adj}(G, v_i)$ 
7   end
8   repeat
9     | Using  $\mathbf{O}(V)$ , select an ordered pair of vertices  $(v_i, v_j)$  that are adjacent in  $G$ 
        | and for which  $|a(v_i) - \{v_j\}| \geq l$ .
10    repeat
11      | Using  $\mathbf{O}(V)$ , select a subset of nodes  $S \subseteq a(v_i) - \{v_j\}$  such that  $|S| = l$ .
12      | if a conditional independence test for  $X_i \perp X_j | S$  has not been performed
        | in a previous step of the algorithm then
13        | | Perform conditional independence test for  $X_i \perp X_j | S$ .
14        | | if it is deemed that  $X_i \perp X_j | S$  then
15        | | | Set  $E := E - \{v_i, v_j\}$ .
16        | | end
17      | end
18    until  $\{v_i, v_j\} \notin E$  or all  $S \subseteq a(v_i) - \{v_j\}$  such that  $|S| = l$  have been
        | examined;
19  until all ordered pairs of vertices  $(v_i, v_j)$  adjacent in  $G$  with  $|a(v_i) - \{v_j\}| \geq l$ 
        | have been considered;
20 until all pairs of adjacent vertices  $(v_i, v_j)$  in  $G$  satisfy  $|a(v_i) - \{v_j\}| < l$ ;
21 return  $G$ 

```

4.2. Evaluation of GRN Inference Methods: Literature and Conceptual Issues

Given the variety of methods advanced for estimating GRNs, a critical question is whether these methods in fact produce good estimates. In a more nuanced view, one should be concerned with a given method's performance over a domain of applicability, and not over all conceivable scenarios. In either case, to perform this kind of assessment it is necessary to compare the estimates from an inference algorithm with 'ground truth' or 'gold standard' GRNs. Below we briefly discuss evaluations of GRNs in the literature.

Evaluations of GRN inference methods are provided, first, as these methods are proposed. For the methods reviewed in the previous section, the authors argue for the usefulness of the algorithms with two general approaches. A summary of these arguments is presented in Table 6-1. On one hand, the proposed algorithms are applied to real gene expression data sets and shown to recover previously known regulatory relations in the organisms studied. Also, the authors consider artificial data sets, simulated from a mathematical model of gene expression in a predefined GRN, and show that the algorithms proposed correctly reconstruct this GRN to a certain degree.

Comprehensive evaluations of GRN inference methods have also been carried out. One landmark project in this field has been the DREAM challenges [40]. DREAM is an open science collaborative initiative to examine complex questions in biology and medicine. Since 2006, DREAM has organized a series of challenges open to researchers to promote the development of algorithms to analyze biological data. In particular, DREAM challenges 3 to 6, held between 2008 and 2011, included a component of 'reverse engineering in silico GRNs' from simulated data. In some tasks, lists of regulators and target genes were provided beforehand, so that the GRN to be estimated was constrained in advance. The data was generated using the well known *GeneNetWeaver* simulator [41], whose underlying model for gene expression is discussed in 5.2. The overall conclusion from these DREAM challenges was that GRN inference methods based on different approaches can have complementary strengths and weaknesses, so that a particular inference task can benefit from the use of more than one algorithm [42].

Apart from the DREAM challenges, other research projects have contributed to the assessment GRN inference algorithms, with a wide range of results. In simulation-based studies, GRN inference algorithms have been found to be notably dependent on the assumed data generating model [43], on sample size [44], on whether data is observational or interventional, and on the employed estimator of mutual information (when applicable) [45]. The success of GRN inference algorithms in published evaluations is varied, with [46] reporting, for example, that well known GRN inference algorithms can perform no better than random

guessing with data simulated to mimic single cell gene expression data. Moreover, in [47, 48] it is proposed that the assessment of GRN inference algorithms itself depends on which metric is used to judge their performance.

In general, the heterogeneity of the literature evaluating GRN inference algorithms makes it difficult to draw widely applicable conclusions. With this in mind, recently Pratapa et al. [49] designed and implemented a comprehensive protocol, named *BEELINE*, to streamline the process of evaluating GRN inference methods using simulated data sets from various theoretical models of gene expression. In their conclusions, they state: “We found considerable variation in the performance of the [reviewed GRN inference] algorithms across the ten different networks (six synthetic and four Boolean) we analyzed. *Nevertheless, we were able to see a few general trends that are noteworthy*” [our emphasis]. This underscores the need for further research on this topic.

4.2.1. Algorithmic Reconstruction or Statistical Inference?

Two salient features of the literature on evaluations of GRN inference methods are the dearth of theoretical analyses of algorithm correctness and the small number of data sets used in simulation tests. On the first observation, we note that some theoretical results do exist for the general purpose of inferring causal graphs. For example, as mentioned in discussing NARROMI, [29] shows that LASSO regressions can asymptotically recover moral graphs. Similarly, as mentioned, [25] gives necessary and sufficient conditions for the coincidence of mutual information matrices and causal graph structures. Also, statistical properties of algorithms based on Bayesian Networks, such as the PC algorithm, are often well documented in the literature.

Nevertheless, for many influential GRN inference algorithms arising out of the field of bioinformatics, such as those reviewed in Section 4.1, we note that they generally are not argued for on the basis of statistical theory. This tendency, together with small number of data sets used in simulation tests, reflects that, to a great degree, GRN inference methods are not explicitly proposed as estimators of a graph-valued functional of a statistical model. Although GRN inference methods invariably involve applying statistical procedures on the basis of some rationale, they are primarily not treated as methods for statistical inference.

In a strictly computational sense, estimating a gene regulatory network from gene expression data effectively requires applying an algorithm whose output is a network to a data set. However, emphasizing the computational properties of such an algorithm can come with the risk of overlooking its statistical properties. Such statistical properties are important to make sense of the validity of an algorithm. Without theoretical results or systematic probing based on repeated sampling for these statistical properties, at least two important

issues arise in evaluations of GRN inference algorithms:

- the inability to distinguish systematic from random mistakes (for example, in specifying edges), and
- an insufficient notion of the variability of the output graph to randomness in the input data.

In this context, we contend that to address these questions of reliability and adequacy it can be worthwhile to cast the construction of gene regulatory networks as a problem in statistical inference. In this view, a gene regulatory network is an underlying parameter that characterizes, and to some degree determines, the probability distribution of gene expression data, and is to be recovered from samples therefrom. This conception of the problem allows one to naturally identify the adequacy of algorithms by their performance under repeated sampling from a probabilistic model, considering them as estimators.

To pursue the statistical approach outlined above requires defining explicit multivariate statistical models of gene expression. Moreover, such models must lend themselves to defining graph-valued parameters that can reasonably be interpreted as a gene regulatory network. We turn our attention to such models in the following chapter.

Table 4-1.: Summary of GRN inference method evaluation protocols.

Algorithm	Data sets	Number of nodes	Replicates in simulated data	Results
Relevance networks [23]	55 samples of brain cancer data. 44 samples of yeast cell-cycle microarrays.	8000 genes in cancer data set. 4000 genes in cancer data set.	—	Modules in output graphs are enriched in biologically relevant ways. Essential genes have high connectivity.
ARACNe [26]	Data set of samples from human B cells. Two simulated data sets from Hill kinetics model in stationary state.	100 genes in synthetic networks. ~12000 genes in human B cell samples.	Results are average of 3 replicates.	ARACNe outperforms competing Relevance Networks and Bayesian Networks algorithms in terms of AUROC. Better performance on random graphs than on scale-free graphs.
CLR [39]	445 <i>E. coli</i> gene expression profiles	4345 genes.	—	CLR identified 1079 regulatory relations. 338 were already known and 741 were novel. 21 novel relations were later confirmed with experiments.
MRNET [27]	30 synthetic data sets of 100 to 1000 observations simulated from gene expression models <i>sRogers</i> and <i>SysTRen</i>	100 to 700 genes in all data sets.	Data sets for different model configurations are simulated once.	MRNET has competitive performance compared to CLR and ARACNe in terms of AUROC.
NARROMI [28]	Data sets simulated from linear Gaussian SEM. Synthetic data sets from DREAM3 challenge. <i>E. coli</i> data from Many Microbe Microarrays compendium.	4297 <i>E. coli</i> genes. 10 to 5000 genes in linear SEMs. 10 and 30 nodes in DREAM3 data sets.	Data sets for different model configurations are simulated once.	NARROMI competes with TIGRESS and GENIE3 on synthetic data sets, and performs especially well with many genes.
TIGRESS [30]	8 simulated data sets from DREAM4 and DREAM5 challenges. 907 <i>E. coli</i> microarrays from Many Microbe Microarrays compendium.	4297 <i>E. coli</i> genes. 100 to 5950 genes in simulated data sets.	Data sets for different model configurations are simulated once.	TIGRESS ranked third overall in the DREAM5 challenge. It performs similarly to GENIE3 on simulated data.
GENIE3 [32]	5 simulated data sets from DREAM4 challenge. 907 <i>E. coli</i> microarrays from Many Microbe Microarrays compendium.	4297 <i>E. coli</i> genes. 100 genes in DREAM4 data sets.	Data sets for different model configurations are simulated once.	GENIE3 had the best performance among competitors in DREAM4 challenge. Despite this, GENIE3 had poor results with <i>E. coli</i> data.

5. Mathematical and Statistical Models of Gene Expression

In order to have better assessments of the strengths and weaknesses of gene regulatory inference algorithms, we argue at the end of Chapter 4 that a productive approach is to view them as statistical procedures, that is, to consider them as estimators in probabilistic models and study their behavior under repeated sampling. However, to pursue this line of thinking, it is necessary to define adequate probabilistic models of gene expression and regulation, and their graph-valued parameters that will represent the “ground truth” gene regulatory networks to be estimated by gene regulatory network inference algorithms.

The choice of probabilistic models and graph-valued parameters to assess gene regulatory network inference hinges on the interpretation of these models. As gene regulatory networks are meant to represent causal relationships, a fair assessment of algorithms should evaluate inference algorithms in their ability to recover a graph-valued functional of the probability distribution that represents causal relationships among the variables.

In the following, we briefly discuss two frameworks to model gene expression arising from GRNs. The first, causal graphical models, has been studied in depth as a way of encoding causal knowledge in general settings, especially since the seminal work of Pearl [50] and Spirtes, Glymour, and Scheines [35]. In simple cases, the statistical features of these models are well understood and can be exploited to infer the structure of a latent causal graph from data. However, basic causal graphical models are limited in their capacity to express realistic dynamics of gene regulation. Therefore, we discuss a second class of dynamic, phenomenological, models of gene expression that based on differential equations, which provide better descriptions of gene regulation.

5.1. Causal Graphical Models

A causal graph is a directed graph (\mathbf{X}, E) where \mathbf{X} is a finite set of (random) variables and an edge $(X_i, X_j) \in E$ indicates that X_i is a direct cause of X_j . Abusing notation, we will also use \mathbf{X} to denote a (random) vector whose components are variables in \mathbf{X} . This representation of a system of causes and effects accords with informal depictions of causal reasoning in which arrows are drawn from causes to effects. An edge $X_i \rightarrow X_j$ in a causal graph is

interpreted as stating that the value of X_j is fixed by ‘a mechanism in nature’ that relies on the value of X_i . Pearl [50] argues for the need of adopting this sort of causal language to answer many scientific questions, and against attempting to reduce such causal concepts to other, more foundational definitions.

The general formulation of a causal model given above does not obviously tie into observable patterns in data. To explicitly establish this link, we consider structural equations models (SEM).

Definition 18 (Causal Structural Equations Model (SEM)). Suppose $G = (\mathbf{X}, E)$ is a causal graph as outlined above. We assume throughout this document that all variables in \mathbf{X} are observable. A *structural equations model (SEM)* associated to G is a set of pairings (X_i, EQ_i) for each variable $X_i \in \mathbf{X}$, where EQ_i is an equation of the form

$$X_i = f_i(\mathbf{PA}_i, \varepsilon_i),$$

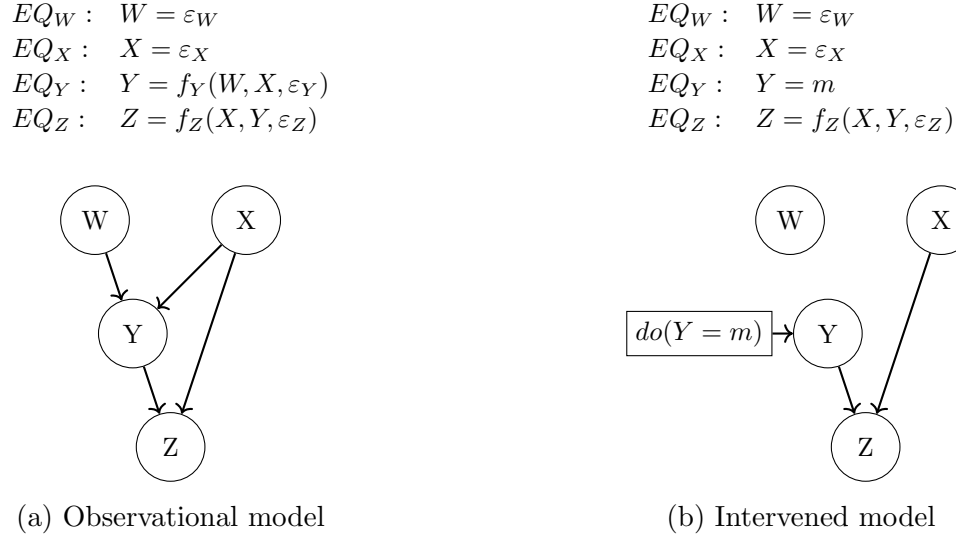
and where f_i is a function whose arguments are the parents of X_i in G , \mathbf{PA}_i , and a perturbation or error term, ε_i . The error term represents unobserved causes of X_i and is usually taken to be a random variable.

◇

In a causal SEM, the equations represent the ‘structural mechanisms’ that assign values to variables based on their causes. They are not to be interpreted as merely a set of observed algebraic relationships among variables, but as ‘assignments’ in the sense of computer science. This distinction lies in that, while a system of equations admits various equivalent algebraic expressions, the pairing of variables to equations in a SEM encodes information about the processes by which variables take their values. In particular, this means that a causal SEM is designed to model the behavior of a system under *external interventions*. An external intervention on a variable X_i in a SEM is represented by replacing the equation $X_i = f_i(\mathbf{PA}_i, \varepsilon_i)$ by $X_i = m_i$, where m_i represents an experimentally fixed value of X_i .¹ This idea is formalized by the “do operator” discussed in [50] (Figure 5-1).

A SEM implies a joint probability distribution for \mathbf{X} . In this context, the question of interest is whether, and under what conditions, it is possible to infer the causal graph associated to a SEM using samples of \mathbf{X} . In what follows, we present conditions to approach this inferential problem in the simplest conceivable case. It should be noted, however, that some of the principles laid out have been extended to more realistic settings (for example, see [35], [51], [52]). The main result, stated below in 5.1, is that the skeleton of a causal graph is an

¹More generally, m_i can represent a possibly stochastic mechanism used to exogenously intervene on the value of X_i



An external intervention $do(Y = m)$ on a causal graph with variables \mathbf{V} is represented as removing inbound edges at Y and replacing EQ_Y by $Y = m$. The intervened model gives rise to a joint distribution $P_{\mathbf{V}|do(Y=m)}$. Note that $P_{\mathbf{V}|do(Y=m)}$ will generally not coincide with the observational distribution $P_{\mathbf{V}|Y=m}$. The interventional distributions is used to define causal effects, e.g., a marginal effect of Y on the mean of Z , $E_{P_{Z|do(Y=m+1)}}(Z) - E_{P_{Z|do(Y=m)}}(Z)$.

Figure 5-1.: Syntax of do-operator in a causal graphical model and associated SEM.

identified parameter of its associated SEM.

Initially, two key assumptions are made:

Assumption 1 (Aciclicality). *Causal graphs are acyclical, that is, for $G = (\mathbf{X}, E)$, if $(X_i, X_j) \in E$, then there is no directed path from X_j to X_i .*

Assumption 2 (Sufficiency). *In a SEM as described in Definition 18, all variables in \mathbf{X} are observed, and the error terms ε_i are mutually independent.*

Assumptions 1 and 2 are restrictive and debatable in applications. The first of them rules out contemporaneous mutual causation among variables: thus, phenomena such as feedback mechanisms can only be expressed in this framework using time-indexed variables, in 'dynamic' SEMs and causal graphs. In turn, the second assumption is interpreted as there being no unobserved common causes of variables in \mathbf{X} , or *confounders*. The absence of unobserved common causes implies that the ε_i terms do not have effects on more than one variable in the model, and hence they must be mutually independent. To satisfy this constraint, a SEM for

a given subject domain must therefore include sufficiently many variables and their causal relationships, in order for the error terms to contain exclusively variable-specific sources of variability.

From Causal Graphs and SEMs to Bayesian Networks

Under Assumptions 1 and 2, the probability distribution of observed variables in a SEM satisfies three essentially equivalent properties known as the *Markov conditions* with respect to the associated causal graph. Below we state two Markov conditions and their equivalence. In general, if a joint distribution P satisfies the Markov conditions with respect to a generic, not necessarily causal, directed acyclic graph G , then G is known in the field of statistics as a Bayesian Network for P . Consequently, the causal networks in this section are commonly known as causal *Bayesian* networks.

Definition 19 (Local and Factorization Markov Conditions). Suppose $G = (\mathbf{X}, E)$ is a causal graph and P is a multivariate distribution of variables \mathbf{X} . Denote by \mathbf{PA}_i the set of parents of X_i in G . P is said to satisfy the *local Markov condition* with respect to G if

$$X_i \perp \mathbf{Y} | \mathbf{PA}_i$$

for each $X_i \in \mathbf{X}$ and each $\mathbf{Y} \subset \mathbf{X}$ whose elements are not descendants of X_i in G excluding X_i itself. If $\mathbf{PA}_i = \emptyset$, then the above statement is to be read as unconditional independence, $X_i \perp \mathbf{Y}$.

Additionally, assuming P has a density p , P is said to satisfy the *factorization Markov condition* with respect to G if

$$p(\mathbf{X} = \mathbf{x}) = \prod_{i=1}^n p(X_i = x_i | \mathbf{PA}_i = \mathbf{pa}_i),$$

where \mathbf{pa}_i is the realization of \mathbf{PA}_i implied by the realization of all observable variables, \mathbf{x} . If $\mathbf{PA}_i = \emptyset$, then the i -th term in the above product is to be understood as the marginal density $p(X_i = x_i)$.

◇

Proposition 1. Let $G = (\mathbf{X}, E)$ be a causal graph with an associated SEM, and P the distribution of \mathbf{X} implied by these. If G and P satisfy Assumptions 1 and 2, then P satisfies the local Markov condition with respect to G .

Proof. See Theorem 3.27 in [21].

□

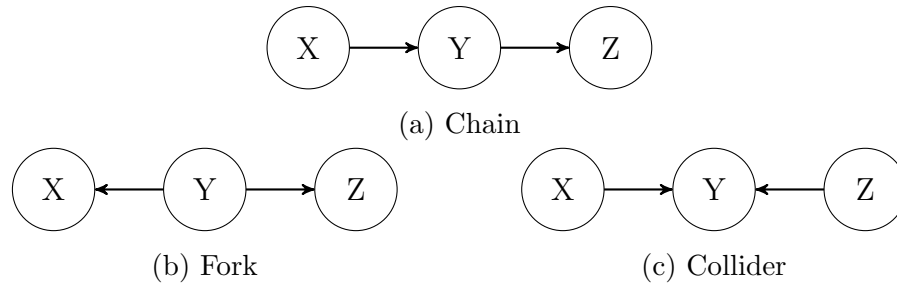


Figure 5-2.: Examples of 3-node causal graphs

Proposition 2. *Let $G = (\mathbf{X}, E)$ be a causal graph with an associated SEM, and P the distribution of X implied by these. If G and P satisfy Assumptions 1 and 2, and P has a density function p , then the local and factorization Markov conditions are equivalent.*

Proof. See Theorem 3.27 in [21]. □

In terms of a causal model, the stated Markov conditions reflect the intuitive notion that once a variable's direct causes are given, its value is determined by an autonomous mechanism that plays out independently from any other processes. Statistically, this conception of causality can be expected to produce a set conditional independence relations among variables; hence, the local Markov condition. Figure 5-2 shows three causal graphs with three nodes each. Assuming underlying SEMs for those causal graphs that satisfy Assumptions 1 and 2, the local Markov condition implies:

- In 5-2a, $X \perp Z|Y$.
- In 5-2b, once again, $X \perp Z|Y$.
- In 5-2c, $X \perp Z$.

When considered as a motif in a larger graph, the graph in Figure 5-2c is known as the *collider motif* or *v-structure motif*. V-structures play a key role in the identification of causal structures, as will be seen shortly.

Statistical Implications of Graph Structure: The Notion of d-Separation

A key observation for the purpose of inferring a causal graph is that all of the conditional independence relations the graph entails, via the local Markov condition, are a function of its skeleton and v-structures. This means that general statistical and probabilistic properties of a causal graphical model can be “read off” its graph. This relationship is established through the notion of *d-separation*.

Definition 20 (d-separation). Let $G = (V, E)$ be a directed acyclic graph, v_i and v_j two distinct nodes in V , and W a subset of V that does not include v_i and v_j . A path $S = (\hat{V}, \hat{E})$ from v_i to v_j in G is said to be *blocked* by W if either of the following conditions holds:

- For some node $w \in W$, $(a, w), (w, b) \in \hat{E}$.
- If a node $w \in \hat{V}$ is such that $(a, w), (b, w) \in \hat{E}$ — in other words, if there is a v-structure in the path \hat{V} at node w —, then $w \notin W$ and $w_{Desc} \notin W$ for all descendants w_{Desc} of w in G .

Two nodes v_i and v_j are *d-separated* in G by $W \subseteq V - \{v_i, v_j\}$ if all paths between them in G are blocked by W . Moreover, two disjoint subsets of V , V_1 and V_2 , are said to be d-separated by a third subset of nodes W if each pair of nodes $v_1 \in V_1$, $v_2 \in V_2$ is d-separated by W . To indicate d-separation, we use the notation $v_i \perp_d v_j | W$ or $V_i \perp_d V_j | W$ for nodes and subsets of nodes, respectively.

◇

In the examples from Figure 5-2, we have the following d-separations:

- In 5-2a, $X \perp_d Z | Y$.
- In 5-2b, once again, $X \perp_d Z | Y$.
- In 5-2c, $X \perp_d Y$.

The concept of d-separation can be used to define an equivalence relation over the set of graphs over a common node set. Additionally, the equivalence classes this relation induces can be shown to contain structurally similar graphs.

Definition 21 (d-separation equivalence). Two directed acyclic graphs $G = (V, E)$ and $G^* = (V, E^*)$ are said to be *d-separation equivalent* if for every three mutually disjoint subsets of V , A , B , and C ,

$$A \perp_d B | C \text{ in } G \iff A \perp_d B | C \text{ in } G^*$$

We denote this relation by $G \sim_d G^*$.

◇

Proposition 3. Let $G = (\mathbf{X}, E)$ and $G^* = (\mathbf{X}, E^*)$ be two causal graphs with corresponding SEMs, and suppose Assumptions 1 and 2 hold for both. $G \sim_d G^*$ if and only if

- a. G and G^* have equal skeletons, and

b. G and G^* have the same instances of v -structures.

Proof. See [53]. □

The usefulness of d-separation for the purpose of graph inference becomes clearer once we note that it allows us to reformulate the above Markov conditions in a third, equivalent, way.

Definition 22 (Global Markov Condition). Suppose $G = (\mathbf{X}, E)$ is a causal graph and P is the multivariate distribution of variables \mathbf{X} . P satisfies the *global Markov condition* for G if for any three mutually disjoint subsets of \mathbf{X} , $\mathbf{A}, \mathbf{B}, \mathbf{C}$

$$\mathbf{A} \perp_d \mathbf{B} | \mathbf{C} \implies \mathbf{A} \perp \mathbf{B} | \mathbf{C}$$

◇

Proposition 4. Let $G = (\mathbf{X}, E)$ be a causal graph with an associated SEM, and P the distribution of X implied by these. If G and P satisfy assumptions 1 and 2, then the global and local Markov conditions are equivalent.

Proof. See Theorem 3.27 in [21]. □

The global Markov condition states that, under the given assumptions, “graphically separated” variables in a causal graph (in the sense of d-separation) are conditionally independent. This characterization is fruitful for two purposes, which we mention below.

In the first place, in terms of prediction, the global Markov condition lends itself to specifying, for each variable X_i , a minimal set of relevant predictor variables in \mathbf{X} . This is achieved by noting that X_i is d-separated from all other variables in \mathbf{X} by its parents, its children, and the parents of its children. Thus, conditioning on this set of variables thus makes X_i independent of all other variables. This set of variables are known as a *Markov blanket*, and can be used to define a surrogate graphical construct known as the *moral graph* of G .

Definition 23 (Markov Blanket and Moral Graph). Suppose $G = (\mathbf{X}, E)$ is a causal graph. The *Markov blanket* of a variable $X_i \in \mathbf{X}$ is defined as

$$\mathbf{MB}_i = \mathbf{PA}_i \cup \mathbf{CH}_i \cup \left(\bigcup_{X_j \in \mathbf{CH}_i} \mathbf{PA}_j \right),$$

where $\mathbf{CH}_j = \{X_k \in \mathbf{X} : (X_j, X_k) \in E\}$ is the set of children of X_j , for $j = 1, \dots, p$. The *moral graph associated to G* is the undirected graph over \mathbf{X} with edges between pairs of variables if they belong to each other’s Markov blanket. ◇

Secondly, for the task of inferring a graph structure, we can see that the mapping between graphical structure and conditional independence facilitated by d-separation can be exploited to classify overall graph structures in terms of their statistical implications. Concretely, the global Markov condition implies that two graphs with the same d-separations entail the same set of conditional independence relations among variables. Taking into account the characterization d-separation equivalence classes in Proposition 3, it follows that *two causal graphs with the same skeleton and v-structures entail the same conditional independence relations*. This observation strongly hints at the possibility of inferring a causal graph structure, at least up to its skeleton, from statistically testing conditional independence.

Faithfulness and the Identification of the Skeleton of G

The link established above between causal graph skeletons and statistical patterns suggests the possibility of inferring the former from data. However, to fully facilitate the task of graph skeleton inference, it is nonetheless necessary to make one further assumption. This is because while a causal graph’s skeleton and v-structures imply a given set of observable conditional independence relations, we cannot rule out additional “accidental” conditional independence relations that are *not* entailed by the graph’s skeleton and v-structures. Therefore, a final key assumption is *the converse* of the global Markov condition, known as faithfulness.

Assumption 3 (Faithfulness). *Given a causal graph $G = (\mathbf{X}, E)$ and an associated SEM, the probability distribution P of \mathbf{X} satisfies*

$$\mathbf{A} \perp \mathbf{B} | \mathbf{C} \implies \mathbf{A} \perp_d \mathbf{B} | \mathbf{C}$$

for any three mutually disjoint subsets of variables \mathbf{A} , \mathbf{B} , and \mathbf{C} . This property is known as faithfulness of P to G .

The significance of faithfulness can be appreciated by examining the collider motif from Figure 5-2c. In the collider from Figure 5-2, conditioning on Y is intuitively expected to induce “selection bias”, generating a spurious correlation between its unconditionally independent causes, X and Z . Faithfulness guarantees that this is the case, as it requires that X and Z be statistically dependent given Y , since X and Z are *not* d-separated by Y . However, in the absence of this assumption, it is conceivable that $X \perp Z | Y$. For instance, considering a causal SEM model over a collider motif, suppose $X \sim \text{Bin}(1, p_X)$, $Z \sim \text{Bin}(1, p_Z)$, and $Y = (X + Z + \varepsilon_Y)_{\text{mod } 2}$, with $\varepsilon_Y \sim \text{Bin}(1, p_Y)$. In this case, $X \perp Z | Y$ if and only if $p_Y = \frac{1}{2}$.

The above example illustrates why faithfulness is argued to be a relatively weak assumption in [35]. The authors assert that violations of faithfulness require, in the context of a SEM,

very specific configurations of parameters. Formally, they prove that in a linear SEM the set of coefficients that entail unfaithful distributions has Lebesgue measure zero (Theorem 3.2). However, this issue is not without controversy, as some authors maintain that violations of faithfulness may be common in real-world scenarios [54]. For example, in [55] the author argues that faithfulness may be expected to fail in systems that evolve or are designed to render some true causal relationships ineffective, such as biological systems that strive to maintain homeostatic equilibria while facing variations of environmental conditions.

Assumptions 1, 2, and 3, are sufficient to guarantee that the skeleton of causal graph is an identifiable functional of the probability distribution of observable variables in a SEM. This is because for two causal graphs G and G^* over the same set of variables \mathbf{X} , it is the case that

$$\begin{aligned}
 &G \text{ and } G^* \text{ have equal skeletons and instances of v-structures} \\
 &\iff \\
 &\text{for any mutually disjoint } \mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{X}, (\mathbf{A} \perp_d \mathbf{B} | \mathbf{C} \text{ in } G \iff \mathbf{A} \perp_d \mathbf{B} | \mathbf{C} \text{ in } G^*) \\
 &\iff \\
 &\text{for any mutually disjoint } \mathbf{A}, \mathbf{B}, \mathbf{C} \subseteq \mathbf{X}, (\mathbf{A} \perp \mathbf{B} | \mathbf{C} \text{ in } G \iff \mathbf{A} \perp \mathbf{B} | \mathbf{C} \text{ in } G^*)
 \end{aligned}$$

In the above, the first double implication is given by the characterization of d-separation equivalence classes from Proposition 3, while the second is given by the global Markov condition from Proposition 4 and the assumption of faithfulness. These implications require that causal graphs with different skeletons must necessarily correspond to different distributions of observable variables, characterized by different sets of conditional independence relations among these variables. Thus, the skeleton of a causal graph is identifiable.

Inference of the Skeleton of a Causal Graph

At this point, the key question is how to estimate a causal graph skeleton from sample data, assuming a causal process that can be represented as a SEM, and Assumptions 1, 2, and 3. The natural way forward suggested by the above considerations is to test for conditional independence relations among variables. This is the strategy employed by the PC algorithm from 4.1.3, which has been applied to GRN inference in, for example, [36]. Furthermore, in the simplest parametric case of multivariate Gaussian distributions, conditional independence is equivalent to zero partial correlation, and partial correlations can be obtained from the inverse of the covariance matrix. In this case, known as a Gaussian Graphical Model (GGM), one strategy has been to attempt estimation of Σ or Σ^{-1} [56, 57].

It is important to note that the identifiability of a parameter — in this case, the skeleton of a causal graph — does not necessarily imply its estimability [58]. Identification of the skeleton of a causal graph means that it is determined by the distribution of observable

variables. This suggests the possibility of reconstructing the skeleton *if the distribution of observable variables is known*. To this effect, Spirtes, Glymour and Scheines [35] proved that the PC algorithm correctly reconstructs the skeleton of a causal graph if “oracle information” about the true conditional independence relations among variables is available. In other words, the PC algorithm is Fisher consistent. However, the true challenge at hand is deciding on conditional independences not with “oracle knowledge” of the true distribution, but rather *using sample data*, which hinges upon the availability of adequate statistical tests.

Regarding the estimability of the skeleton of a causal graph, it is clear that if a pointwise consistent test of conditional independence is used to test dependencies for the PC algorithm, then Fisher consistency guarantees that the output skeleton pointwise-consistently estimates the true skeleton []. However, this result is not a sufficient practical assurance that the PC algorithm will estimate a graph with any degree of confidence. On one hand, Robins *et al.* [37] show that there can be no uniformly consistent estimator of the skeleton of arbitrary causal graphs. This implies that although the output of the PC algorithm may converge to a true causal graph, it may require an arbitrarily large sample size to estimate this graph with a high degree of confidence. Furthermore, recent research highlights the difficulty of obtaining a pointwise consistent test of conditional independence valid outside of highly restrictive parametric statistical models. Shah and Peters [22] show that if the only restriction to the joint probability distribution of variables is finite second moments, then if S is a continuous variable, $X \perp Y|S$ is *untestable*, meaning that no α -level test has power against any alternative.

5.2. Dynamic Models of Gene Expression

The theory of causal graphs from the previous section provides a framework to reason about generic causal processes. Consequently, our presentation of this theory does not address the specifics of modeling gene regulation. However, it is reasonable to ask how to specifically model gene regulatory processes realistically. Adequate mathematical models of gene regulation can help understand to what degree the general results from the theory of causal graphs are applicable to inferring gene regulatory networks in particular, or how methods from causal graphs or other paradigms may be refined for this specific subject domain.

An initial observation is that the dimension of time must be considered to model gene regulation realistically, as many of regulation’s salient features are processes that are fundamentally dynamic, such as feedback cycles or chemical degradation. From the point of view of causal graphs, a natural way forward would be to model relations between time-indexed variables in a dynamic causal graph. However, this approach comes with difficulties. For example, for a naive application of the PC algorithm in such a dynamic context, several observations

of multivariate *time series* would be in theory required. In the case of gene regulation, this kind of gene expression data is often not available. In this scenario, if the frequency of observations is significantly lower than the time scale of the causal relations, inference of causal networks would have to be undertaken using time series data with missing values. In an extreme situation, inference of this causal graph would have to be faced with only observations from the dynamic system's steady-state.

For the above reasons, currently available methods for inference and interpretation of causal graphs can be argued to be better suited to simple causal processes which can be considered 'static', as opposed to 'dynamic' processes such as gene regulation. In this manner, with some exceptions (for example, [59, 60]), dynamic models of gene regulatory networks are often not formulated explicitly within the framework of causal graphs. In this regard, there is ongoing research into how to productively bridge gaps between explicit frameworks of causation, such as causal graphs, and dynamic mathematical models [61, 62], [63].

In the following, we briefly present two dynamic models of gene regulation, and consider the problem of GRN inference in their contexts. The first, proposed by Ackers *et al.* [64] and further extended in [65, 66, 41, 40], models chemical concentration of mRNA and gene protein products as a system of deterministic ordinary differential equations derived from the thermodynamics of gene expression. Under several assumptions and approximations, statistical inference of the parameters of this model, and of an associated gene regulatory network, is possible using steady-state data contaminated with random noise. The second approach, by Young *et al.* [67], models gene expression as following a *VAR*(1) process. In this model, again under several strong assumptions, inference for a gene regulatory network is possible using data from the steady-state distribution by exploiting a relationship between long and short run covariance matrices.

Thermodynamic ODE model

In the model of gene expression of [64], for each gene i the following differential equations govern the quantities of mRNA and protein products inside a cell at a given point in time, denoted by x_i and y_i respectively:

$$\begin{aligned}\frac{dx_i}{dt} &= \tau_i f_i(y) - \lambda_i^{\text{RNA}} x_i \\ \frac{dy_i}{dt} &= r_i x_i - \lambda_i^{\text{Protein}} y_i\end{aligned}\tag{5-1}$$

In the above, the coefficients λ represent instantaneous degradation rates of the respective chemical compounds, while τ_i is the rate of RNA transcription when RNA-polymerase is bound to the promoter region of gene i , and $f_i(y)$ models the probability that this is the case. Additionally, protein gene product y_i is assumed to be translated at a fixed rate r_i of

available mRNA x_i . Together, these equations form a simplified but realistic model of the changing concentrations of mRNA and protein products of gene expression.

The function $f_i(y)$ relates protein gene products, y , and the presence of RNA-polymerase at the promoter region of gene i . Using a statistical physics model of this process, the authors in [65] derive the following functional form for $f_i(y)$:

$$f_i(y) = \frac{b_{i0} + \sum_{S \subseteq \mathbf{y}} b_{iS} \prod_{y_k \in S} y_k}{c_{i0} + \sum_{S \subseteq \mathbf{y}} c_{iS} \prod_{y_k \in S} y_k} \quad (5-2)$$

As $0 \leq f_i(y) \leq 1$ for any y , it is imposed that $c_{iS} \geq b_{iS} \geq 0$. Also, to normalize these coefficients, it is assumed that $c_{i0} = 1$. In this expression, the sets S such that $b_{iS} > 0$ represent sets of genes whose protein products combine to form a regulator of gene i . It is assumed that the quantities of these regulators are proportional to the product of the quantities of protein gene products that combine to form them.

This model structure accommodates a rich variety of motifs in gene regulation, including cooperative and synergistic repression and activation [68]. Furthermore, it lends itself to naturally defining a gene regulatory network that summarizes, in a simplified manner, the flow of causal influence among genes. This GRN (\mathbf{x}, E) is given by

$$(x_j, x_i) \in E \iff y_j \in S \subseteq \mathbf{y} \text{ for some } S \text{ such that } c_{iS} > 0. \quad (5-3)$$

Under this definition, a gene regulatory network associated to this model can be recovered from estimates of the parameters b and c . Meister *et al.* consider this problem. In [66], the authors characterize the steady state of this model, showing that the coefficients b and c and steady-state observations of x jointly satisfy linear constraints. Thus, assuming randomly perturbed steady-state measurements of x , and restricting the order of polynomials in the functions $f_i(y)$, they propose using $L1$ -regularized regression (LASSO) to approximately estimate non-zero coefficients b_{iS} , c_{iS} .

A noteworthy alternative to the functional form of $f_i(y)$ given by Marbach *et al.* [40] is presented below. This functional form has been used to generate data for the DREAM network inference challenges, and is used in the data generating mechanism from the widely used GeneNetWeaver application [41]:

$$f_i(y) = \frac{b_{i0} + \sum_{S \subseteq \mathbf{y}} b_{iS} \prod_{y_k \in S} \left(\frac{y_k}{p_{ik}} \right)^{\eta_{ik}}}{1 + \sum_{S \subseteq \mathbf{y}} \prod_{y_k \in S} \left(\frac{y_k}{p_{ik}} \right)^{\eta_{ik}}}$$

In the above, it is assumed that the coefficients b_{i0} and b_{iS} are non-negative and less than 1, and denote relative activation of gene i when the transcription factor associated to S is bound to gene i 's promoter region. The terms p_{ik} are dissociation constants which mark half-maximal concentrations of y_k , while the parameters η_{ik} , known as Hill coefficients, control the overall sensitivity of $f_i(y)$ to gene product y_k . These additional parameters and the non-linear functional form they give rise to make inference more difficult in comparison to the case considered in [66].

Linear Dynamic Model

An alternative approach, laid out by Young *et al.* [67], also attempts to recover a dynamic gene regulatory network from steady state observations. The authors consider, in a less biophysically inspired model, a vector-valued time series of gene expression levels as following a $VAR(1)$ process:

$$\begin{aligned}\mathbf{X}_0 &= \boldsymbol{\varepsilon}_0 \\ \mathbf{X}_t &= \mathbf{A}\mathbf{X}_{t-1} + \boldsymbol{\varepsilon}_t \text{ for } t \geq 1\end{aligned}$$

In this equation, \mathbf{A} is an adjacency matrix for the a ‘‘period-to-period’’ causal graph that contains relations between \mathbf{X}_{t-1} and \mathbf{X}_t . Furthermore, the components of $\boldsymbol{\varepsilon}_t$ are assumed to be mutually independent of each other, and $\boldsymbol{\varepsilon}_t$ is assumed independent of $\boldsymbol{\varepsilon}_s$ for $s \neq t$ (exogeneity). Thus, in the terms of Section 5.1, this model can be viewed as a dynamic causal graph in which all edges link variables in consecutive time periods.

The authors consider estimation of A using steady-state data. Under the assumption that $\boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \mathbf{D})$, with \mathbf{D} a diagonal matrix, it is the case that if the eigenvalues of \mathbf{A} are less than 1 in absolute value, then \mathbf{X}_t converges in distribution to $N(0, \Sigma)$, with $\Sigma = \sum_{i=0}^{\infty} \mathbf{A}^i \mathbf{D} (\mathbf{A}^\top)^i$. In this manner, the problem is posed as finding conditions for the identification and inference of A using samples of $N(0, \Sigma)$.

In [67], the authors advance a necessary condition for the identifiability of A in terms of sample size and number of genes considered. However, to obtain sufficient conditions, they impose the following additional assumptions:

- \mathbf{D} is known.
- \mathbf{A} can be row and column permuted to be lower triangular - which implies that the causal process is acyclical, such that there is no mutual causation among gene expressions, neither contemporaneously nor as time unfolds.

The strict conditions in this model then allow for the identifiability and inference of \mathbf{A} using steady-state data. However, perhaps most importantly, these results suggest the general

difficulty of inferring gene regulatory networks from a presumably stochastic gene expression process, using measurements of gene expression that are not time series but are instead assumed independent and identically distributed. This observation therefore raises questions regarding the applicability of gene regulatory network inference algorithms reviewed in Section 4.1.

6. Methods

In this chapter we outline the methods we use to conduct an assessment of a selection of inference algorithms as estimators for GRNs in probabilistic models of gene regulation and expression. In Section 6.1, we describe the specific statistical models that we assume as generators of gene expression data, from which graph-valued parameters are taken as the “ground truth” GRNs to be estimated. Then, in Section 6.2, we discuss how we make a simple theoretical characterization of the asymptotic bias of Relevance Networks under the given statistical models. Finally, in Section 6.3, we describe the setup of a simulation study to evaluate the performance of the selected algorithms in finite sample scenarios.

6.1. Statistical Models for Gene Expression Data

We analyze the performance of GRN inference algorithms as applied to data arising from two types of probabilistic models reviewed in Chapter 5: causal graphical models, and a version of the thermodynamic ODE-based model of [65] with random variation. For the former case, we consider the following causal graphical models $G = (\mathbf{X}, E)$ 5.1, in decreasing order of generality:

- Model 1: A general causal graphical model with a general associated SEM, given by equations $X_i = f_i(\mathbf{PA}_i, \varepsilon_i)$, with arbitrary functions f_i and mean-zero error terms ε_i that satisfy common regularity conditions (for example, finite second moments).
- Model 2: A causal graphical model with an associated SEM given by equations

$$X_i = (F_i^{-1} \circ \Phi) \left(\sum_{X_j \in \mathbf{PA}_i} a_{ij} (\Phi^{-1} \circ F_j)(X_j) + \varepsilon_i \right),$$

where Φ and Φ^{-1} are the distribution and quantile functions of a standard normal random variable, F_i and F_i^{-1} are arbitrary pairs of distribution and corresponding quantile functions, and $\varepsilon_i \sim N(0, \sigma_i^2)$. The matrix $A = (a_{ij})$ can be thought of as the weighted adjacency matrix of the causal graph.

We note that the surrogate variables defined as $Y_i = (\Phi^{-1} \circ F_i)(X_i)$ follow a multivariate Gaussian distribution with covariance matrix $\Sigma = BDB^\top$, where D is the diagonal matrix containing the variances σ_i^2 and $B = (I - A)^{-1}$. The corresponding correlation matrix is given by $R = P\Sigma P$, where $P = \text{Diag}(\Sigma)^{-\frac{1}{2}}$. We note that if $\Sigma = R$, then the marginal distribution function of each variable X_i is F_i .

Noting that the variables X_i follow a Gaussian copula with correlation matrix R , we name this model the *Gaussian Copula Structural Equations Model*, following [69].

- Model 3: The *linear Gaussian Structural Equations Model*, defined by setting $F_i = \Phi$ for all variables i in Model 2.

For these models, the target “ground truth” GRN to be estimated is simply the causal graph. We assume throughout our study causal graphical models that satisfy the assumptions from Section 5.1. We note that for Models 2 and 3, the zero-pattern of matrix A reflects the GRN skeleton to be estimated, while the zero-pattern of $\Sigma = BDB^\top$ is equivalent to the zero-pattern of the mutual information matrix of gene expressions. Without loss of generality, A is taken to be lower triangular.

As for the ODE-based model of gene expression, initially we consider a simplified version of [65], common in the literature (for example, in evaluations in [66, 49]). where gene product concentrations are represented by a single variable per gene, instead of two distinct variables that track RNA and protein concentrations separately. Concretely, this model amounts to the following modification of Equation 5-1:

$$\begin{aligned}\frac{dx_i}{dt} &= \tau_i f_i(y) - \lambda_i^{\text{RNA}} x_i \\ x_i &= y_i\end{aligned}$$

We use the expression for $f_i(y)$ given in Equation 5-2. For this model, the target “ground truth” GRN to be estimated is defined as presented in Equation 5-3.

Given the above, to induce random variation in observed gene expression values, we then consider a generalization of this model as a stochastic process. Following [41, 66, 49], we recast the differential equation governing gene product concentration as a chemical Langevin equation, with a forcing process given by Gaussian white noise stochastic processes. In general, this can be carried out for differential equations $\frac{du}{dt} = V(u) - D(u)$, where $V(u)$ and $D(u)$ are production and degradation rates, respectively. Then, the stochastic version we consider is

$$\frac{du}{dt} = V(u) - D(u) + k \left[\sqrt{V(u)} \eta_{Vt} + \sqrt{D(u)} \eta_{Dt} \right], \quad (6-1)$$

where η_{Vt} and η_{Dt} are independent white noise Gaussian processes, and k is a constant that controls noisiness.

6.2. Methods: Theoretical Analysis of Relevance Networks

For Relevance Networks, we give a numerical sense of how plausible it is that they produce asymptotically accurate estimates of a causal graphical model's skeleton. This is done by assuming a Gaussian Copula SEM causal model (Model 2 above), and assuming that the Relevance Network estimator is thresholded in such a way that it asymptotically recovers the zero-pattern of the mutual information matrix.

This analysis is based on the observation that in the context of a causal graph $G = (\mathbf{X}, E)$ that follows a Gaussian Copula SEM (Model 2), the asymptotic consistency of the Relevance Network estimator is determined by whether, for the lower triangular weighted adjacency matrix A and the covariance matrix of latent variables $\Sigma = BDB^\top$ with $B = (I - A)^{-1}$,

$$\Sigma_{ij} = \Sigma_{ji} = 0 \iff A_{ij} = A_{ji} = 0. \quad (6-2)$$

Theorem 1 from [25] states necessary and sufficient conditions for Condition 6-2 to hold, assuming all components of the diagonal of D are positive. These are as follows.

1. The skeleton of G , \widehat{G} , is *homogeneous*, which means that neither of the following graphs are motifs (induced subgraphs) in \widehat{G} :
 - the 4-chain $A_4 = (V, E)$, given by $V = \{a, b, c, d\}$ and $E = \{\{a, b\}, \{b, c\}, \{c, d\}\}$ (see Figure 6-1a)
 - the 4-cycle $C_4 = (V, E)$, given by $V = \{a, b, c, d\}$ and $E = \{\{a, b\}, \{b, c\}, \{c, d\}, \{a, d\}\}$ (see Figure 6-1b).
2. G follows a *Hasse perfect vertex elimination order*. This holds when, for any two adjacent nodes X_i, X_j in G ,

$$adj(X_j, G) \cup \{X_j\} \subsetneq adj(X_i, G) \cup \{X_i\} \implies (X_i, X_j) \in V.$$

Thus, to characterize the plausibility of mistakes of Relevance Networks, we characterize the frequency of homogeneous-skeleton directed acyclic graphs that follow a Hasse perfect vertex elimination order. First, to characterize the frequency of homogeneous skeletons for graphs of up to ten nodes, we directly count the numbers of unlabelled connected homogeneous graph skeletons, and compare them to the total numbers of connected unlabelled skeletons. Also, for a more general overview, we give an argument based on the Erdos-Renyi random graph model to support the notion that homogeneous graphs are extremely rare. Secondly,

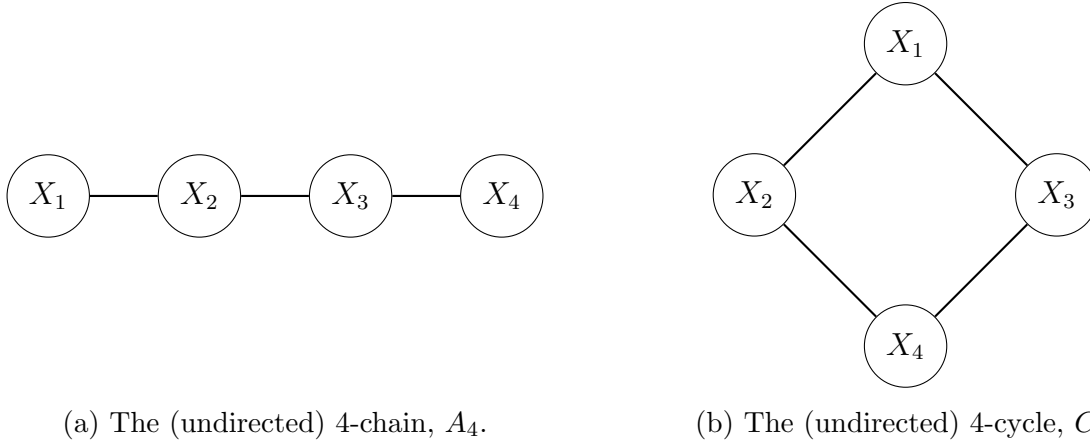


Figure 6-1.: A_4 and C_4 . A homogeneous graph does not have these subgraphs as motifs.

in terms of orientations, we provide some propositions and numerical examples that show that Hasse perfect vertex elimination orders are rare among all possible orientations of labelled and unlabelled homogeneous undirected graphs.

Moreover, to characterize the numbers of mistakes made by Relevance Networks on arbitrary graph topologies, we compute the discrepancies between A and $\Sigma = BDB^\top$ for large samples of connected directed acyclic graphs of up to ten nodes.

6.3. Methods: Simulation Study

For a more detailed appraisal of GRN inference algorithms as estimators in statistical models, especially in finite sample settings, we conduct a simulation study. For this, we apply a selection of GRN inference algorithms to data arising from several statistical models for gene expression based on different GRN graph topologies. We then compare algorithm outputs to the “ground truth” GRNs through several metrics. In the following, we outline the specification of each component in this procedure.

6.3.1. Inference Algorithms

We evaluate the performance of GRN inference algorithms reviewed in Chapter 4 in the R statistical software [70]. Most algorithms are available as functions in widely distributed packages for the statistical software R Project. The only exception is NARROMI, for which we provide an R implementation adapted from the original code MATLAB by its creators (Table 6-1).

When applying our selected GRN inference algorithms and seeking a fair comparison of their

outputs, we are faced with the problem of how to choose values for the diverse parameters they rely on. On one hand, with the objective of mimicking naive uses of these algorithms, we apply them using parameter values that can be considered “default”. However, in some cases we also entertain more sophisticated approaches to parameter specification. Concretely, we apply the following estimators with corresponding parameter specifications:

- **Relevance Networks:** The output is an estimated mutual information matrix. Each component is calculated by aggregating Miller-Madow entropy estimators on discretized data. Discretization of data is done by splitting the sample of N observations into $N^{1/3}$ equal-width bins, as per the default in `fastGeneMI`, and according to the recommendation in [45]. Also, bootstrapped critical values to test null mutual information at 95% confidence are calculated based on 500 random shuffles of the data.
- **ARACNe:** Beginning with the estimated mutual information matrix from Relevance Networks, edges are pruned according to the original statement of ARACNe in [26], discussed in 2. This requires setting `eps = 1e-10` in the corresponding function from package `parmigene`.
- **CLR:** No parameters are required. The input is the mutual information matrix from Relevance Networks.
- **MRNET:** No parameters are required. The input is the mutual information matrix from Relevance Networks.
- **NARROMI (defaults):** We apply NARROMI with default parameters from the original code. These are: $L1$ -penalty weight $\lambda = 1$; threshold for coefficients in iterative LAD-LASSO estimation $\gamma = 0.05$; weight of regression-based score in adjacency matrix $\phi = 0.6$; threshold to discretize edges $\tau = 0.05$.
- **NARROMI (c.v. λ):** We also apply NARROMI using cross validation to select the value for λ . This is done through fitting LAD-LASSO with $\lambda = 0.01, 0.1, 0.5, 1, 2, 4$ on 5 random 80%-20% train-test splits of sample data, and selecting the value corresponding to the lowest empirical mean absolute error. We also set $\gamma = 0$, in the spirit of the original formulation of the algorithm.
- **GENIE3:** This algorithm is fit using random forests as the regression tree ensemble algorithm. Each forest is fit with 1000 trees, and each sample split in each node of each tree is done considering a random subset of $(p - 1)^{\frac{1}{2}}$ variables (of the full $p - 1$ variables). All of these are the defaults in package `GENIE3`.
- **TIGRESS:** The algorithm is fit using defaults from the implementation by its authors. These are: number of LARS steps $L = 5$; number of stability selection replicates $R = 100$; parameter for random noise in stability selection $\alpha = 0.2$.

- **PC-Stable (Gaussian MI)**: We apply the PC-Stable algorithm using a parametric conditional independence developed for Gaussian data [56]. This test is based on a partial correlation estimator derived from a shrunk estimator of the covariance matrix. For data sets of 100 observations or less, each test’s nominal type I error rate is set at $\alpha = 0.05$. For larger sample sizes, we set $\alpha = 0.01$, and we additionally restrict the algorithm to only consider conditioning sets of up to three variables, in order to prevent exploding runtimes.
- **PC-Stable (discretized data)**: We also apply the PC-Stable algorithm using a semi-parametric conditional independence test on discretized data proposed in [71]. This test follows an approach intermediate between an asymptotic Chi-squared test for conditional independence and a test based on permutations. To discretize data, we use the same procedure as for Relevance Networks. Specification of α and maximum conditioning sets is the same as for the previous case.

Table 6-1.: Implementations of Algorithms Tested

Algorithm	R Package (version)	Available on
Relevance Network (MI Estimator)	<code>fastGeneMI</code> (1.0) [72]	Bitbucket repository
ARACNe	<code>parmigene</code> (1.0.2) [73]	CRAN
CLR	<code>parmigene</code> (1.0.2) [73]	CRAN
MRNET	<code>minet</code> (3.44.1) [74]	Bioconductor
NARROMI	Own implementation	Adapted from MATLAB code available in [75]
TIGRESS	<code>tigress</code> (0.1.0) [30]	GitHub repository
GENIE3	<code>GENIE3</code> (1.8.0) [32]	Bioconductor
PC-Stable	<code>bnlearn</code> (4.5) [76]	CRAN

6.3.2. Models for Simulation: Causal Graphical Models with Gaussian Copula SEMs

As a simple baseline, we initially study the inference of GRNs with gene expression generated by causal graphical models associated to a Gaussian Copula SEMs.

GRN Topologies

Simulations are conducted for five graph structures of gene regulatory networks compiled from the relevant literature. First, we use the “Curated Models” used in the GRN evaluations

in [49], which consist of four GRNs that have been posited in several studies, based on syntheses of experimental results. These are:

1. **GSD**: a model for gonadal sex determination, in [77].
2. **HSC**: a model for hematopoietic stem cell differentiation, from [78].
3. **VSC**: a model for the regulatory relations relevant to ventral spinal cord development, from [79].
4. **mCAD**: a model of gene interactions for mammalian cortical area development, proposed in [80].

Second, to analyze performance of our selected algorithms on a larger graph, we include a network given by a 50-node induced subset of a curated directed and acyclic GRN for *Saccharomyces cerevisiae*, published in [47].

5. **SC50**: a subset of a genome-wide of *Saccharomyces cerevisiae*.

Both types of models require some preprocessing. As for the “Curated Models” from [49], the original networks include multiple edges (more than one edge between a pair of nodes), self-loops, and cycles. Thus, we simplify these graphs to make them compatible with data generating mechanisms given by SEMs of causal graphical models. For this purpose, we collapse multiple edges and remove self-loops, to initially obtain an undirected simple graph. Then, when possible, to obtain a directed acyclic graph we retain the orientations of the original edges and choose orientations that do not result in cycles for the collapsed multiple edges. If this procedure inevitably produces cycles, we then choose a random topological order of nodes, and orient edges accordingly, also producing a directed acyclic graph.

For the SC50 network, we select the 50 gene subset with the goal of preserving topological characteristics of the whole network. This is done following the module extraction algorithm proposed in [81]. This procedure was used to generate the benchmark networks for DREAM challenges 4 and 5, and is a part of the *GeneNetWeaver* application.

Visualizations of the final graph topologies considered are pictured in Figure in **6-2**. Table **6-2** shows a summary of some of their topological features.

Data Generating Models

For each graph topology listed above, we simulate 1000 replicates of data sets arising from a Gaussian Copula SEM with the following characteristics:

Table 6-2.: Topological features of GRNs for simulation study.

Graph	# nodes	# edges	# edges (moral graph)	Freq. of (dir.) collider	Freq. of (undir.) triangle	Freq. of (undir.) 4-chain	Freq. of (undir.) 4-cycle
GSD	19	58	75	26	63	331	39
HSC	11	20	23	3	10	29	3
mCAD	5	9	9	0	7	0	0
VSC	8	10	12	2	3	7	0
SC50	50	52	53	4	0	212	6

- **Marginal distributions:** In each model specification, all gene expressions X_i are assumed to follow one of the following distributions:

1. Standard Gaussian
2. Laplacian, with scale parameter = $2^{-1/2}$, calibrated to produce unit variance. The Laplacian distribution has excess kurtosis equal to 3.
3. Pareto, with location parameter = 13 and scale parameter = 15. These parameters imply approximately unit variance and excess kurtosis ≈ 10 .

These options thus capture both the baseline Gaussian case and deviations from it in the form of heavier-tailed distributions.

- **Calibration of A :** Coefficients of the weighted adjacency matrices are calibrated to achieve “noise-to-signal” ratios of surrogate variables $Y_i = (\Phi^{-1} \circ F_i)(X_i)$, $r = \frac{Var(\varepsilon_i)}{Var(Y_i)}$ of

1. $r = 0.35$ (low noise)
2. $r = 0.65$ (high noise)

For simplicity, all parents of a given gene are given equal weights in absolute value.

- **Sample sizes:** Data sets of 20, 50, 100, and 500 observations are simulated.

All gene expression profiles of all data sets are standardized before being supplied to the inference algorithms.

6.3.3. Models for Simulation: Causal Graphical Model with Non-Linear SEM

In order to assess the sensitivity of inference algorithms to the assumption of a Gaussian copula modeling the dependence structure of gene expressions, we consider a causal graphical model with a highly non-linear SEM.

GRN Topologies

For these experiments, we use the same graph topologies as the for the Gaussian Copula SEM simulations, except for the SC50 network.

Data Generating Model

For each network topology, we simulate one SEM with the following functional form for each equation:

$$X_i = \cos \left(\sum_{X_j \in \mathbf{PA}_i} a_{ij} X_j \right) + \varepsilon_i, \quad \text{with } \varepsilon_i \sim N(0, \sigma_i^2)$$

The cosine transformation is chosen not on the grounds of realistically depicting gene expression, but rather as a device to induce significant non-linearity and, in some cases, appreciable non-monotonicity. Parameter values a_{ij} and σ_i^2 are calibrated for this purpose. Figure **6-3** shows the induced relation between a selection of gene expressions X_i in the GSD topology and their parents, represented by $\sum_{X_j \in \mathbf{PA}_i} a_{ij} X_j$. Again, data sets of 20, 50, 100, and 500 observations are simulated, data sets are standardized before being supplied to the GRN inference algorithm, and 1000 replications are carried out for each sample size.

6.3.4. Models for Simulation: Thermodynamic ODE Model of Gene Expression

A third type of data generating mechanism for which we simulate is given by a phenomenological model of gene expressions based on stochastic differential equations, as described in the first section of this chapter. This allows us to assess GRN inference algorithm performance under a presumably more realistic description of gene expression and regulation.

GRN Topology

For these simulations, we use the VSC GRN topology from [79], also used for previously described simulations. However, for the present simulations we include cycles that are present in the original network.

Data Generating Model

To specify the data generating mechanism for these simulations we follow the description provided in [79] of the nature of regulatory relations in the VSC model. The authors present their model as a Boolean network in which each gene's level of expression is in one of two possible states at any given point in time. Also, they propose that all regulatory relations in the VSC network are repressive. Thus, for a given gene X_i , they model its level of expression

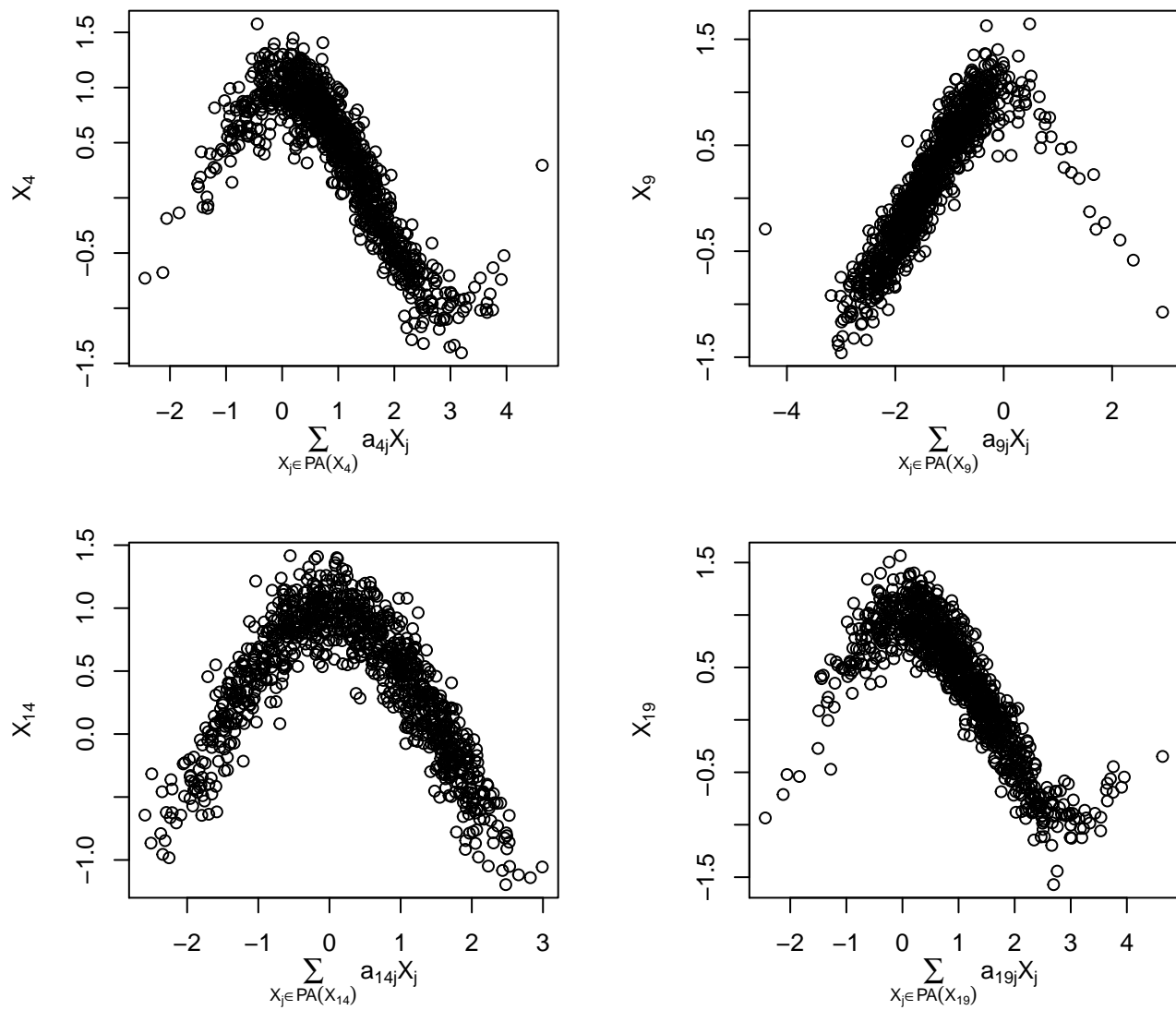


Figure 6-3.: Induced statistical dependences in non-linear SEM.

as the Boolean function which is equal to **False** if $X_j = \text{True}$ for any regulator X_j of X_i , and is equal to **True** otherwise. We translate this to our ODE-based framework by setting the parameters in Equation 5-2 for all genes to the values

- $b_{i0} = 1$,
- $c_{i0} = 1$,
- $b_{iS} = 0$ for all $S \subseteq y$, and
- $c_{iS} = 2$ for all singletons $S \subseteq y$ which contain a repressor of X_i .

Furthermore, the degradation rate of gene products is fixed at $\lambda^{\text{RNA}} = 0.1$. Finally, for the parameter k in Equation 6-1 which controls the randomness of gene expressions, we simulate data sets with the values of 0.1 and 0.2. Altogether, these parameters seem to entail non-zero stationary state gene expression and moderate levels of variability. Figure 6-4 presents sample paths generated by this model.

For this model, we simulate data sets with $n = 20, 50, 100$, and 200 observations. The initial values of gene expressions are chosen randomly between 0 and 10 with a uniform distribution. All observations included in the data sets are sampled at regular intervals of a simulated time series of gene expression after a 20 time period burn-in, such that they mostly reflect the dynamical system’s steady state.

Once again, 1000 replicates are generated for each type of simulation, and data sets are standardized before being supplied to GRN inference algorithms.

6.3.5. Evaluation Metrics

In general, our algorithms output estimated signed and weighted adjacency matrices that are not necessarily symmetric nor asymmetric. Also, in some algorithms there is no built-in restriction or specific thresholding procedure to discretize this matrix. These features pose the problem of how to adequately compare algorithm outputs to underlying GRNs, whose adjacency matrices are discrete, and to each other. We make several choices to carry out these comparisons, outlined below.

First, although some of the selected inference algorithms output signed edge weights and include heuristics to orient estimated edges, we decide to only analyze the unsigned and undirected GRN skeleton estimates they imply. When an inference algorithm outputs a non-symmetric and weighted estimated adjacency matrix \hat{A} , we instead use a “symmetrized” and unsigned GRN skeleton estimate \hat{A}^* specified by

$$\hat{A}_{ij}^* = \max \left\{ |\hat{A}_{ij}|, |\hat{A}_{ji}| \right\}.$$

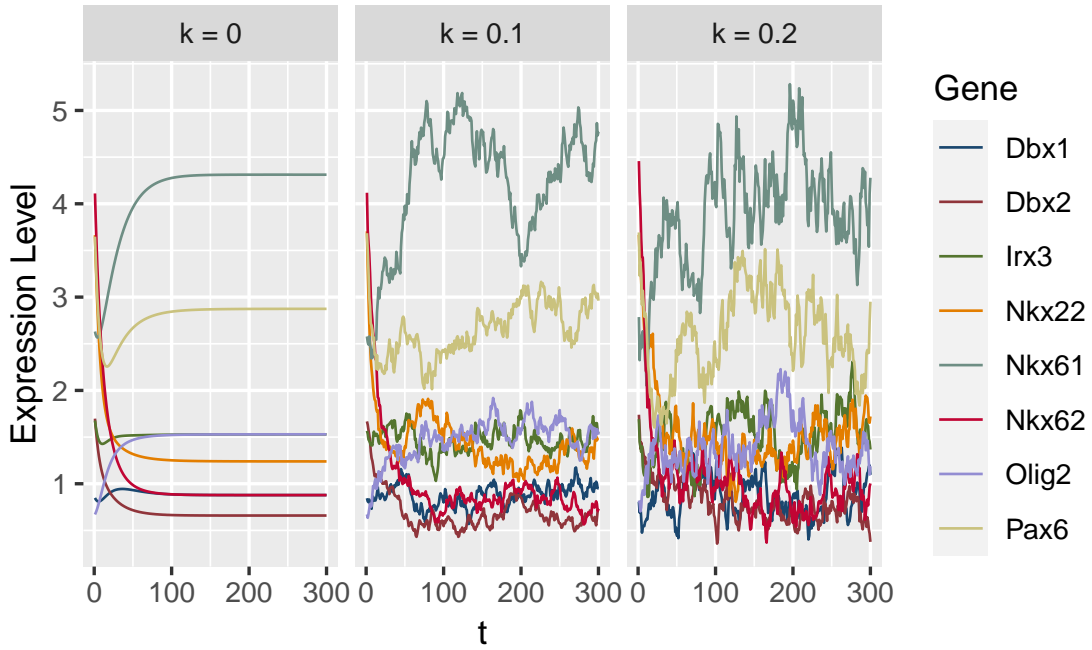


Figure 6-4.: Sample time series from Thermodynamic ODE model of gene expression.

This procedure gives each possible undirected edge its best chance of being included in a discretized skeleton estimate after thresholding.

Given an estimated undirected weighted adjacency matrix \hat{A}^* for a GRN skeleton, we employ several different metrics to assess algorithm performance in recovering the true skeleton adjacency matrix, A . For this purpose, we also consider the discretized skeleton estimate corresponding to a threshold τ , which we denote by $A^*(\tau)$.

- **Default:** When an inference algorithm includes a built-in discretization procedure (for example, ARACNe), or a natural threshold τ^* (for example, critical levels for tests of zero mutual information in Relevance Networks), we compare this natural discretized skeleton estimate to the true GRN skeleton through standard metrics of False Positive Rate (FPR), False Negative (FNR), and Accuracy (ACC):

$$FPR(\hat{A}^*) = \frac{|\{(i,j): A_{ij}=0 \text{ and } \hat{A}_{ij}^* \neq 0\}|}{|\{(i,j): \hat{A}_{ij}^* \neq 0\}|}$$

$$FNR(\hat{A}^*) = \frac{|\{(i,j): A_{ij} \neq 0 \text{ and } \hat{A}_{ij}^* = 0\}|}{|\{(i,j): A_{ij} \neq 0\}|}$$

$$ACC(\hat{A}^*) = \frac{|\{(i,j): (A_{ij}=0 \text{ and } \hat{A}_{ij}^*=0) \text{ or } (A_{ij} \neq 0 \text{ and } \hat{A}_{ij}^* \neq 0)\}|}{p(p-1)}$$

- **Top-m = |E| edges:** For a second set of evaluation metrics, we assume that the

number of edges in the undirected GRN skeleton is known to be m , and select the threshold τ_m consistent with a discretized skeleton estimate with at most m edges. We then measure $FPR\left(\hat{A}^*(\tau_m)\right)$, $FNR\left(\hat{A}^*(\tau_m)\right)$, and $ACC\left(\hat{A}^*(\tau_m)\right)$.

- **AUROC**: Finally, we measure the area under the receiver operating characteristic curve (*AUROC*). This is the area under the parametric curve

$$\left(FPR\left(\hat{A}^*(\tau)\right), 1 - FNR\left(\hat{A}^*(\tau)\right)\right),$$

with τ varying in $[0, \infty)$. The *AUROC* can be interpreted as the probability of a randomly selected edge $\{X_i, X_j\}$ in the true GRN skeleton and a randomly selected pair of non-adjacent nodes $\{X_k, X_l\}$ satisfying $\hat{A}_{ij}^* > \hat{A}_{kl}^*$. This metric allows us to evaluate algorithm outputs while remaining agnostic about the single correct threshold for discretization.

To assess the sensitivity of algorithms to the randomness in data, we consider not only measures of location of the metrics considered, but also measures of variability. We note that the optimal values of these performance metrics (those which could in theory be achieved with perfect knowledge of the underlying data generating model) are equal to zero for *FPR* and *FNR*, and to one for *ACC* and *AUROC*. On the contrary, for all algorithms, a value of 0.5 is achieved on average by randomly guessing the presence of each possible edge in the graph.

We use the following additional benchmarks for results from simulations of causal graphical models:

- **Moral graph approximation**: the performance metrics obtained by approximating a GRN skeleton with its moral graph. By its definition, a moral graph differs from its corresponding causal graph skeleton by only the edges it places between non-adjacent parent nodes with children nodes in common.
- **MI matrix approximation**: for simulations of Gaussian Copula SEMs, we also include the performance metrics obtained by approximating a GRN skeleton with the zero-pattern of the mutual information matrix of variables, which is equivalent to the zero-pattern of $\Sigma = BDB^\top$.

7. Results

To assess certain aspects of GRN inference algorithms discussed in previous chapters, we present a theoretical discussion on asymptotic biases affecting their performance, and we conduct a simulation study of their behavior with finite samples. First, for a theoretical appraisal, in Section 7.1 we characterize the prevalence of mistakes of Relevance Networks when asymptotically estimating a causal graph associated to a Gaussian Copula SEM. Then, in Section 7.2 we show the key results of the simulation study described in Chapter 6.

7.1. Theoretical Observations for Relevance Networks

Recalling Section 6.2, here we seek to give a sense of how plausible it is that an arbitrary GRN modeled by a Gaussian Copula SEM satisfies the conditions of Theorem 1 from [25]. In the following, we use on the comprehensive repository of simple graphs from [82] whenever counting or sampling graph skeletons.

Regarding the first condition, which refers to the homogeneity of a graph skeleton, Figure 7-1 contrasts the number of unlabelled (non-isomorphic)¹ connected homogeneous graphs to that of all unlabelled connected graphs of up to 10 nodes. Clearly, the proportion of unlabelled homogeneous graphs becomes minuscule quickly as the number of nodes increases.

On the other hand, when considering all labelled undirected graphs with k nodes – that is, treating isomorphic graphs as distinct – then the plausibility of observing homogeneous graphs over k nodes can be conceptualized as the probability of observing a homogeneous realization of the Erdos-Renyi random graph model $G(k, p)$ with $p = \frac{1}{2}$. It can be verified that for any fixed p , this probability will increasingly small as the number of nodes grows.

Proposition 5. *Let $\{G(k, p)\}_{k \in \mathbb{N}}$ be the sequence of Erdos-Renyi random graphs with edge probability p over k nodes. The probability of obtaining a homogeneous realization of $G(k, p)$, denoted by $P_k(H)$, is $O(k^{-1})$ as $k \rightarrow \infty$, and consequently $\text{Lim}_{k \rightarrow \infty} P_k(H) = 0$.*

Proof. See appendix. □

¹Two unlabelled graphs are distinct if there is no isomorphism between them. In other words, unlabelled graphs can be defined as equivalence classes of graphs that are isomorphic to each other.

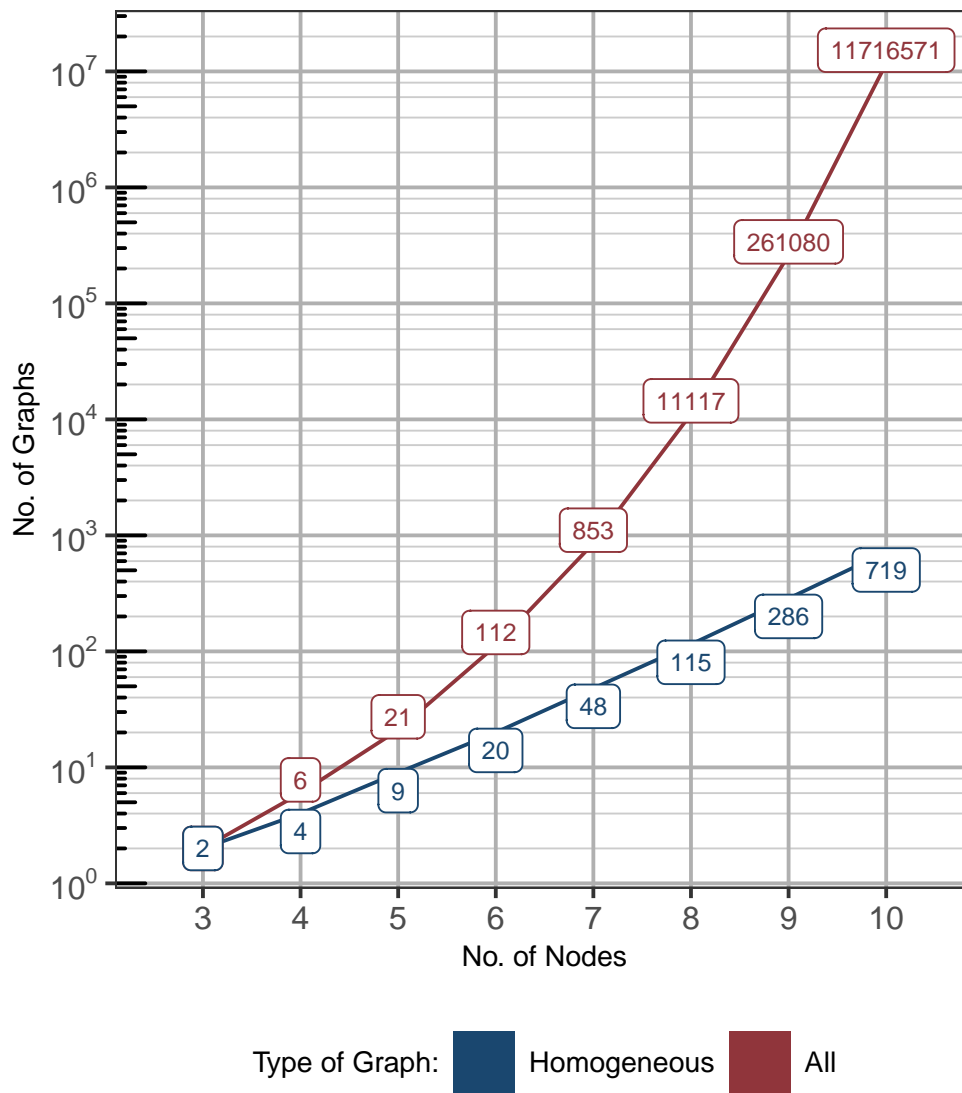


Figure 7-1.: Number of unlabelled connected undirected graphs.

Against the characterization implied by Proposition 5, it can be argued in favor of Relevance Networks that not all possible undirected graphs of a given sparsity are equally likely to be skeletons of GRNs, regardless of the number of nodes. Slightly more realistically, the sparsity of observed graphs could be expected to increase with the number of nodes, instead of remaining fixed at $1 - p$. To model this situation, consider the Erdos-Renyi random graph $G(k, k^{-\alpha})$, with $\alpha > 0$. In this setting, it can be proved again that homogeneous graphs are rare for sufficiently sparse random graphs.

Proposition 6. *Let $\{G(k, k^{-\alpha})\}_{k \in \mathbb{N}}$ be the sequence of Erdos-Renyi random graphs with edge probabilities $k^{-\alpha}$, with $\alpha > 0$, and let $P_k(H)$ be the probability of obtaining a homogeneous graph from $G(k, k^{-\alpha})$. Then,*

$$\alpha > \frac{4}{3} \implies \lim_{k \rightarrow \infty} P_k(H) = 0.$$

Proof. See appendix. □

Although random Erdos-Renyi graphs may not accurately reflect the topological features of GRNs, these results nonetheless illustrate that strict conditions on the sparsity of the underlying GRN are needed for the asymptotic correctness of Relevance Networks. Barring these conditions, the underlying GRN cannot be expected to be homogeneous, and Relevance Networks will make systematic mistakes in inferring them, even asymptotically.

Concerning the second condition of Theorem 1 from [25], we find that even after assuming a homogeneous GRN skeleton and the data generating mechanism of a Gaussian Copula SEM (Model 2), Relevance Networks will make mistakes in recovering a GRN for many orientations of edges. Figures 7-2 and 7-3 compare the total numbers of labelled and unlabelled acyclic orientations of homogeneous connected graphs of up to 10 nodes with the numbers of such orientations that follow Hasse perfect vertex elimination orders. These figures are based on Propositions 7 and 8 below.

Proposition 7. *Suppose $G = (V, E)$ is an undirected graph. The total number of labelled acyclic orientations of G is given by*

$$O_l(G) = (-1)^{|V|} \chi_G(-1),$$

where $\chi_G(-1)$ is the chromatic polynomial of G . Furthermore, the number of unlabelled (non-isomorphic) acyclic orientations of G is given by

$$O_u(G) = \frac{1}{|Aut(G)|} \sum_{A \in Aut(G)} (-1)^{|V_A|} \chi_{G_A}(-1),$$

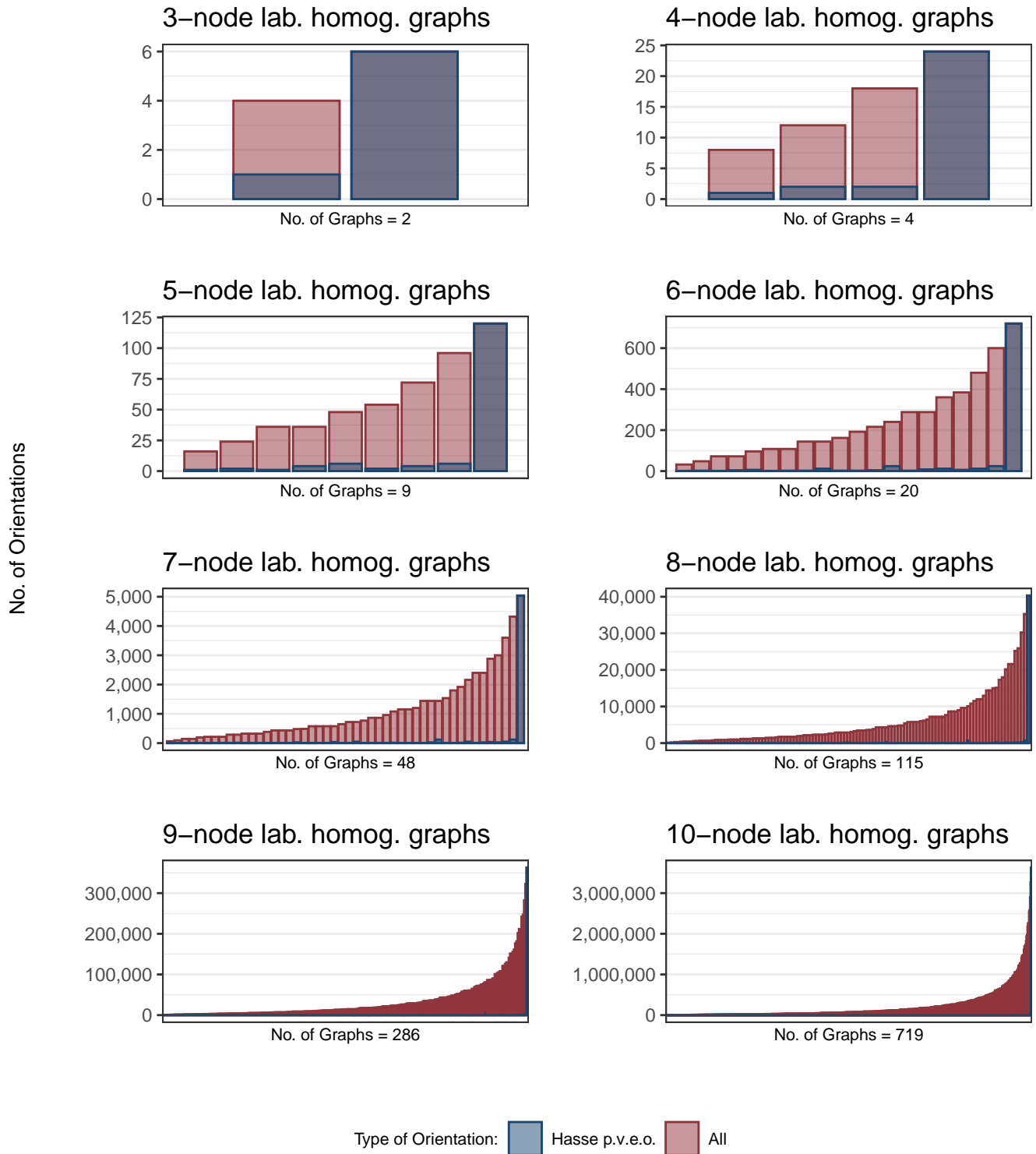


Figure 7-2.: Number of labelled acyclic orientations of connected homogeneous graphs.

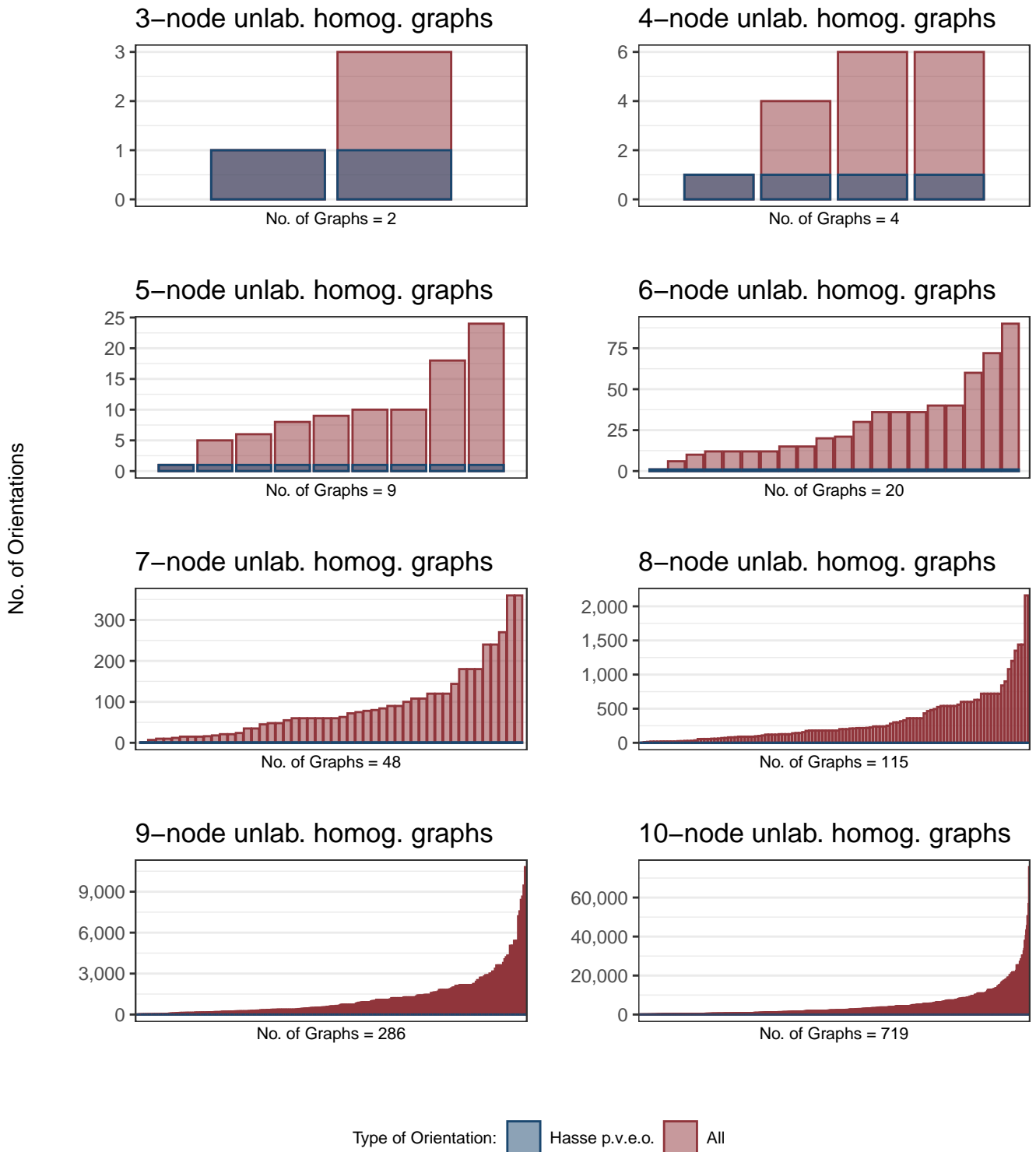


Figure 7-3.: Number of unlabelled acyclic orientations of connected homogeneous graphs. Note that for any such graph there is only one unlabelled acyclic Hasse perfect vertex order.

where $\text{Aut}(G)$ is the automorphism group of G , and $G_A = (V_A, E_A)$ is the quotient graph of G under automorphism A , which is found by merging (specifying) nodes in V that belong to the same cycle of A .

Proof. See [83] and [84]. □

Proposition 8. *Let $G = (V, E)$ be a homogeneous undirected graph. Then, any two labelled acyclic orientations of G that follow a Hasse perfect vertex elimination order are isomorphic. In other words, up to isomorphism, there is only one directed acyclic graph \widehat{G} with skeleton G that follows a Hasse perfect vertex elimination order. Furthermore, the number of labelled acyclic orientations of G that follow a Hasse perfect vertex elimination order is given by*

$$O_l(G) = \prod_{C \in \mathcal{C}} |C|!,$$

where \mathcal{C} is the set of equivalence classes over V induced by the equivalence relation R defined by

$$v_i R v_j \iff \text{adj}(v_i, G) \cup \{v_i\} = \text{adj}(v_j, G) \cup \{v_j\}.$$

Proof. See appendix. □

Altogether, these observations show that Relevance Networks should not be expected to recover a GRN asymptotically, assuming a Gaussian Copula SEM as the data generating mechanism. In this context, we make a comment on the *type* of mistakes that Relevance Networks will make. A simple argument shows that, under the general causal graphical model (Model 1), false negatives will not be prevalent as long as the mutual information matrix can be accurately (consistently) estimated. This is because adjacent nodes in a causal graph are marginally dependent under the assumption of faithfulness. Therefore, their mutual information is non-zero, which will be detected with high probability by a consistent estimator as sample size grows.

The implication of the argument above is that all of the asymptotic bias of a Relevance Networks estimator will in the form of false positives – that is, edges that are not present in the underlying GRN but are predicted to exist. Figure 7-4 shows the distribution of the False Positive Rate (FPR) that results from approximating randomly sampled directed acyclic graphs that are assumed to represent Gaussian Copula SEM data generating mechanisms (Model 2) by their asymptotic Relevance Networks. The observed masses on $FPR = 1$ indicate a noticeable tendency of Relevance Networks to output *complete* GRN skeletons, where all possible undirected edges are predicted to exist.

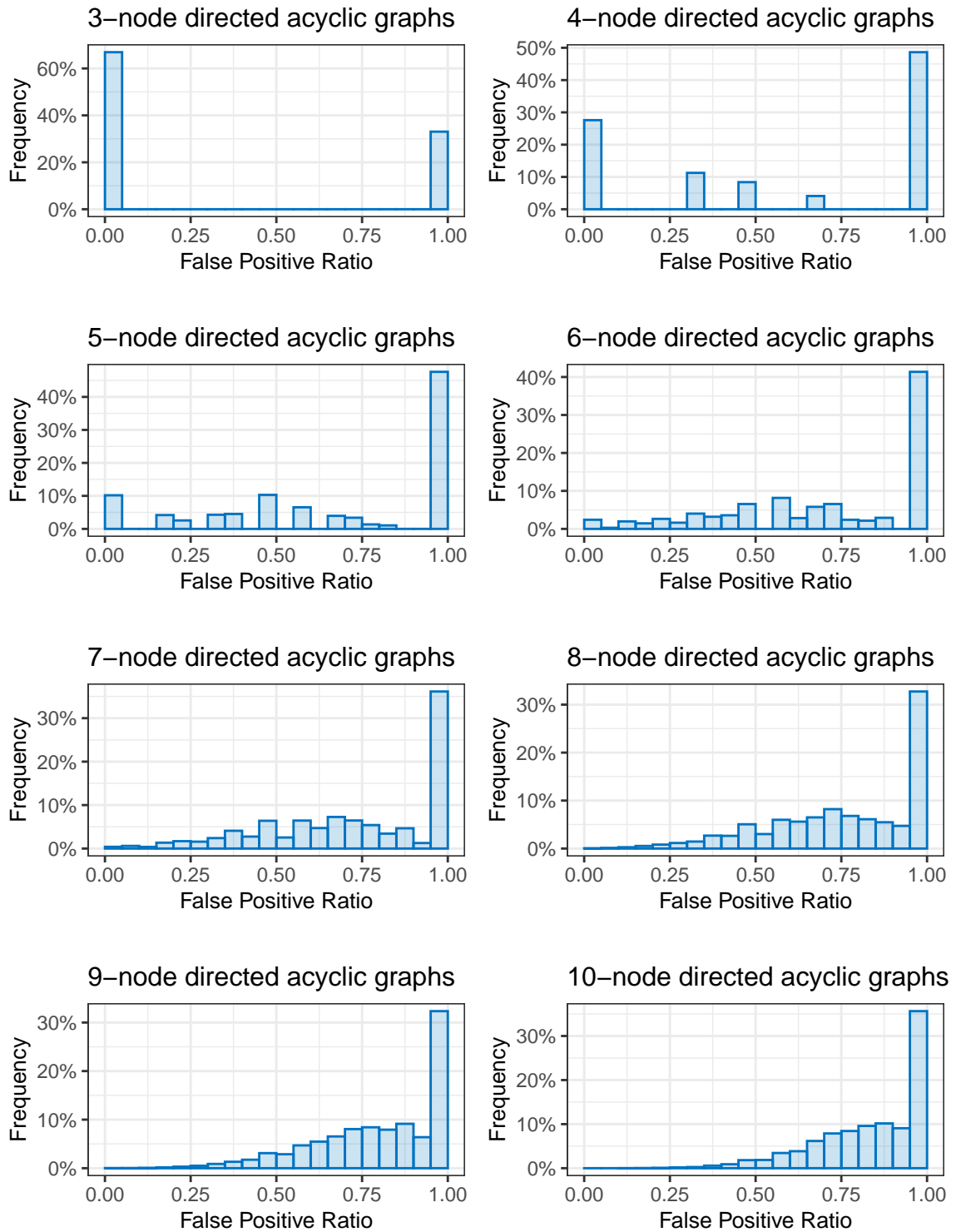


Figure 7-4.: Histograms of False positive rate of Relevance Networks on randomly chosen unlabelled undirected connected graphs and acyclic orientations. Each panel is based on 100,000 samples of connected unlabelled undirected graphs and orientations. Note the masses on $FPR = 1$.

Aggregated ranks of inference algorithms by average AUROC

Algorithm	1	2	3	4	5	6	7	8	9	10	Total
GENIE3	60	57	3	0	0	0	0	0	0	0	120
TIGRESS	56	54	7	3	0	0	0	0	0	0	120
NARROMI (def.)	3	1	54	20	11	11	5	10	5	0	120
Rel. Net.	0	4	21	59	27	5	2	1	1	0	120
NARROMI (c.v.)	0	3	11	12	28	24	17	10	9	6	120
MRNET	1	1	11	12	31	50	10	4	0	0	120
PC (Gaussian)	0	0	12	6	11	13	28	20	30	0	120
CLR	0	0	1	8	10	3	29	14	27	28	120
ARACNE	0	0	0	0	2	11	26	54	26	1	120
PC (disc.)	0	0	0	0	0	3	3	7	22	85	120
Total	120	120	120	120	120	120	120	120	120	120	1200

Table 7-1.: Frequency of rank among algorithms across all combinations of simulation options (network, marginal distribution/transformation, sample size, noise-to-signal ratio), by average AUROC.

7.2. Simulation Study

In this section we describe some salient patterns observed in the simulation experiments outlined in Chapter 6. Most of the reported patterns can be visually confirmed in Figures 7-5 to 7-22, which present compact summaries of algorithm performance in a selection of the simulation experiments carried out. A complete set of such plots will be made available online. We warn that in all of these illustrations the optimal value of the metric depicted is fixed at the right side of the x -axis.

7.2.1. Gaussian Copula SEM Simulations

For many simulation specifications GRN inference algorithms seem to be systematically biased

An inspection of Figures 7-5 to 7-16 reveals that for many simulation exercises and algorithms, judging by the performance metrics considered, there is no clear convergence of estimated causal graphs to the targeted “ground truth” graph skeletons with increasing sample size. While in many cases performance metrics do improve with larger sample sizes, these do not unambiguously appear to approach the optimal values for such metrics (0 or 1), which is what would be expected if inference algorithms accurately captured graph structure with increasingly high probability.

GENIE3 and TIGRESS are the best performing algorithms overall by AUROC, but sidestep the issue of discretization

When judging GRN inference algorithm performance by AUROC, GENIE3 and TIGRESS consistently outperform other algorithms. Table 7-1 summarizes the rankings by average AUROC of all inference algorithms in all simulation experiments for Gaussian Copula SEMs. GENIE3 and TIGRESS occupy the top two ranks in a large majority (144/152) of experiments. Essentially identical results hold when ranking algorithms by accuracy with top- m thresholding.

Despite scoring edges appropriately, GENIE3 and TIGRESS suffer from the drawback of not including straightforward methods to threshold these scores to obtain a discretized GRN estimate. In the absence of an externally supplied threshold, these algorithms generally output non-zero edge weights between all marginally dependent gene expressions, regardless of whether their coexpression patterns reflect causal influence or not. This can be seen by observing Figures 7-11 to 7-16, where, repeatedly, the zero-patterns of the non-thresholded outputs of GENIE3 and TIGRESS approximate the true underlying GRN by reproducing the zero-pattern of the mutual information matrix (equivalent to the zero pattern of BDB^T).

Among algorithms with default or built-in discretization, the PC algorithm with Gaussian conditional independence tests is most accurate

A downside of measuring algorithm performance through AUROC is that it bypasses the problem of how to discretize estimates appropriately. In this regard, when examining algorithms with default or built-in discretization procedures, we find that the PC algorithm based on Gaussian conditional mutual information consistently ranks within the top three algorithms by average accuracy (Table 7-3). In fact, the PC algorithm with Gaussian mutual information is competitive with other algorithms under Top- m edge thresholding, especially with high sample sizes. Across all simulation combinations, and considering both top- m and default discretization, the PC algorithm with Gaussian conditional independence is within the top three performing algorithms in 48 of 120 cases, only surpassed in this regard by GENIE3 and TIGRESS with top- m thresholding. We see these findings as compatible with the fact that the PC algorithm is asymptotically consistent given a consistent test for conditional independence.

Topology of ground truth GRN drives algorithm performance

When comparing simulation results across different ground truth GRNs, we observe that inference algorithms consistently perform better at recovering certain GRNs than others (Table 7-2). Except for both configurations of the PC algorithm, our GRN inference algorithm overwhelmingly perform best on the mCAD and SC50 GRNs.

Counts of best-recovered GRN across simulations, by average AUROC

Algorithm	mCAD	SC50	VSC	HSC	GSD	Totals
ARACNE	8	11	0	5	0	24
CLR	0	24	0	0	0	24
GENIE3	17	4	0	3	0	24
MRNET	3	19	0	2	0	24
NARROMI (c.v.)	14	7	0	0	3	24
NARROMI (def.)	21	1	0	2	0	24
PC (disc.)	1	0	23	0	0	24
PC (Gaussian)	8	2	10	4	0	24
Rel. Net.	22	2	0	0	0	24
TIGRESS	14	2	0	8	0	24
Totals	108	72	33	24	3	240

Table 7-2.: Each cell shows the number of combinations of marginal distributions, sample sizes, and noise-to-signal ratios for which each GRN inference algorithm recovers each displayed ground truth network with the highest average AUROC among all ground truth networks.

We conjecture this partially results from the influence of “collider bias” in the ability of these algorithms to adequately reconstruct the underlying GRNs. Collider motifs typically induce selection bias in SEMs, which affects statistical procedures that in some way depend on conditioning on several variables at a time. We support our conjecture by noting that the accuracy of approximating each GRN by its moral graph is equal to 1 and 0.999 for mCAD and SC50, respectively, and equal to 0.945, 0.929, and 0.901, for VSC, HSC, and GSD, respectively. Since causal graphs and their associated moral graphs differ as a result of the presence of collider motifs, the discrepancy between a causal graph and its moral graph can be considered a proxy for the pervasiveness of this kind of confounding in the SEM.

Our the other hand, the PC algorithm shows a markedly different pattern in this regard. In particular, the well performing PC algorithm with Gaussian conditional independence has similar performance on different graph topologies. This reflects that the PC algorithm was explicitly formulated as an estimator for causal graphs and is known to be consistent regardless of graph topology.

Noise-to-signal ratio has ambiguous effects on algorithm performance

Across data generating models and inference methods, the noise-to-signal ratio r has unclear effects on the performance of the algorithms. Despite the expectation that lower noise

Aggregated ranks by average accuracy with default discretization

Algorithm	1	2	3	4	5	6	7	8	9	10	Total
PC (Gaussian)	56	31	13	1	3	8	1	7	0	0	120
ARACNE	6	13	45	12	16	3	9	14	2	0	120
NARROMI (def.)	14	27	15	14	12	13	16	9	0	0	120
PC (disc.)	20	8	8	8	7	9	30	7	16	7	120
NARROMI (cv)	4	8	12	33	30	17	14	2	0	0	120
CLR	1	12	8	18	14	26	14	5	6	16	120
Rel. Net.	1	2	15	29	18	19	7	28	1	0	120
MRNET	0	0	2	2	16	23	29	48	0	0	120
Total	138	105	122	121	116	118	120	122	185	53	1200

Table 7-3.: Frequency of rank by mean accuracy among algorithms with default/built-in discretization protocols, across all combinations of simulation options (network, marginal distribution/transformation, sample size, noise to signal ratio). Ties in average accuracy are treated equally and assigned the minimum possible rank they achieve.

produces more accurate estimates, we find that over 600 distinct combinations of GRN skeletons, marginal distributions, inference algorithms, and sample sizes, in 376 cases $r = 0.35$ results in better average AUROC scores, while $r = 0.65$ is associated to better AUROC in 222 cases and 2 combinations produce ties.

Our specification of the PC algorithm with discretized data performs poorly

In contrast to the effectiveness of the PC algorithm based on Gaussian mutual information, we find that the PC algorithm based on discretized data is generally inaccurate in recovering graph structure. In additional experiments, we find the accuracy of this algorithm to be very sensitive the number of bins used to discretize. Against the findings in [45], estimating mutual information based on $n^{\frac{1}{3}}$ bins in our simulation experiments leads to underpowered tests of marginal independence, which results in the PC algorithm eliminating excessive numbers of edges in early steps of the procedure. This is reflected in the fact that in 95 out of 120 experiments, this specification of the PC algorithm results in a false negative ratio of over 0.9.

GENIE3 is the most robust algorithm to heavy tailed marginal distributions

When comparing across simulation configurations with varying marginal distributions, we find that GENIE3 suffers the smallest hit to performance from heavy-tailed Pareto distributions of marginals (For example, compare Figure 7-7 to Figures 7-5 and 7-6). This is an expected product of the flexibility afforded to GENIE3 by its use of regression trees as

auxiliary models for edge weighting.

Relevance Networks with thresholding based on statistical tests for null mutual information converge as expected, but they perform well by AUROC and Top-m thresholding

When observing the results of Relevance Networks, a stark contrast appears between their performance when discretized on the basis on statistical tests for zero mutual information (default case) and their performance with Top-m thresholding or by AUROC. In the first case, we often see the Relevance Networks estimator accurately recovering the zero pattern of the mutual information matrix with large sample sizes (for example, see the 'MI matrix approximations' in Figures 7-11 to 7-16), as expected. Whether this provides an good approximation of the causal graph structure is entirely dependent on its particular topology. However, despite this behavior, we also see that Top-m discretization and AUROC show Relevance Networks as generating increasingly better approximations of the causal graph skeleton to be estimated as sample size grows.

We reconcile this apparent contradiction by noting that good performance of Relevance Networks by Top-m thresholding and AUROC is indicative of a correct sorting of true edges *vis-à-vis* absent edges by estimated mutual information values, regardless of whether true mutual information is zero or not. In other words, given several pairs of variables with non-zero mutual information, some of which are adjacent in a causal graph and some of which are not, Top-m thresholding and AUROC reward associations between estimated mutual information values and the true presence of edges in the causal graph. In contrast, the accuracy of Relevance Networks with discretization based on marginal independence testing ignores such association for pairs of variables once they are deemed to be marginally dependent, since it places edges between all such pairs of variables.

The fact that the performance Relevance Networks by AUROC and Top-m thresholding improves with sample size thus suggests that, beyond the binary choice of whether mutual information is zero or not, the *magnitude* of non-zero mutual information is indicative of the presence of edges in our assumed ground truth GRNs, given the associated statistical models we consider.

NARROMI with cross validation is not clearly better nor worse than NARROMI with default parameter values

In Figures 7-5 to 7-16, we observe no clear patterns setting apart NARROMI with cross validation from NARROMI with default parameter values. We conjecture two mechanisms at play that may play a role in producing this phenomenon.

In the first place, we suspect the existence of a theoretical property for cross validated LAD-LASSO regression in estimating graphical models analogous to that stated in [29] for LASSO regression. The authors of [29] prove that estimating LASSO using the optimal $L1$ penalty weight λ in terms of population mean squared prediction error can generate an asymptotically biased estimator of a causal graphical model’s moral graph. If the same were also the case for LAD-LASSO regression, we could expect cross validated LAD-LASSO regression to mistakenly select edges in estimating a causal graph in a systematic manner.

Second, in producing a faithful R implementation of NARROMI based on the code provided by its creators, we observe an important feature of the original formulation. In the original, the authors implement and optimize the loss function

$$\mathcal{L}(\beta_i) = \sum_{k=1}^n |X_{ik} - \sum_j \beta_{ij} X_{jk}| + \lambda \left| \sum_j \beta_{ij} \right|.$$

For this expression, a fixed λ does not imply a fixed relative weight of the $L1$ penalization in the loss function for all sample sizes. Rather, this relative weight behaves as n^{-1} , decaying at a rather high rate. Thus, with standardized data and $n > p$, we could expect NARROMI with the default value of $\lambda = 1$ to become numerically very similar to non-penalized LAD regression as sample size grows. Furthermore, since for the linear Gaussian case LAD regression consistently estimates the conditional mean of the response variable [85], in this case we may expect LAD-LASSO as implemented by the authors of NARROMI to approximate the moral graph of a linear Gaussian SEM.

Relative performance of GRN inference algorithms based on pairwise dependency measures depends on graph topology

As pointed out in Chapter 4, ARACNe, CLR, and MRNET are proposed as refinements or improvements of Relevance Networks. In our simulations, we do not see any of these algorithms consistently outperform Relevance Networks. Rather, their relative performance depends on graph topology and the metric considered. As noted above, for GRN topologies whose mutual information matrices approximate them poorly, such as HSC and GSD, Relevance Networks with default discretization tends to produce similarly poor accuracy (see Figures 7-14 to 7-16). In these cases, we see the sparsity-inducing mechanisms of CLR, ARACNe, and MRNET to improve accuracy. On the other hand, when the zero-pattern of the mutual information matrix closely resembles the ground truth topology, or when evaluating performance by AUROC, we do not see a clear pattern in favor of ARACNe, CLR, or MRNET (see Figures 7-11 to 7-13). In these cases, these algorithms are overly assiduous in eliminating edges in comparison to Relevance Networks.

The variability of performance metrics is small

One of the guiding principles of this study is to consider GRN inference algorithms as estimators in probabilistic models, and to thus account for their variability under repeated sampling. In this sense, the error bars in the presented figures suggest that, when applied to random gene expression data, performance metrics of algorithm outputs are not very variable. This is especially the case especially when comparing variability around the average performance metrics to the deviation between average performance metrics and optimal performance metrics. However, this interpretation should be taken with caution. Since all the considered performance metrics are real numbers in $[0, 1]$, and also since their optimal values are either 0 or 1, we may expect that variance from the data generating process be reflected not only in the variance of performance metrics, but in their mean values as well. These observations raise the issue of how to define distances in spaces of graphs that lend themselves to reasonable interpretations of variability when estimating graph-valued parameters.

7.2.2. Cosine transformation SEM and Thermodynamic ODE Simulations

GENIE3 and Relevance Networks can detect non-linear dependency structure better than other algorithms

Figures 7-17 to 7-20 illustrate that GENIE3 and Relevance Networks can partially capture the non-linear and non-monotonic dependency structure of the cosine transformation SEM. On one hand, we observe that Relevance Networks with default discretization converge to the mutual information matrix. Furthermore, both GENIE3 and Relevance Networks perform comparatively well in recovering GRN skeleton structure. These results are consistent with the flexibility of the estimators on which these algorithms rely (discretized mutual information for Relevance Networks, and regression trees for GENIE3)

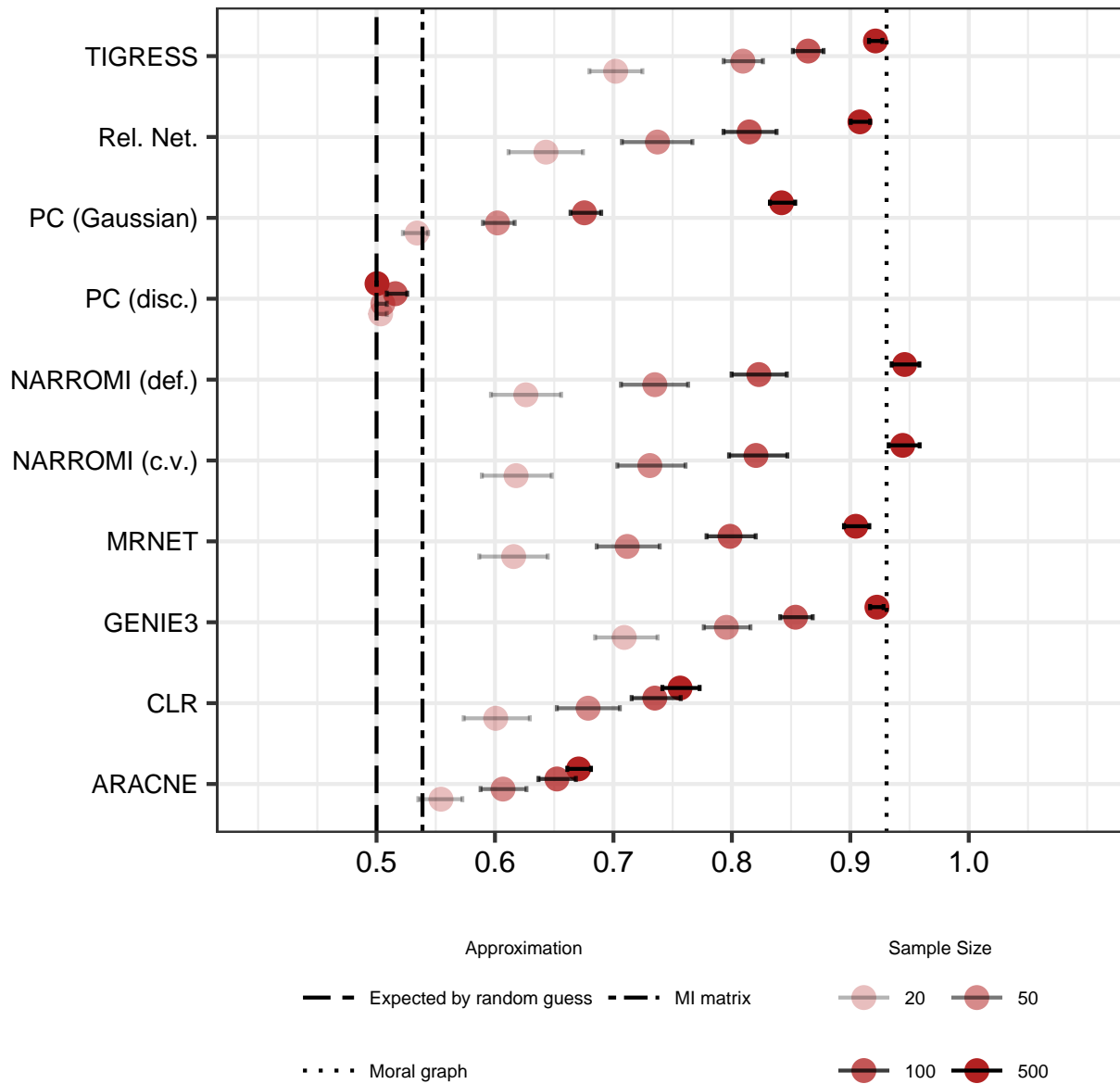
Overall, GRN inference algorithm performance is worse for the cosine transformation SEM and for the Thermodynamic ODE model than for Gaussian Copula SEMs

Although more exhaustive simulations are required to make conclusive statements, preliminarily we find that our selected GRN inference algorithms are less accurate in reconstructing a graph skeleton with the cosine transformation SEM than a causal graph skeleton with a Gaussian Copula Model. Also, as Figures 7-21 and 7-22 show, our evaluated GRN inference algorithms perform worse in recovering the GRN associated to the Thermodynamic ODE model we simulate than they do under Gaussian Copula SEMs. For example, the highest average AUROC in these experiments is 0.83, achieved by TIGRESS with a sample size of 200. This is lower than any average AUROC of the top-performing algorithm in any simu-

lations with Gaussian Copula SEM and a sample size of 100 or more observations, except for the Pareto Gaussian Copula simulations for the GSD network with 100 observations. These partial results underscore the need of formulating and assessing the behavior of GRN inference algorithms in the context of specific models of gene regulation.

AUROC

Net: GSD.
 Model: Linear Gaussian SEM.
 Noise-to-Signal: 0.35.



Dots represent averages. Bars cover the range between percentiles 25 and 75.

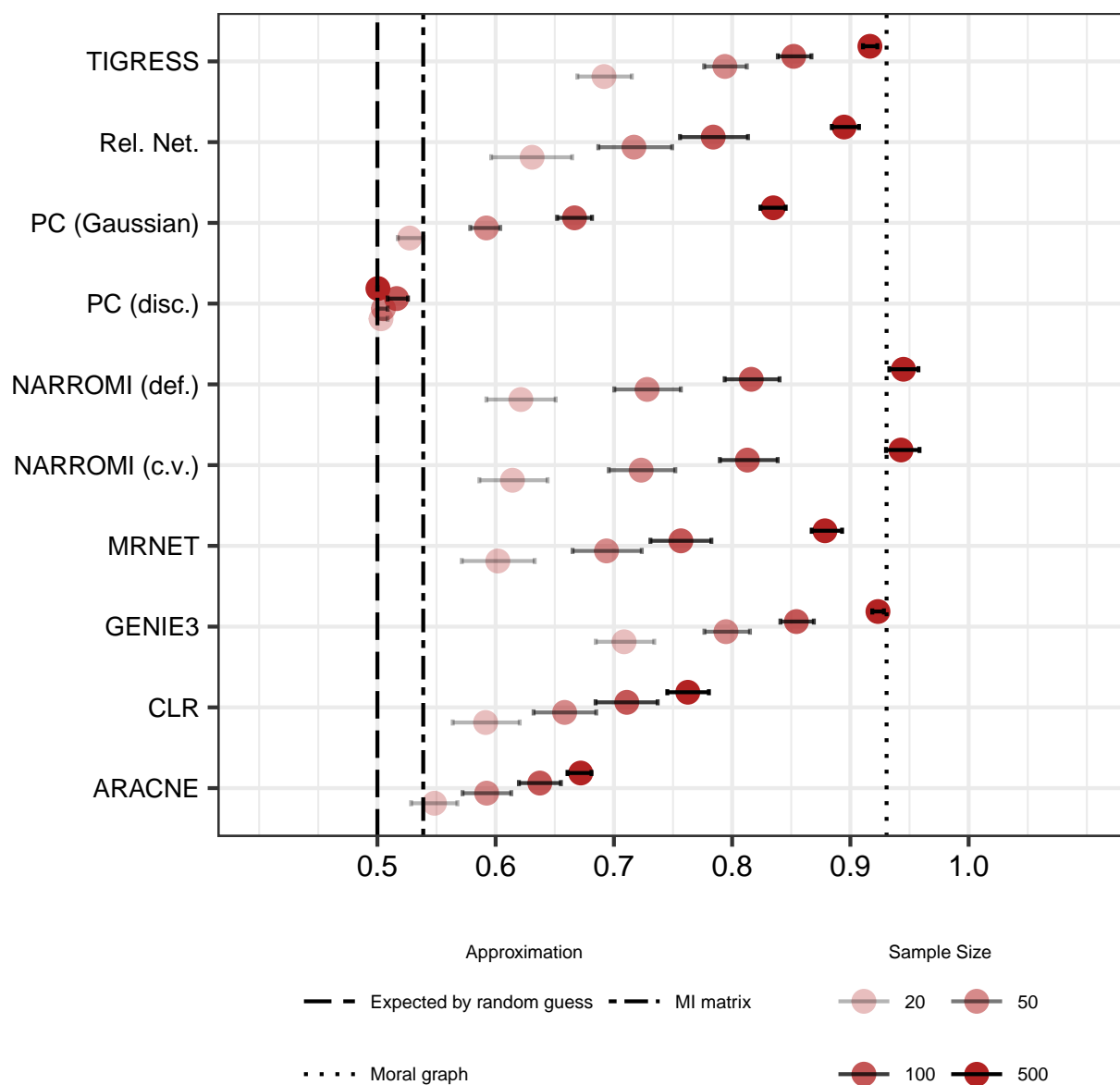
Figure 7-5.: Algorithm AUROC on linear Gaussian SEM on GSD network.

AUROC

Net: GSD.

Model: Gaussian copula + Laplacian marginals.

Noise-to-Signal: 0.35.

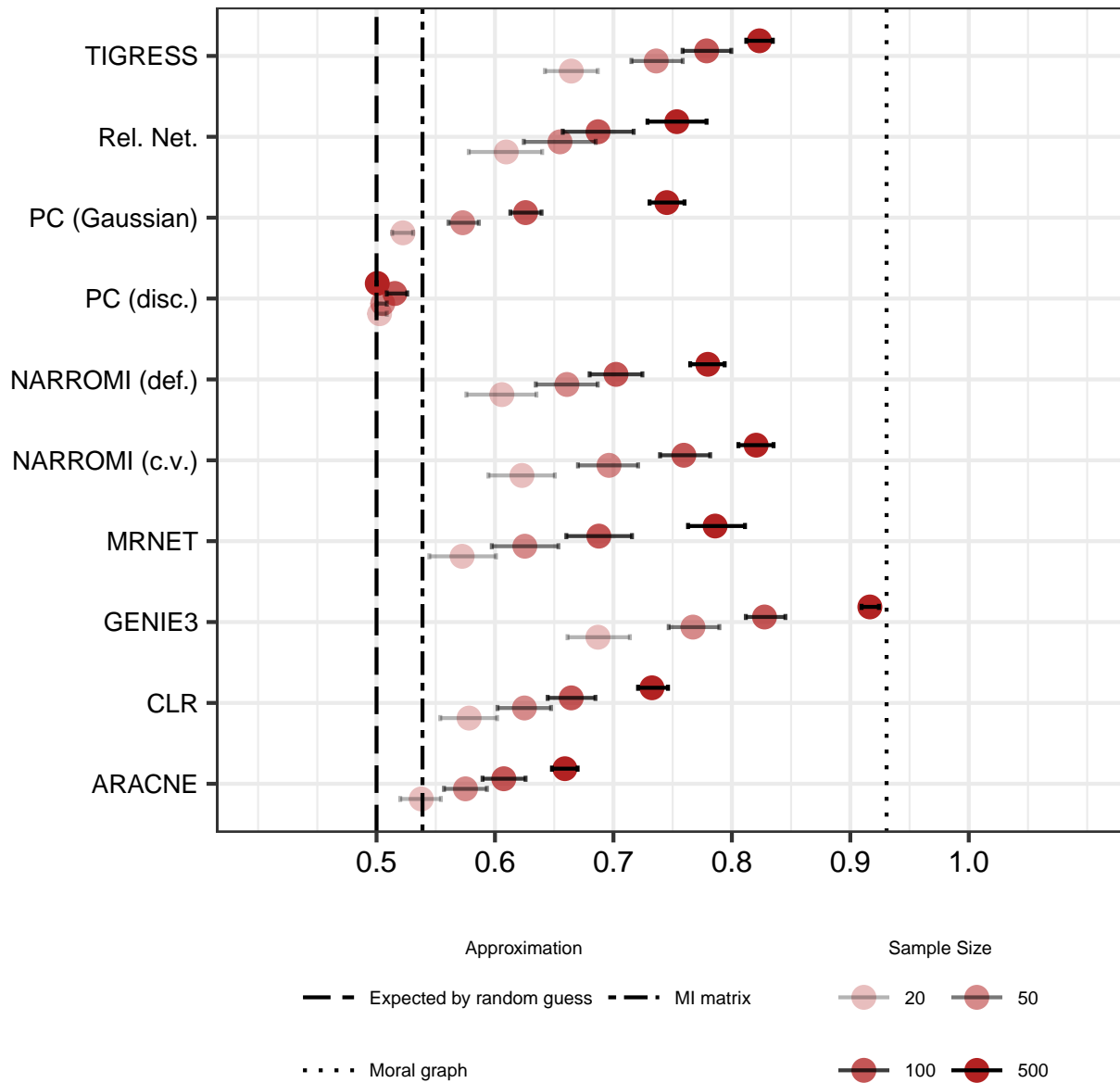


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-6.: Algorithm AUROC on Gaussian Copula SEM with Laplacian marginals on GSD network.

AUROC

Net: GSD.
 Model: Gaussian copula + Pareto marginals.
 Noise-to-Signal: 0.35.

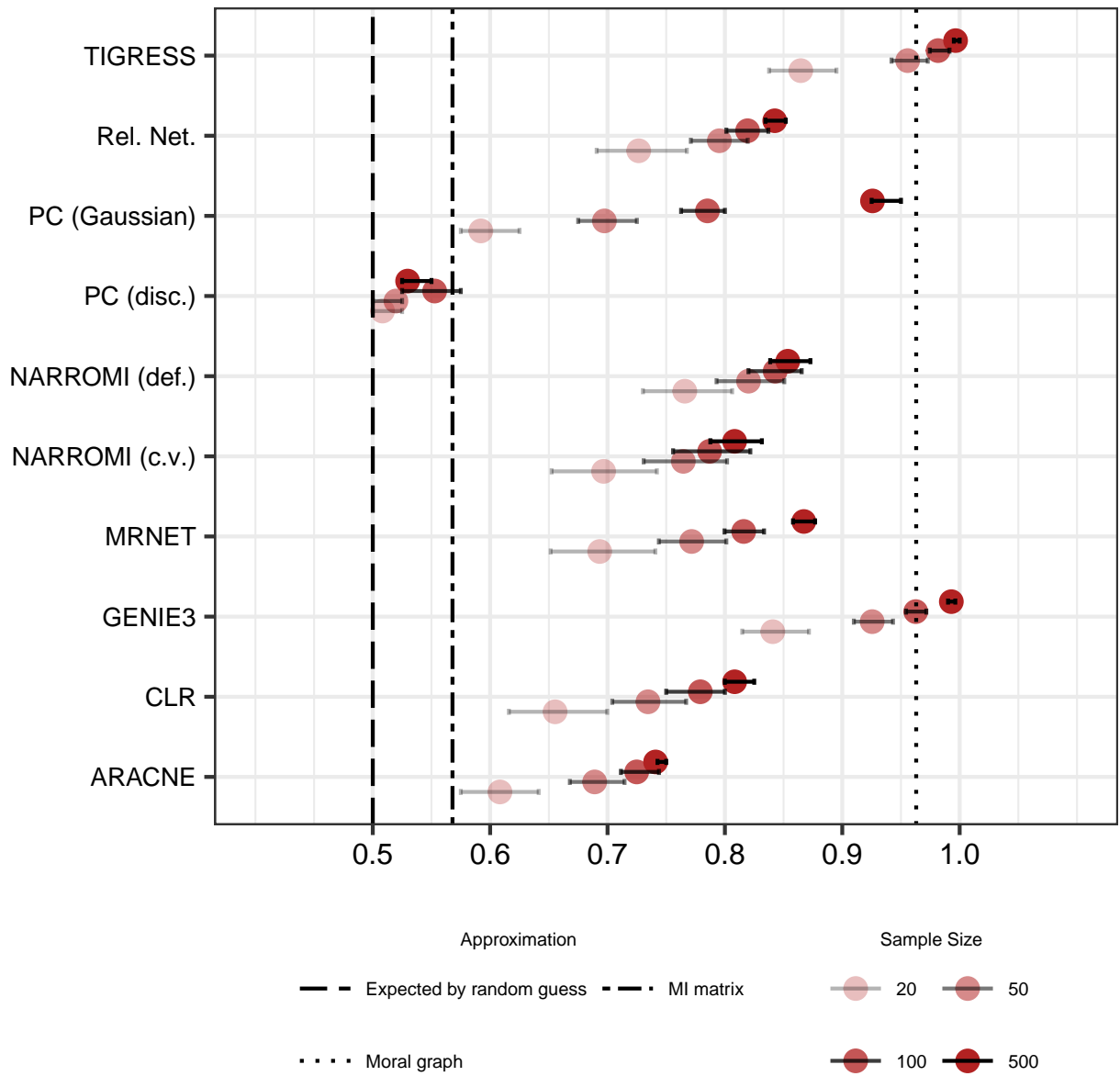


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-7.: Algorithm AUROC on Gaussian Copula SEM with Pareto marginals on GSD network.

AUROC

Net: HSC.
 Model: Linear Gaussian SEM.
 Noise-to-Signal: 0.35.



Dots represent averages. Bars cover the range between percentiles 25 and 75.

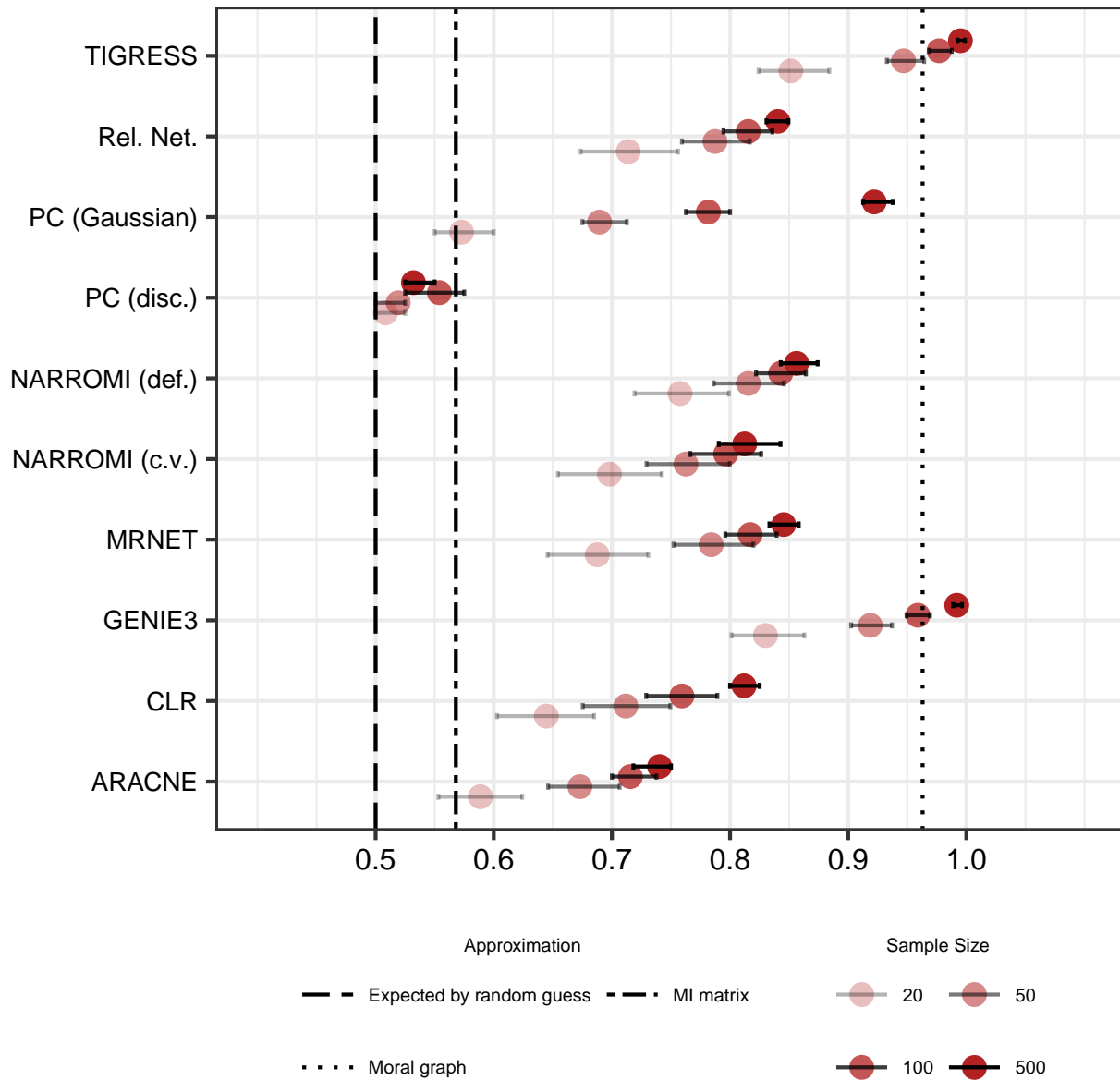
Figure 7-8.: Algorithm AUROC on linear Gaussian SEM on HSC network.

AUROC

Net: HSC.

Model: Gaussian copula + Laplacian marginals.

Noise-to-Signal: 0.35.

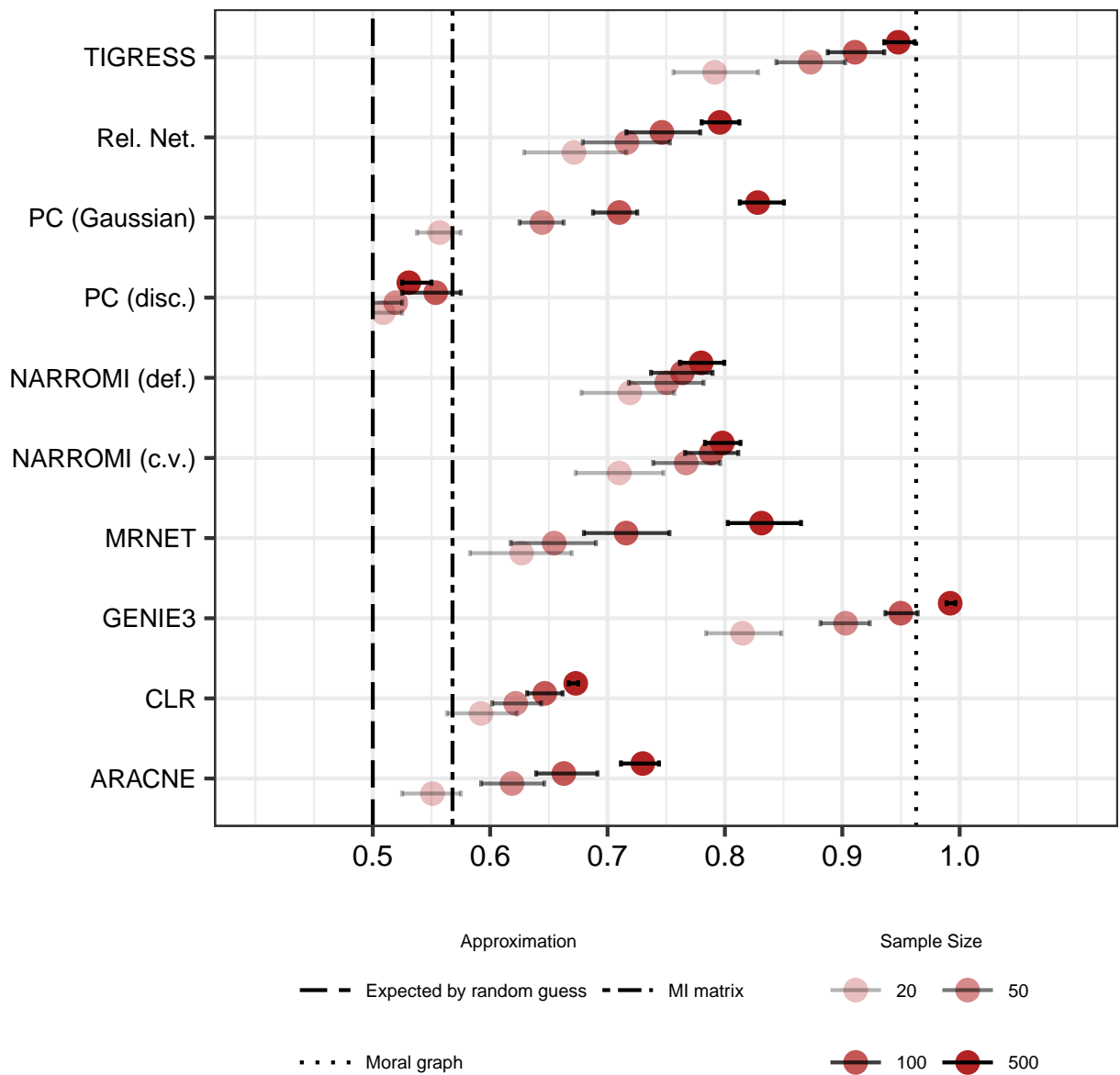


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-9.: Algorithm AUROC on Gaussian Copula SEM with Laplacian marginals on HSC network.

AUROC

Net: HSC.
 Model: Gaussian copula + Pareto marginals.
 Noise-to-Signal: 0.35.

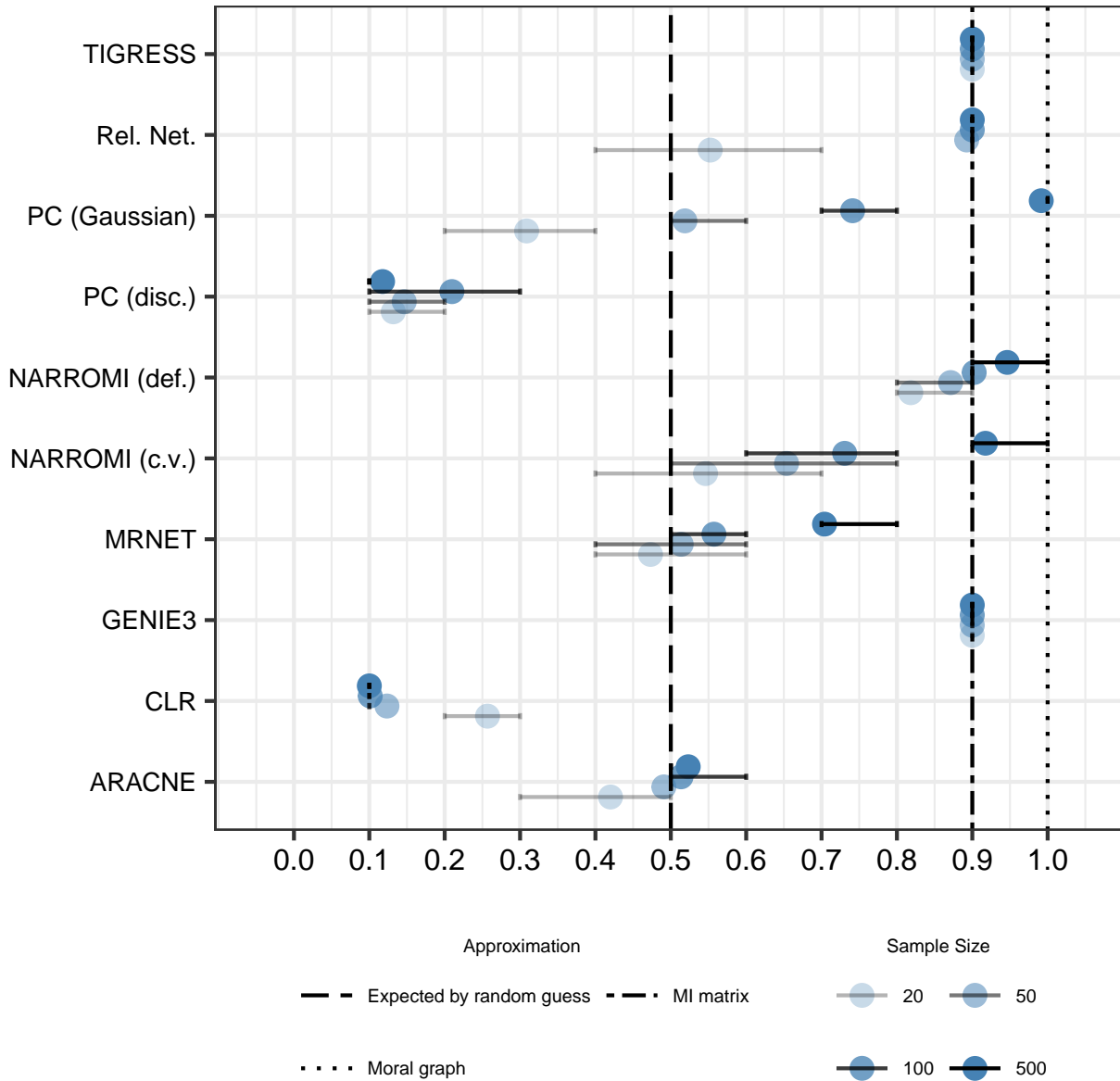


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-10.: Algorithm AUROC on Gaussian Copula SEM with Pareto marginals on HSC network.

Accuracy

Net: mCAD.
 Model: Linear Gaussian SEM.
 Noise-to-Signal: 0.35.
 Threshold: Default.

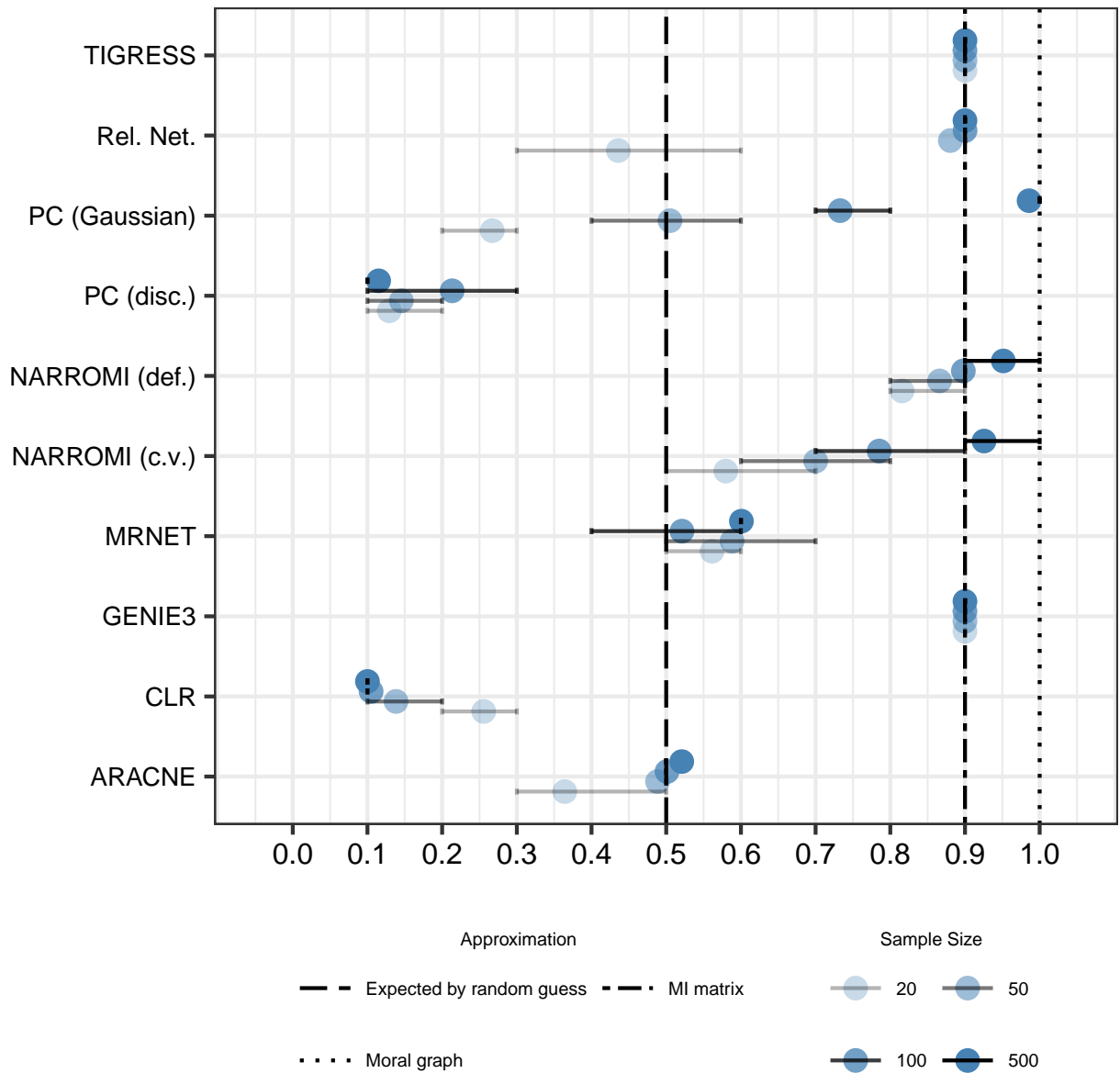


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-11.: Algorithm accuracy (default discretization) on linear Gaussian SEM on mCAD network.

Accuracy

Net: mCAD.
 Model: Gaussian copula + Laplacian marginals.
 Noise-to-Signal: 0.35.
 Threshold: Default.

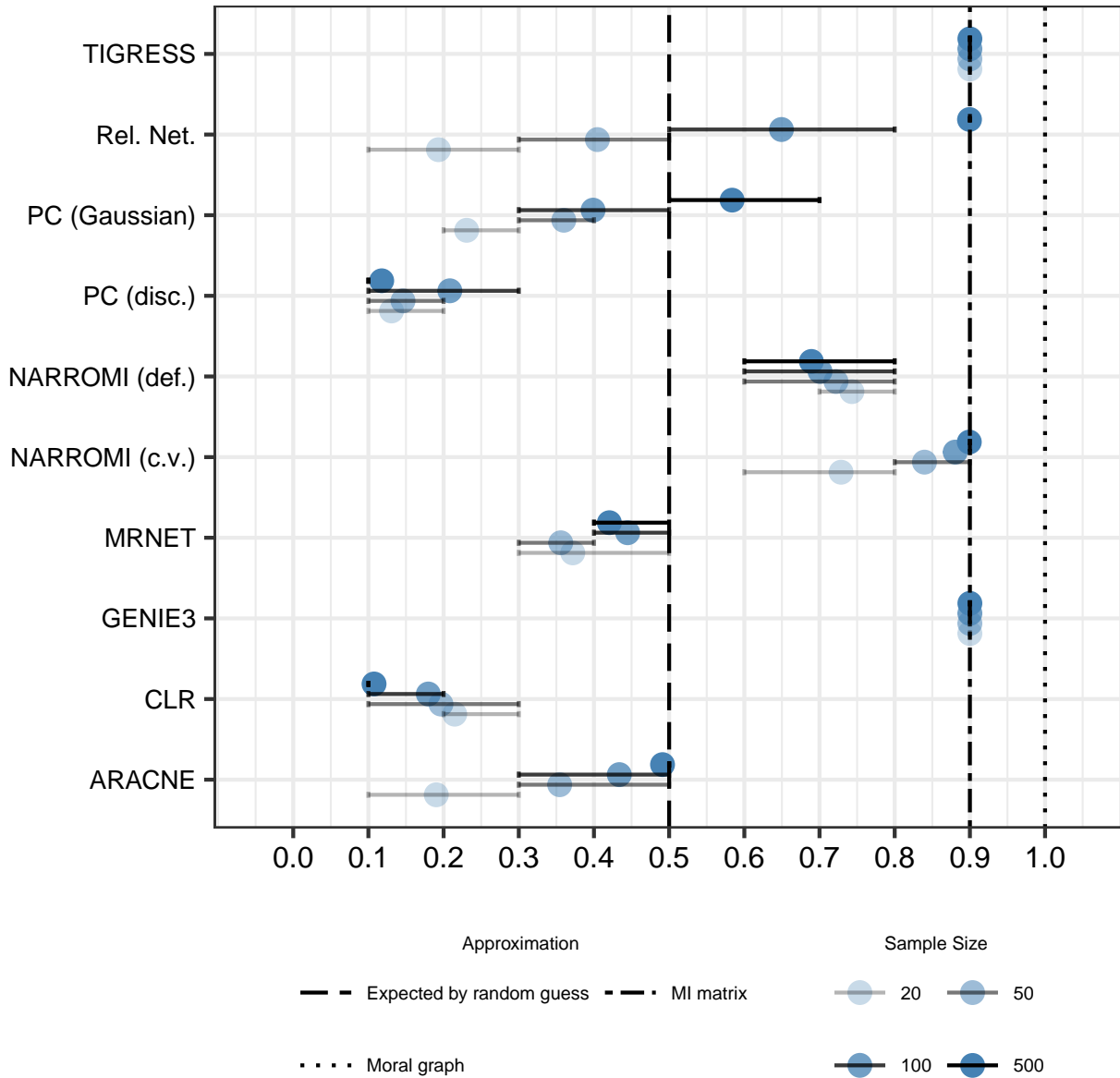


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-12.: Algorithm accuracy (default discretization) on Gaussian Copula SEM with Laplacian marginals on mCAD network.

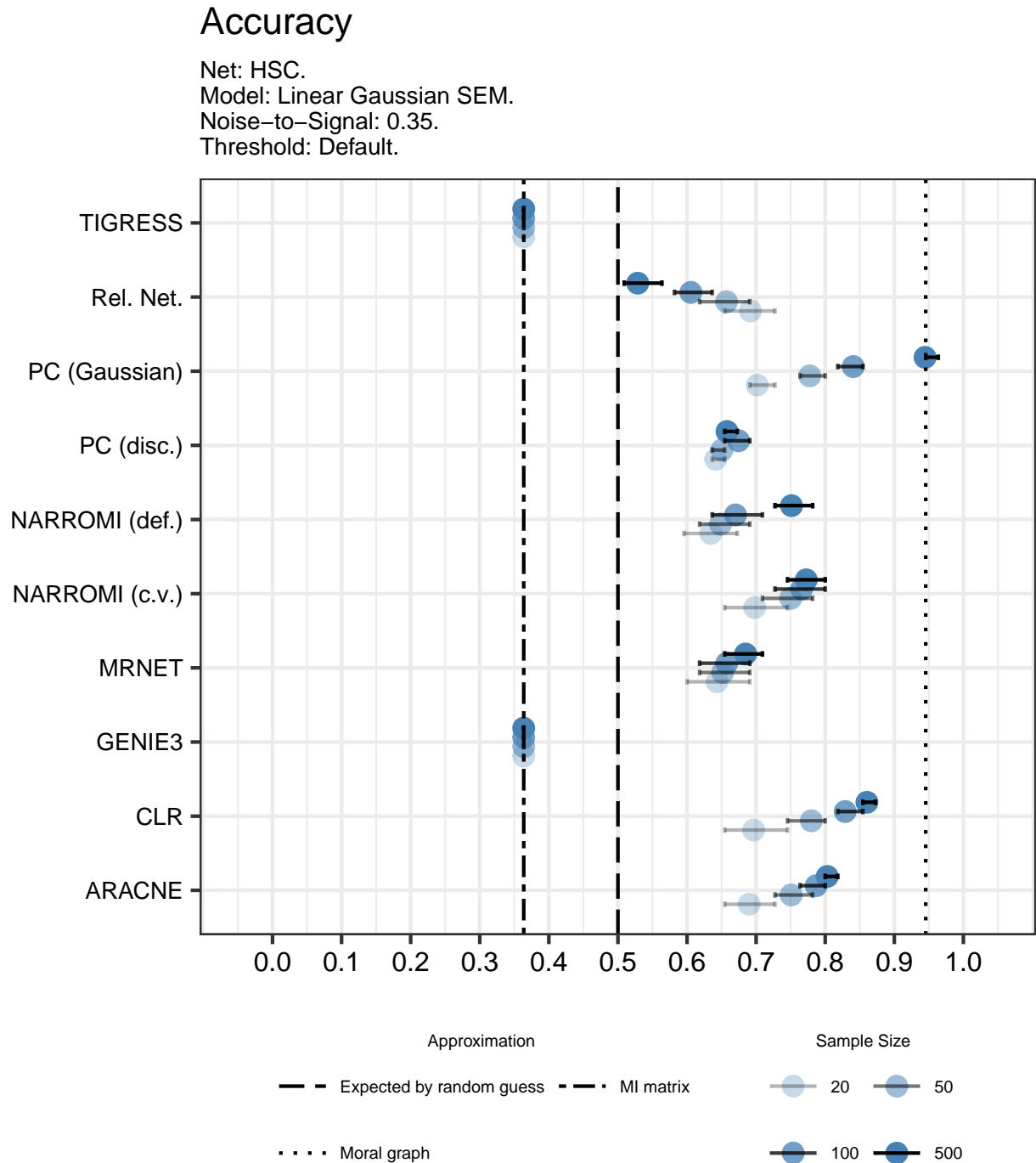
Accuracy

Net: mCAD.
 Model: Gaussian copula + Pareto marginals.
 Noise-to-Signal: 0.35.
 Threshold: Default.



Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-13.: Algorithm accuracy (default discretization) on Gaussian Copula SEM with Pareto marginals on mCAD network.

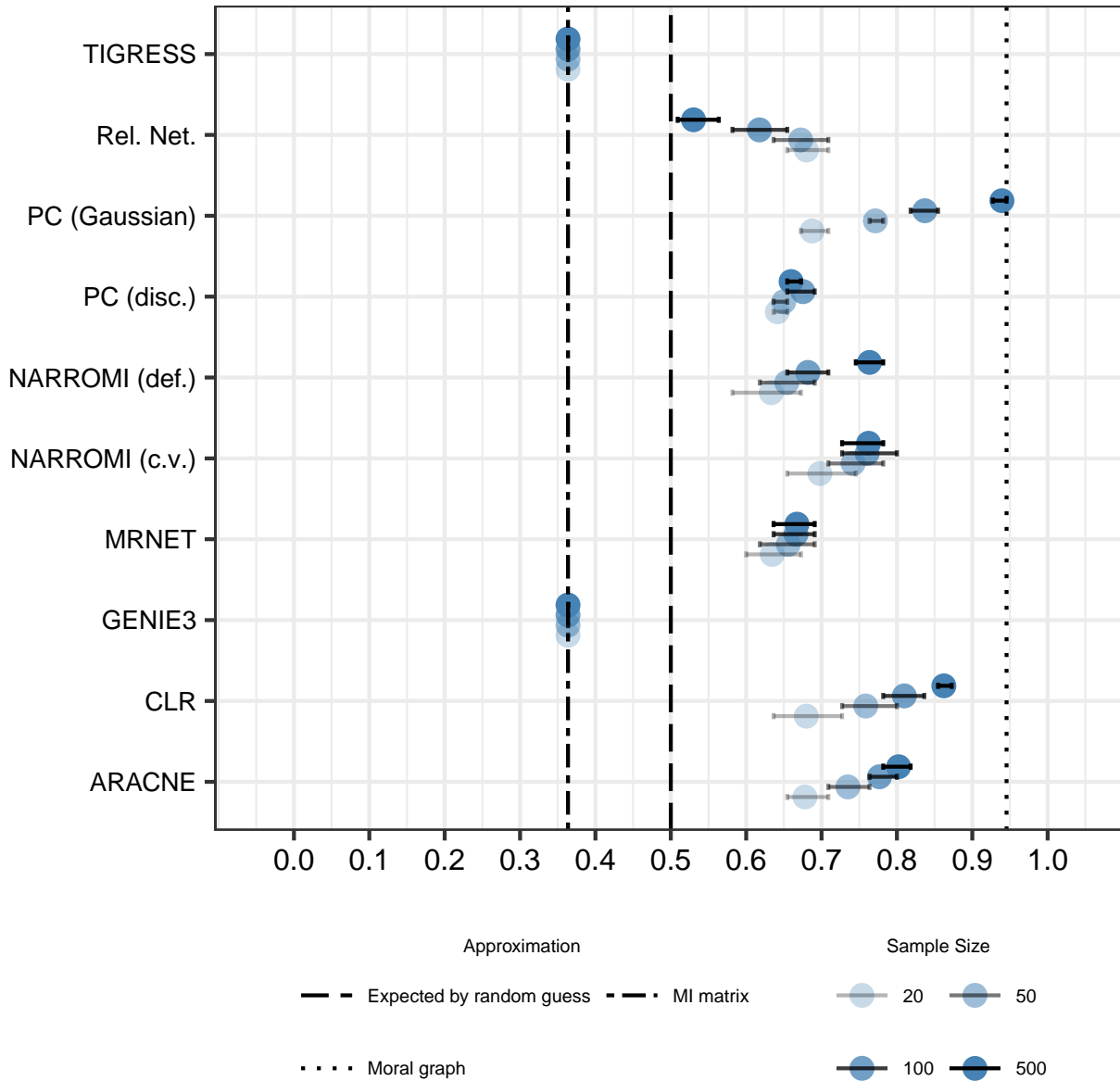


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-14.: Algorithm accuracy (default discretization) on linear Gaussian SEM on HSC network.

Accuracy

Net: HSC.
 Model: Gaussian copula + Laplacian marginals.
 Noise-to-Signal: 0.35.
 Threshold: Default.



Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-15.: Algorithm accuracy (default discretization) on Gaussian Copula SEM with Laplacian marginals on HSC network.

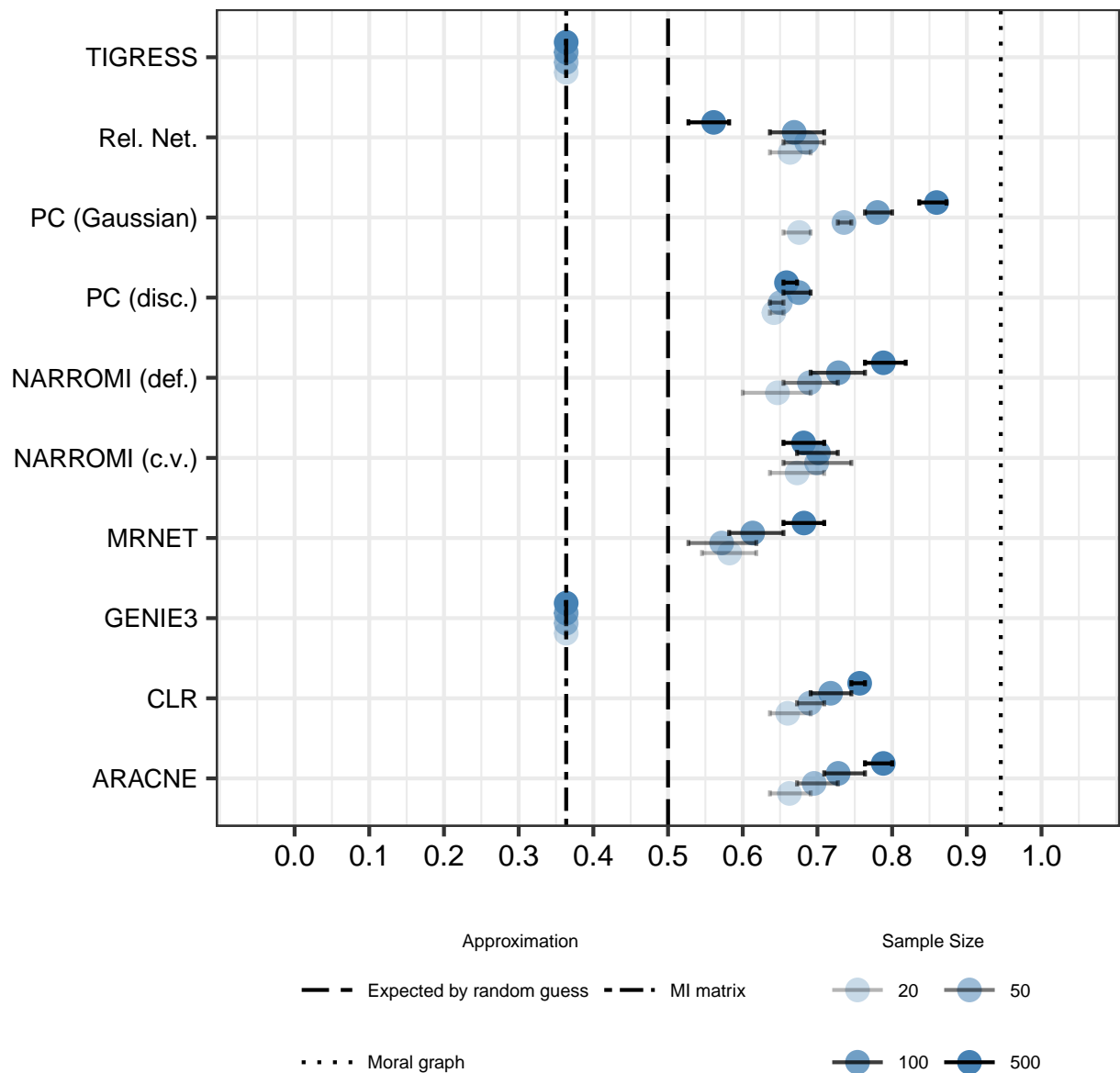
Accuracy

Net: HSC.

Model: Gaussian copula + Pareto marginals.

Noise-to-Signal: 0.35.

Threshold: Default.

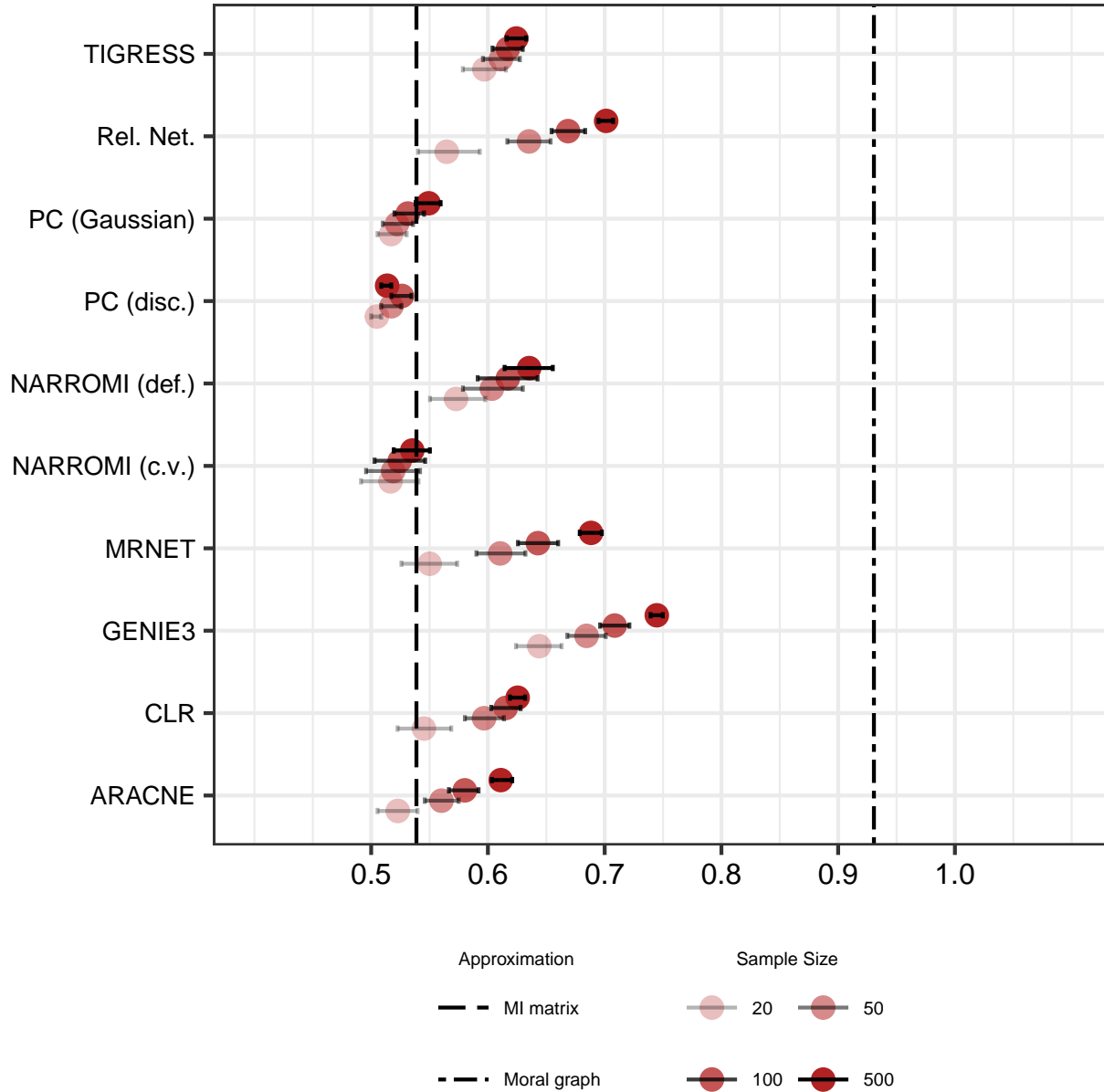


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-16.: Algorithm accuracy (default discretization) on Gaussian Copula SEM with Pareto marginals on HSC network.

AUROC

Net: GSD.
Model: Cosine transformation.

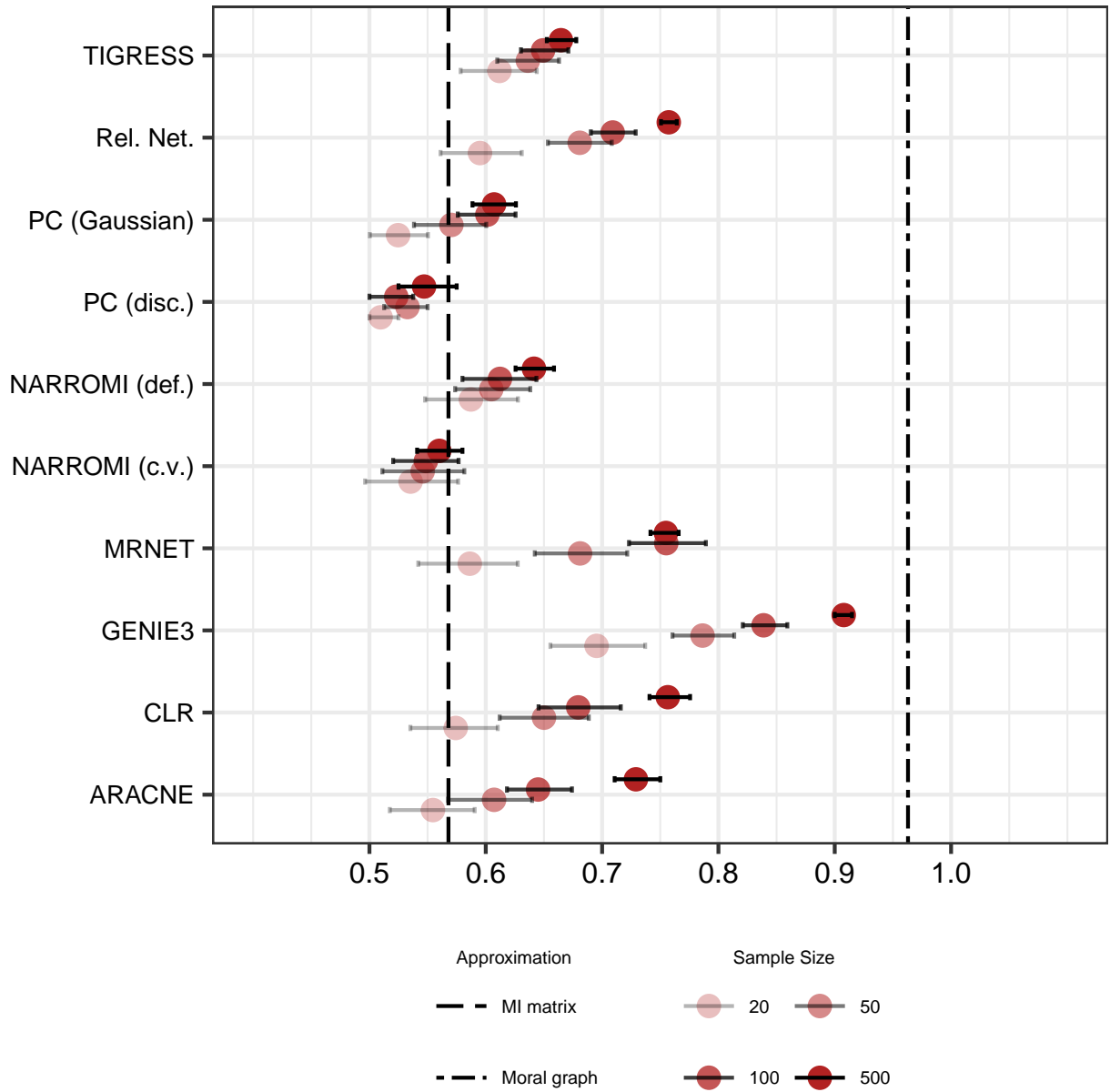


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-17.: Algorithm AUROC on cosine transformation causal graphical model on GSD network.

AUROC

Net: HSC.
Model: Cosine transformation.

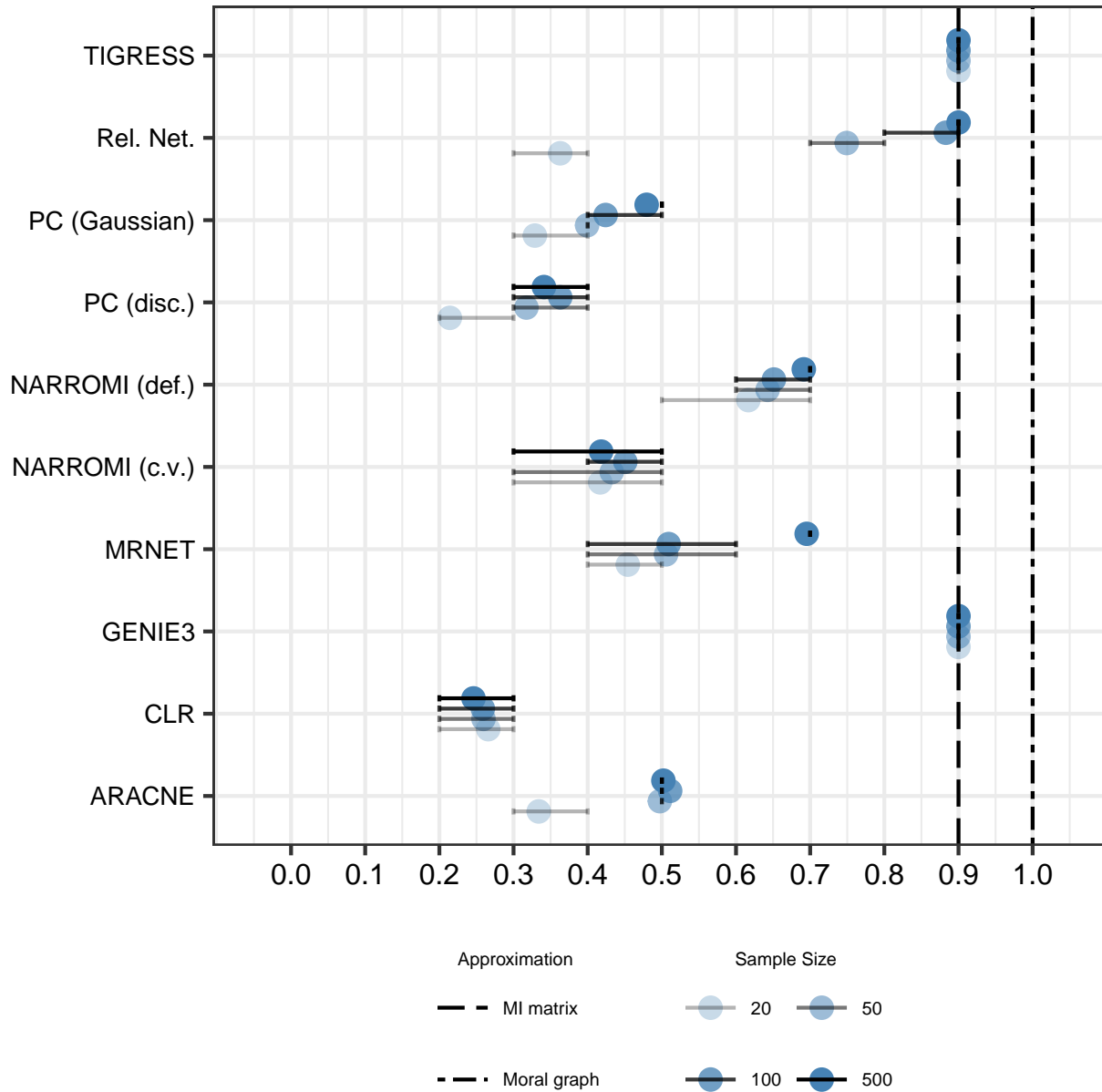


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-18.: Algorithm AUROC on cosine transformation causal graphical model on HSC network.

Accuracy

Net: mCAD.
Model: Cosine transformation.
Threshold: Default.

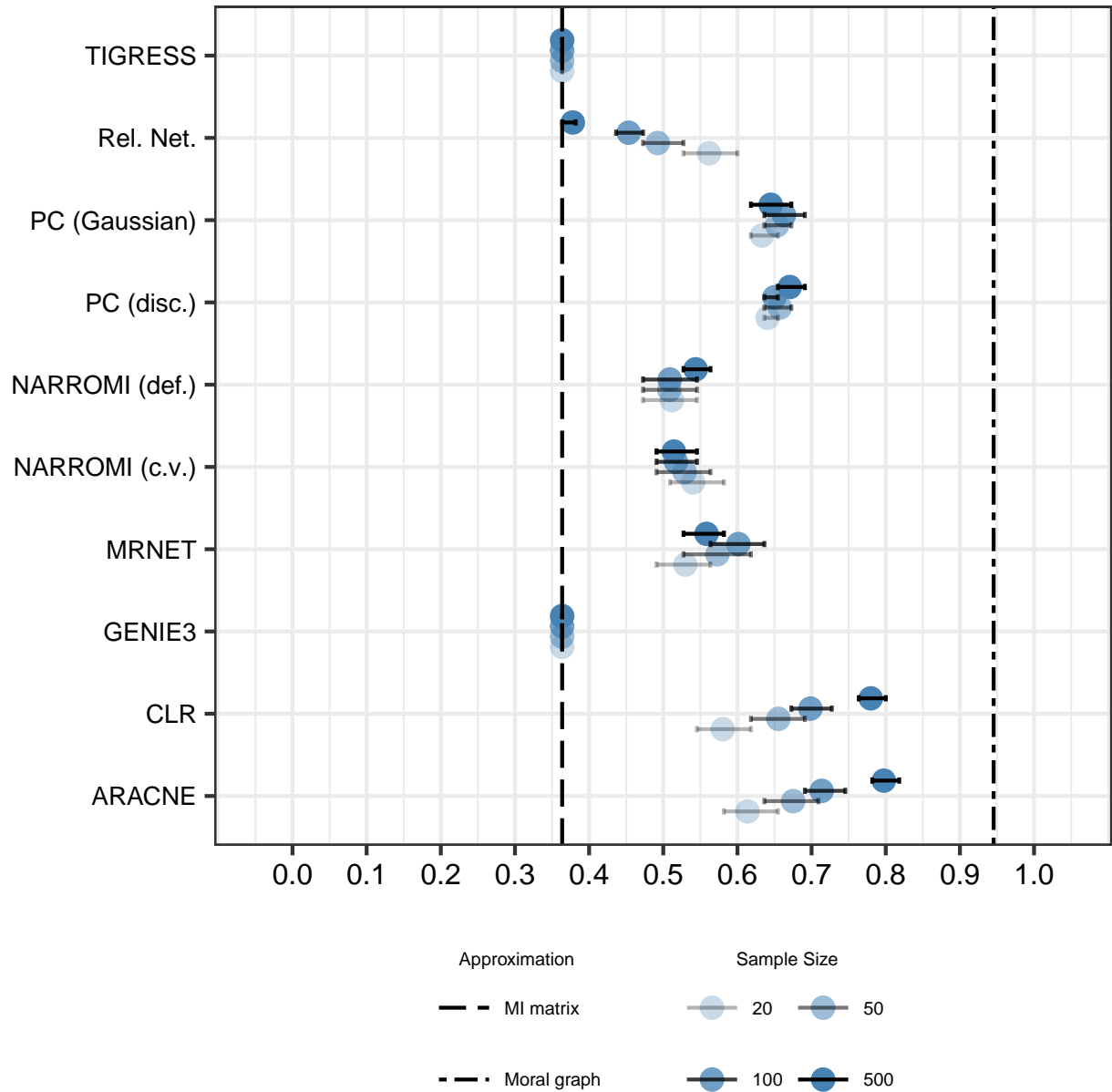


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-19.: Algorithm accuracy (default discretization) on cosine transformation causal graphical model on mCAD network.

Accuracy

Net: HSC.
Model: Cosine transformation.
Threshold: Default.

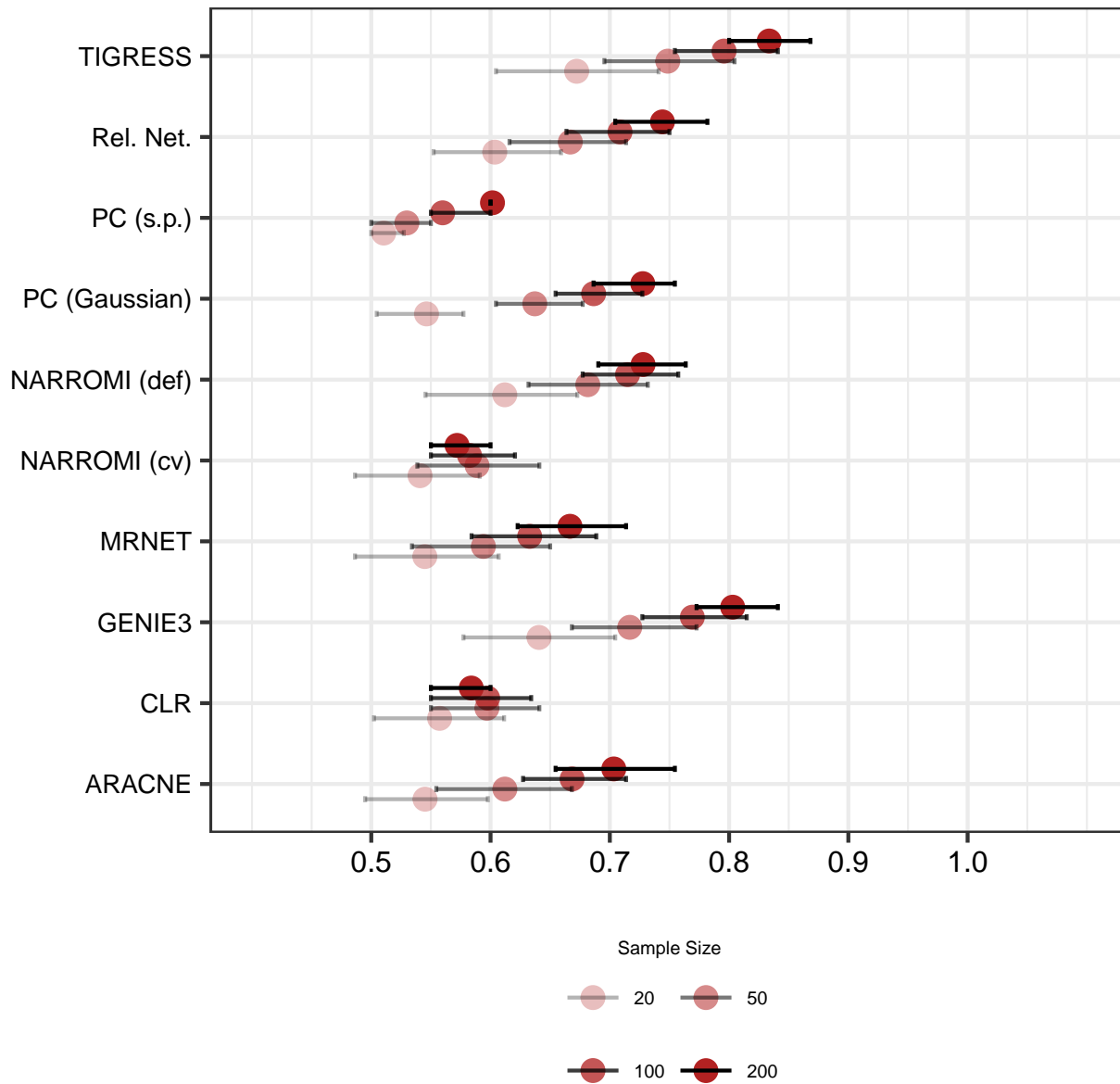


Dots represent averages. Bars cover the range between percentiles 25 and 75.

Figure 7-20.: Algorithm accuracy (default discretization) on cosine transformation causal graphical model on HSC network.

AUROC

Net: VSC.
 Model: Thermodynamic ODE model.
 k (noise parameter) = 0.1.

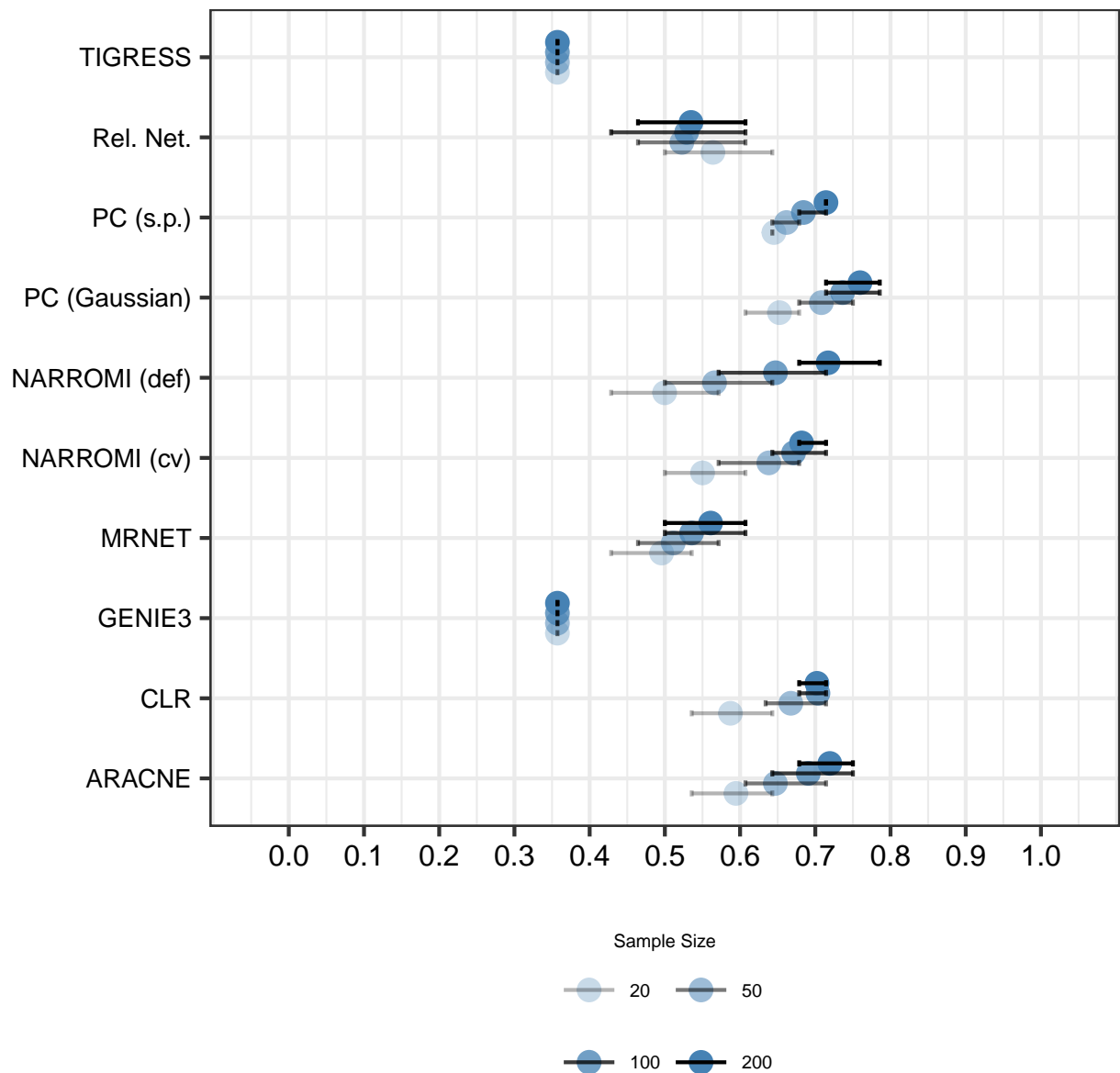


Dots represent means. Bars cover the range between percentiles 25 and 75.

Figure 7-21.: Algorithm AUROC on Thermodynamic ODE Model on VSC network.

Accuracy

Net: VSC.
 Model: Thermodynamic ODE model.
 k (noise parameter) = 0.1.
 Threshold: Default.



Dots represent means. Bars cover the range between percentiles 25 and 75.

Figure 7-22.: Algorithm accuracy (default discretization) on Thermodynamic ODE Model on VSC network.

8. Conclusions

In this work we engage with the field of GRN inference and the assessment of algorithms that perform it. On this basis, we carry out several exercises to assess the performance of a selection of GRN inference algorithms, considering them as estimators in explicit models for gene expression. In terms of theoretical analysis, we provide some simple propositions and numerical examples that characterize how unlikely it is for a network of marginal dependencies (Relevance Network) to accurately approximate a causal graphical model’s skeleton. Furthermore, we conduct a simulation study that sheds light on the strengths and weaknesses of a selection of GRN inference algorithms in estimating the ground truth GRNs associated to particular models of gene expression. From this work, we draw several conclusions and lessons, which we outline below.

On the practical side, we find that the GRN inference algorithms GENIE3 and TIGRESS almost uniformly perform best in our simulation experiments when judging algorithms by AUROC. This is in agreement with other assessments in the literature. Also, we find GENIE3 to be relatively robust to non-linear shapes of regulatory relations and heavy-tailedness of marginal distributions, in comparison to other algorithms reviewed. Despite the above, we also note that, as originally formulated, both GENIE3 and TIGRESS lack built-in thresholding procedures to discretize estimates, which may be problematic. Whether this will imply a practical problem or not in an application of GRN inference with real data sets will depend on the specific research question pursued and the implied costs of different kinds of mistakes. Conceivably, these GRN inference algorithms can be used in exploratory analyses to suggest regulatory relations to be later validated experimentally. In this case, an accurate ranking of gene pairs in terms of the likelihood of regulatory relations will suffice, short of an explicit discretization.

On the other hand, among algorithms with built-in discretization procedures, we observe that the PC algorithm with Gaussian conditional independence testing most accurately recovers causal graphs associated to Gaussian Copula SEMs. Moreover, its performance is robust to the topology of the underlying ground truth GRN, unlike that of other algorithms. We interpret this as reflecting the proven consistency of the PC algorithm for structure learning. Overall, these features make the the PC algorithm appealing, and suggest it can potentially be useful in applications where not only a ranking of possible regulatory relations is desired, but also a binary decision regarding their presence. On the down side, the very

high computational complexity of the PC algorithm can hinder the feasibility of applying it in high-dimensional settings if strict conditions on the sparsity of causal structure do not hold (see [86]).

Our study of Relevance Networks leads to several critical reflections. On one hand, our theoretical observations show that Relevance Networks discretized by testing marginal independence are in general very unlikely to accurately reproduce an arbitrary causal graph under the Gaussian Copula SEM model. This is validated by our simulations, where Relevance Networks converge to the zero-pattern of the mutual information matrix. In hindsight, this is obvious; our analyses simply constitute a corroboration of the idea that “correlation does not equal causation” by the way of showing, numerically and theoretically, to what extent the two notions can differ.

Despite correlation not equaling causation, in our simulation experiments we nonetheless find Relevance Networks to be competitive with other algorithms when judging them by AUROC. This is explained by the fact that, for our selected Gaussian Copula SEMs, estimated mutual information is associated to the presence of edges in the underlying causal graph. This suggests that, beyond the binary approach that simply contrasts marginal independence and marginal dependence, the *magnitude* of mutual information is a useful signal of the underlying causal structure, so that even if “correlation does not equal causation”, at least “correlation is correlated with causation”. This observation raises the question of whether it is possible to characterize which statistical models and causal graph topologies will elicit this pattern in such a way for it to be successfully exploited to infer graphical structure, if only to a certain degree of accuracy.

All the above above being said, our work also gives reason for caution in both using the reviewed GRN inference algorithms and drawing overly general conclusions about their behavior. On one hand, for the case of Gaussian Copula SEMs, performance of our selected GRN inference algorithms is well tracked by graph topology. The implication of this is that, as formulated, these algorithms cannot be counted on to recover a GRN accurately regardless of the specific data generating process it gives rise to, even under the relatively restrictive model of a Gaussian Copula SEM. Therefore, while these algorithms may be useful in many practical scenarios, it seems important to better characterize their domains of applicability.

More broadly, we also observe that most of the results mentioned above are derived from the performance of algorithms in estimating causal graphical models with the specific functional forms of Gaussian Copula SEMs. In this sense, our work also offers some evidence that these results do not carry over to other more complex or realistic settings. In particular, all of our reviewed algorithms perform noticeably worse in recovering the structure of a causal graphical model with non-linear functional forms, and only marginally better in recovering

a ground truth GRN with data from a more realistic, phenomenological, ODE-based model of gene expression. This is a crucial caveat that needs to be taken into account.

The main theme of this project is that GRN inference algorithms should be formulated and viewed as estimators of graph-valued parameters of explicit models of gene regulation. While the practical relevance of GRN inference in applied research may demand the use of heuristics in practice, it is nonetheless important to know under what circumstances an algorithm may or may not be suitable for the task at hand. Tracing out this kind of characterization greatly benefits from a principled approach to inference. Statistics, in its various forms, offers frameworks for reasoning about inference. This leads us to believe that there is great potential for future fruitful exchanges between computational biology and statistics, additional to all those that have already taken place.

A. Appendix: Proofs

A.1. Proofs of Propositions 5 and 6

Let $G(k, p)$ be the Erdos-Renyi random graph with edge probability p on k nodes, whose node set we denote by V . Consider the associated random variables N_{A_4} and N_{C_4} , which count the number of induced 4-paths and 4-cycles in $G(k, p)$, respectively. Then, the probability of obtaining a homogeneous graph as a realization of $G(k, p)$, denoted by $P(H)$, is $P(N_{A_4} = 0, N_{C_4} = 0)$. Note that $P(N_{A_4} = 0) + P(N_{C_4} = 0) - P(H) \leq 1$ and that $P(H) \leq \min\{P(N_{A_4} = 0), P(N_{C_4} = 0)\}$.

Since both N_{A_4} and N_{C_4} are non-negative, the Markov inequality,

$$\begin{aligned} 1 - P(N_{A_4} = 0) &= P(N_{A_4} \geq 1) \leq E(N_{A_4}) \\ 1 - P(N_{C_4} = 0) &= P(N_{C_4} \geq 1) \leq E(N_{C_4}). \end{aligned}$$

On the other hand, by Chebyshev's inequality,

$$\begin{aligned} P(N_{A_4} = 0) &\leq P(|N_{A_4} - E(N_{A_4})| \leq E(N_{A_4})) \leq \frac{\text{Var}(N_{A_4})}{E(N_{A_4})^2} \\ P(N_{C_4} = 0) &\leq P(|N_{C_4} - E(N_{C_4})| \leq E(N_{C_4})) \leq \frac{\text{Var}(N_{C_4})}{E(N_{C_4})^2}. \end{aligned}$$

Altogether, these inequalities imply

$$1 - E(N_{A_4}) - E(N_{C_4}) \leq P(H) \leq \frac{\text{Var}(N_{A_4})}{E(N_{A_4})^2}. \quad (\text{A-1})$$

$E(N_{A_4})$ is given by

$$E(N_{A_4}) = E\left(\sum_{\substack{S \subseteq V \\ |S|=4}} \mathbf{1}_{I(S)=A_4}\right) = \binom{k}{4} \kappa_{A_4} p^3 (1-p)^3,$$

where $\mathbf{1}_{I(S)=A_4}$ is the indicator variable for the event that A_4 is the induced subgraph over the subset of nodes S , $\kappa_{A_4} = (4!)|Aut(A_4)|^{-1} = 12$ is a constant that reflects the number

of graphs isomorphic to A_4 over 4 nodes, and $Aut(A_4)$ is the automorphism group over A_4 . Similarly, for C_4 ,

$$E(N_{C_4}) = \binom{k}{4} \kappa_{C_4} p^4 (1-p)^2,$$

with $\kappa_{C_4} = (4!) |Aut(C_4)|^{-1} = 3$.

On the other hand, the variance of N_{A_4} is given by

$$\begin{aligned} \text{Var}(N_{A_4}) &= \sum_{\substack{S_i \subseteq V \\ |S_i|=4}} \sum_{\substack{S_j \subseteq V \\ |S_j|=4}} \text{Cov}(\mathbf{1}_{I(S_i)=A_4}, \mathbf{1}_{I(S_j)=A_4}) \\ &= \sum_{\substack{S_i \subseteq V \\ |S_i|=4}} \left[\sum_{m=0}^4 \sum_{\substack{S_j \subseteq V \\ |S_j|=4 \\ |S_i \cap S_j|=m}} \text{Cov}(\mathbf{1}_{I(S_i)=A_4}, \mathbf{1}_{I(S_j)=A_4}) \right] \\ &= \sum_{\substack{S_i \subseteq V \\ |S_i|=4}} \left[\sum_{m=0}^4 \sum_{\substack{S_j \subseteq V \\ |S_j|=4 \\ |S_i \cap S_j|=m}} E(\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4}) - E(\mathbf{1}_{I(S_i)=A_4}) E(\mathbf{1}_{I(S_j)=A_4}) \right] \end{aligned}$$

Consider the terms in the sum for each m . To begin, note that for all m and all i, j , $E(\mathbf{1}_{I(S_i)=A_4}) E(\mathbf{1}_{I(S_j)=A_4}) = E(\mathbf{1}_{I(S_i)=A_4})^2$. The expected value $E(\mathbf{1}_{I(S_i)=A_4})$ is equal to $\kappa_{A_4} p^3 (1-p)^3$.

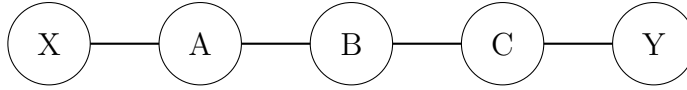
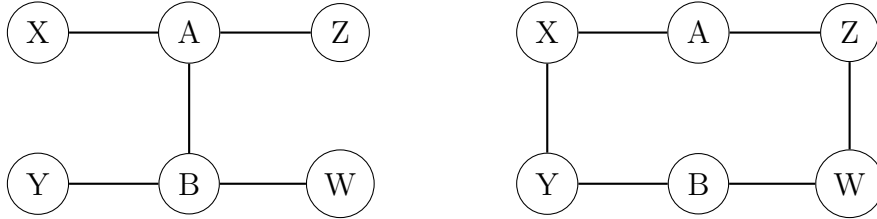
For $m = 0, 1$, the induced subgraphs of S_i and S_j cannot have edges in common, so that $\mathbf{1}_{I(S_i)=A_4}$ and $\mathbf{1}_{I(S_j)=A_4}$ are independent, and $\text{Cov}(\mathbf{1}_{I(S_i)=A_4}, \mathbf{1}_{I(S_j)=A_4}) = 0$. For $m = 4$, necessarily $S_i = S_j$, so that $E(\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4}) = E(\mathbf{1}_{I(S_i)=A_4})$.

For $m = 2, 3$, note that $E(\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4})$ is the probability of observing A_4 as the induced subgraph of both S_i and S_j . For $m = 3$, this is the probability of observing a (not necessarily induced) subgraph over $S_i \cap S_j$ isomorphic to the one depicted in Figure **A-1**, with the restrictions that

- the induced subgraph on $S_1 \cap S_2$ is a 3-path, and
- no additional edges are present among the nodes of S_i nor among the nodes of S_j (although additional edges between nodes in $S_i - S_j$ and nodes $S_j - S_i$ may exist).

This probability is equal to $\kappa_2 p^6 (1-p)^2$, with $\kappa_2 = 3$.

For $m = 2$, $E(\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4})$ is found by characterizing the event $\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4} = 1$ as the union over the two disjoint cases depicted in Figure **A-2**, namely, when the two nodes

Figure A-1.: $m = 3$ case: $S_1 \cap S_2 = \{A, B, C\}$ 

(a) Case 1.

(b) Case 2.

Figure A-2.: $m = 2$ case: $S_1 \cap S_2 = \{A, B\}$

in $S_i \cap S_j$ are either not adjacent or adjacent. In each case, $E(\mathbf{1}_{I(S_i)=A_4} \mathbf{1}_{I(S_j)=A_4})$ is, again, the probability of observing subgraphs isomorphic to the those in A-2a and A-2b, with provisions analogous to those for the $m = 3$ case. These probabilities are equal to $\kappa_{3,0} p^8 (1-p)^3$ and $\kappa_{3,1} p^7 (1-p)^4$, with $\kappa_{3,0} = 1$ and $\kappa_{3,1} = 4$.

Putting all of the above together, the variance of N_{A_4} is

$$\begin{aligned} \text{Var}(N_{A_4}) = & \binom{k}{4} \left[\binom{4}{2} \binom{k-4}{2} (\kappa_2 p^6 (1-p)^2 - \kappa_{A_4}^2 p^6 (1-p)^6) \right. \\ & + \binom{4}{3} \binom{k-4}{3} (\kappa_{3,0} p^8 (1-p)^3 + \kappa_{3,1} p^7 (1-p)^4 - \kappa_{A_4}^2 p^6 (1-p)^6) \\ & \left. + \binom{4}{4} \binom{k-4}{0} (\kappa_{A_4} p^3 (1-p)^3 - \kappa_{A_4}^2 p^6 (1-p)^6) \right] \end{aligned}$$

The expressions given for $\text{Var}(N_{A_4})$ and $E(N_{A_4})$ imply that, for a fixed p , the upper bound in A-1 behaves as k^{-1} as $k \rightarrow \infty$. Thus $\text{Lim}_{k \rightarrow \infty} P(H) = 0$ for any fixed p , proving Proposition 5.

Now, suppose that the random graph model is $G(k, k^{-\alpha})$ for $\alpha > 0$, and consider the bounds on $P(H)$ from A-1. In this case, $E(N_{A_4}) \sim k^{4-3\alpha}$ and $E(N_{C_4}) \sim k^{4-4\alpha}$ as $k \rightarrow \infty$. Also, it can be seen that $\text{Var}(N_{A_4}) \sim k^q$ as $k \rightarrow \infty$, where $q = \max\{7 - 6\alpha, 4 - 3\alpha\}$. From this, it follows directly that $\text{Lim}_{k \rightarrow \infty} P(H) = 0$ for $\alpha > \frac{4}{3}$, which proves Proposition 6.

A.2. Proof of Proposition 8

To prove Proposition 8 we use the following two facts:

Proposition 9. *Let $G = (V, E)$ be a homogeneous undirected graph. Then, for any v_i, v_j in V such that $\{v_i, v_j\} \in E$, it is the case that*

$$\text{adj}(v_j, G) \cup \{v_j\} \subseteq \text{adj}(v_i, G) \cup \{v_i\}$$

or

$$\text{adj}(v_i, G) \cup \{v_i\} \subseteq \text{adj}(v_j, G) \cup \{v_j\}$$

in G .

Proposition 10. *Let $G = (V, E)$ be a complete directed acyclic graph, also known as a transitive tournament. Denote by $\text{sign}(v)$ the indegree (or signature) of $v \in V$ in G . Then, it follows that $\text{sign}(v)$ is a bijection between V and the subset of natural numbers between 0 and $|V| - 1$. Moreover, $(v_i, v_j) \in E$ if and only if*

$$\text{sign}(v_i) < \text{sign}(v_j)$$

Now we prove Proposition 8.

Consider a homogeneous graph $G = (V, E)$ and two directed acyclic orientations of G that follow Hasse perfect vertex elimination orders as defined in Chapter 6, $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$. Denote by $\mathcal{C} = \{C_1, \dots, C_m\}$ the equivalence classes of V induced by the relation R defined by

$$v_i R v_j \iff \text{adj}(v_i, G) \cup \{v_i\} = \text{adj}(v_j, G) \cup \{v_j\}.$$

Denote by $C(v_i)$ the equivalence class in \mathcal{C} to which v_i belongs. Note that for any $C_i \in \mathcal{C}$, its induced subgraph in G is a complete graph, and therefore its induced subgraphs in G_1 and G_2 are transitive tournaments. Denote by $\text{sign}_1(v_i)$ the indegree of v_i in the subgraph induced by $C(v_i)$ in G_1 , and define $\text{sign}_2(v_i)$ analogously for G_2 . Now, consider the following bijection $g : V \rightarrow V$:

$$g(v_i) = v_j \text{ such that } C(v_i) = C(v_j) \text{ and } \text{sign}_1(v_i) = \text{sign}_2(v_j)$$

We prove that g is an isomorphism between G_1 and G_2 .

In the first place, consider $(v_i, v_j) \in E_1$ such that $C(v_i) \neq C(v_j)$. Since G is homogeneous, it follows from Proposition 9 and the fact that G_1 follows a Hasse perfect elimination order that

$$\text{adj}(v_j, G) \cup \{v_j\} \subsetneq \text{adj}(v_i, G) \cup \{v_i\}.$$

Furthermore, by definition of g , it follows that $C(v_i) = C(g(v_i))$ and $C(v_j) = C(g(v_j))$. Thus, $adj(v_j, G) \cup \{v_j\} = adj(g(v_j), G) \cup \{g(v_j)\}$ and $adj(v_i, G) \cup \{v_i\} = adj(g(v_i), G) \cup \{g(v_i)\}$, and therefore

$$adj(g(v_j), G) \cup \{g(v_j)\} \subsetneq adj(g(v_i), G) \cup \{g(v_i)\}$$

Since G_2 also follows a Hasse perfect elimination order, $(g(v_i), g(v_j)) \in E_2$.

Now, consider $(v_i, v_j) \in E_1$ such that $C(v_i) = C(v_j)$. Since $C(v_i)$ induces a transitive tournament in G_1 , it follows from Proposition 10 that $\text{sign}_1(v_i) < \text{sign}_1(v_j)$. Furthermore, by definition of g , we have that $\text{sign}_1(v_i) = \text{sign}_2(g(v_i))$ and $\text{sign}_1(v_j) = \text{sign}_2(g(v_j))$. It then follows that $(g(v_i), g(v_j)) \in E_2$.

An identical line of reasoning can be used to prove the converse, that if $(g(v_i), g(v_j)) \in E_2$, then $(v_i, v_j) \in E_1$. Altogether, this proves that g is an isomorphism.

For the second part of Proposition 8, consider the number of acyclic orientations G_k of G that follow Hasse perfect vertex elimination orders. We observe that for any $\{v_i, v_j\} \in E$ such that $C(v_i) \neq C(v_j)$, the direction of the corresponding directed edge in any orientation G_k of G depends exclusively on whether $C(v_i) \subsetneq C(v_j)$ or $C(v_j) \subsetneq C(v_i)$. Thus, all orientations G_k coincide in the directions of edges of this kind. On the other hand, undirected edges between nodes v_i, v_j in the same equivalence class $C(v_i) = C(v_j)$ are directed in G_k according to whether $\text{sign}_k(v_i) < \text{sign}_k(v_j)$ or viceversa. Therefore, choosing $\text{sign}_k(v_i)$ for each node v_i in each equivalence class $C \in \mathcal{C}$ determines all edge orientations in G_k . This can be done in

$$O_l(G) = \prod_{C \in \mathcal{C}} |C|!$$

ways.

Bibliography

- [1] Siddhartha Mukherjee. *The gene: an intimate history*. Scribner, 2016.
- [2] Greg Elgar and Tanya Vavouri. “Tuning in to the signals: noncoding sequence conservation in vertebrate genomes”. In: *Trends in genetics* 24.7 (2008), pp. 344–352.
- [3] ENCODE Project Consortium et al. “An integrated encyclopedia of DNA elements in the human genome”. In: *Nature* 489.7414 (2012), pp. 57–74.
- [4] Vân Anh Huynh-Thu and Guido Sanguinetti. “Gene Regulatory Network Inference: An Introductory Survey”. In: *Gene Regulatory Networks: Methods and Protocols*. Ed. by Guido Sanguinetti and Vân Anh Huynh-Thu. New York, NY: Springer New York, 2019, pp. 1–23. ISBN: 978-1-4939-8882-2. DOI: 10.1007/978-1-4939-8882-2_1. URL: https://doi.org/10.1007/978-1-4939-8882-2_1.
- [5] Terry Brown. *Understanding a Genome Sequence*. Wiley-Liss, 2002. Chap. 7.
- [6] Louis M. Straudt. *Mouse cDNA Microarray*. National Cancer Institute, 2001. URL: <https://visualsonline.cancer.gov/details.cfm?imageid=1849>.
- [7] Jessica C Mar. “The rise of the distributions: why non-normality is important for understanding the transcriptome and beyond”. In: *Biophysical reviews* 11.1 (2019), pp. 89–94.
- [8] Albert-Laszlo Barabasi and Zoltan N Oltvai. “Network biology: understanding the cell’s functional organization”. In: *Nature reviews genetics* 5.2 (2004), pp. 101–113.
- [9] Anna D Broido and Aaron Clauset. “Scale-free networks are rare”. In: *Nature communications* 10.1 (2019), pp. 1–10.
- [10] Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. “Power-law distributions in empirical data”. In: *SIAM review* 51.4 (2009), pp. 661–703.
- [11] Romualdo Pastor-Satorras, Eric Smith, and Ricard V Solé. “Evolving protein interaction networks through gene duplication”. In: *Journal of Theoretical biology* 222.2 (2003), pp. 199–210.
- [12] Robert D Leclerc. “Survival of the sparsest: robust gene networks are parsimonious”. In: *Molecular systems biology* 4.1 (2008), p. 213.
- [13] Z Burda et al. “Motifs emerge from function in model gene regulatory networks”. In: *Proceedings of the National Academy of Sciences* 108.42 (2011), pp. 17263–17268.

- [14] Réka Albert, Hawoong Jeong, and Albert-László Barabási. “Error and attack tolerance of complex networks”. In: *nature* 406.6794 (2000), pp. 378–382.
- [15] Momoko Otsuka and Sho Tsugawa. “Robustness of network attack strategies against node sampling and link errors”. In: *Plos one* 14.9 (2019), e0221885.
- [16] Reuven Cohen and Shlomo Havlin. “Scale-free networks are ultrasmall”. In: *Physical review letters* 90.5 (2003), p. 058701.
- [17] Ulrike Von Luxburg. “A tutorial on spectral clustering”. In: *Statistics and computing* 17.4 (2007), pp. 395–416.
- [18] M. E. J. Newman. “Modularity and community structure in networks”. In: *Proceedings of the National Academy of Sciences* 103.23 (2006), pp. 8577–8582. ISSN: 0027-8424. DOI: 10.1073/pnas.0601602103. eprint: <https://www.pnas.org/content/103/23/8577.full.pdf>. URL: <https://www.pnas.org/content/103/23/8577>.
- [19] Vincent D Blondel et al. “Fast unfolding of communities in large networks”. In: *Journal of statistical mechanics: theory and experiment* 2008.10 (2008), P10008.
- [20] Roger B Nelsen. *An introduction to copulas*. Springer Science & Business Media, 2007.
- [21] Steffen L Lauritzen. *Graphical models*. Vol. 17. Clarendon Press, 1996.
- [22] Rajen D Shah, Jonas Peters, et al. “The hardness of conditional independence testing and the generalised covariance measure”. In: *Annals of Statistics* 48.3 (2020), pp. 1514–1538.
- [23] Bin Zhang and Steve Horvath. “A general framework for weighted gene co-expression network analysis”. In: *Statistical applications in genetics and molecular biology* 4.1 (2005).
- [24] Atul J Butte and Isaac S. Kohane. “Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements.” In: *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* (2000), pp. 418–29.
- [25] Kshitij Khare and Bala Rajaratnam. “Sparse matrix decompositions and graph characterizations”. In: *Linear algebra and its applications* 437.3 (2012), pp. 932–947.
- [26] Adam A. Margolin et al. “ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context”. In: *BMC Bioinformatics* 7 (2006), S7 –S7.
- [27] Patrick E Meyer et al. “Information-theoretic inference of large transcriptional regulatory networks”. In: *EURASIP journal on bioinformatics and systems biology* ().

- [28] Xiujun Zhang et al. “NARROMI: a noise and redundancy reduction technique improves accuracy of gene regulatory network inference”. In: *Bioinformatics* 29.1 ().
- [29] Nicolai Meinshausen, Peter Bühlmann, et al. “High-dimensional graphs and variable selection with the lasso”. In: *The annals of statistics* 34.3 (2006), pp. 1436–1462.
- [30] Anne-Claire Haury et al. “TIGRESS: trustful inference of gene regulation using stability selection”. In: *BMC systems biology* 6.1 (2012), p. 145.
- [31] Bradley Efron et al. “Least angle regression”. In: *The Annals of statistics* 32.2 (2004), pp. 407–499.
- [32] Vân Anh Huynh-Thu et al. “Inferring regulatory networks from expression data using tree-based methods”. In: *PloS one* 5.9 (2010), pp. 1–10.
- [33] Onureena Banerjee, Laurent El Ghaoui, and Alexandre d’Aspremont. “Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data”. In: *Journal of Machine learning research* 9.Mar (2008), pp. 485–516.
- [34] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. “Sparse inverse covariance estimation with the graphical lasso”. In: *Biostatistics* 9.3 (2008), pp. 432–441.
- [35] Peter Spirtes, Clark N Glymour, and Scheines. *Causation, prediction, and search*. 2nd ed. The MIT Press, 2000.
- [36] Xiujun Zhang et al. “Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information”. In: *Bioinformatics* 28.1 (2012), pp. 98–104.
- [37] James M Robins et al. “Uniform consistency in causal inference”. In: *Biometrika* 90.3 (2003), pp. 491–515.
- [38] Diego Colombo and Marloes H Maathuis. “Order-independent constraint-based causal structure learning”. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 3741–3782.
- [39] Jeremiah J Faith et al. “Large-Scale Mapping and Validation of Escherichia coli Transcriptional Regulation from a Compendium of Expression Profiles”. In: *PLoS Biology* 5.1 (Jan. 2007), pp. 1–13. DOI: 10.1371/journal.pbio.0050008. URL: <https://doi.org/10.1371/journal.pbio.0050008>.
- [40] Daniel Marbach et al. “Revealing strengths and weaknesses of methods for gene network inference”. In: *Proceedings of the national academy of sciences* 107.14 (2010), pp. 6286–6291.
- [41] Thomas Schaffter, Daniel Marbach, and Dario Floreano. “GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods”. In: *Bioinformatics* 27.16 (2011), pp. 2263–2270.

- [42] Daniel Marbach et al. “Wisdom of crowds for robust gene network inference”. In: *Nature methods* 9.8 (2012), p. 796.
- [43] Adriano V Werhli, Marco Grzegorzczuk, and Dirk Husmeier. “Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models and Bayesian networks”. In: *Bioinformatics* 22.20 (2006), pp. 2523–2531.
- [44] Wenbin Guo et al. “Evaluation and improvement of the regulatory inference for large co-expression networks with limited sample size”. In: *BMC systems biology* 11.1 (2017), p. 62.
- [45] Jonathan Ish-Horowicz and John Reid. “Mutual information estimation for transcriptional regulatory network inference”. In: *bioRxiv* (2017). DOI: 10.1101/132647. eprint: <https://www.biorxiv.org/content/early/2017/07/26/132647.full.pdf>. URL: <https://www.biorxiv.org/content/early/2017/07/26/132647>.
- [46] Shuonan Chen and Jessica C Mar. “Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data”. In: *BMC bioinformatics* 19.1 (2018), p. 232.
- [47] Sisi Ma et al. “De-Novo Learning of Genome-Scale Regulatory Networks in *S. cerevisiae*”. In: *PLOS ONE* 9.9 (Sept. 2014), pp. 1–20. DOI: 10.1371/journal.pone.0106479. URL: <https://doi.org/10.1371/journal.pone.0106479>.
- [48] Jeffrey D Allen et al. “Comparing statistical methods for constructing large scale gene networks”. In: *PloS one* 7.1 (2012).
- [49] Aditya Pratapa et al. “Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data”. In: *Nature Methods* (2020), pp. 1–8.
- [50] Judea Pearl. *Causality*. Cambridge University Press, 2009. DOI: 10.1017/CB09780511803161.
- [51] Diego Colombo et al. “Learning high-dimensional directed acyclic graphs with latent and selection variables”. In: *The Annals of Statistics* (2012), pp. 294–321.
- [52] Stephan Bongers et al. “Theoretical Aspects of Cyclic Structural Causal Models”. In: *arXiv.org preprint arXiv:1611.06221v2 [stat.ME]* (Aug. 2018). URL: <https://arxiv.org/abs/1611.06221v2>.
- [53] Judea Pearl, Thomas Verma, et al. “A theory of inferred causation.” In: *KR* 91 (1991), pp. 441–452.
- [54] Nancy Cartwright. *Hunting causes and using them: Approaches in philosophy and economics*. Cambridge University Press, 2007.

- [55] Holly Andersen. “When to expect violations of causal faithfulness and why it matters”. In: *Philosophy of Science* 80.5 (2013), pp. 672–683.
- [56] Juliane Schäfer and Korbinian Strimmer. “A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics”. In: *Statistical applications in genetics and molecular biology* 4.1 (2005).
- [57] Hiroyuki Toh and Katsuhisa Horimoto. “Inference of a genetic network by a combined approach of cluster analysis and graphical Gaussian modeling”. In: *Bioinformatics* 18.2 (2002), pp. 287–297.
- [58] Oliver J Maclaren and Ruanui Nicholson. “What can be estimated? Identifiability, estimability, causal inference and ill-posed inverse problems”. In: *arXiv preprint arXiv:1904.02826* (2019).
- [59] Sunyong Kim, Seiya Imoto, and Satoru Miyano. “Dynamic Bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data”. In: *Biosystems* 75.1-3 (2004), pp. 57–65.
- [60] Bruno-Edouard Perrin et al. “Gene networks inference using dynamic Bayesian networks”. In: *Bioinformatics* 19.suppl.2 (2003), pp. ii138–ii148.
- [61] Stephan Bongers and Joris M Mooij. “From random differential equations to structural causal models: The stochastic case”. In: *arXiv preprint arXiv:1803.08784* (2018).
- [62] Alexander Sokol and Niels Richard Hansen. “Causal interpretation of stochastic differential equations”. In: *Electronic Journal of Probability* 19.100 (2014), pp. 1–24.
- [63] Paul K Rubenstein et al. “From deterministic ODEs to dynamic structural causal models”. In: *arXiv preprint arXiv:1608.08028* (2016).
- [64] Gary K Ackers, Alexander D Johnson, and Madeline A Shea. “Quantitative model for gene regulation by lambda phage repressor”. In: *Proceedings of the National Academy of Sciences* 79.4 (1982), pp. 1129–1133.
- [65] Lacramioara Bintu et al. “Transcriptional regulation by the numbers: models”. In: *Current opinion in genetics & development* 15.2 (2005), pp. 116–124.
- [66] Arwen Meister et al. “Learning a nonlinear dynamical system model of gene regulation: A perturbed steady-state approach”. In: *The Annals of Applied Statistics* 7.3 (2013), pp. 1311–1333.
- [67] William Chad Young, Ka Yee Yeung, and Adrian E Raftery. “Identifying dynamical time series model parameters from equilibrium samples, with application to gene regulatory networks”. In: *Statistical Modelling* 19.4 (2019), pp. 444–465.
- [68] Lacramioara Bintu et al. “Transcriptional regulation by the numbers: applications”. In: *Current opinion in genetics & development* 15.2 (2005), pp. 125–135.

- [69] Ruifei Cui et al. “Learning the Causal Structure of Copula Models with Latent Variables.” In: *UAI*. 2018, pp. 188–197.
- [70] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020. URL: <https://www.R-project.org/>.
- [71] Ioannis Tsamardinos and Giorgos Borboudakis. “Permutation Testing Improves Bayesian Network Learning”. In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by José Luis Balcázar et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 322–337. ISBN: 978-3-642-15939-8.
- [72] Jonathan Ish-Horowicz. *fastGeneMI: A Suite of Mutual Information Estimators used for Gene Regulatory Network Inference from Microarray Expression Data*. R package version 1.0. 2018.
- [73] Gabriele Sales and Chiara Romualdi. *parmigene: Parallel Mutual Information estimation for Gene Network reconstruction*. R package version 1.0.2. 2012. URL: <https://CRAN.R-project.org/package=parmigene>.
- [74] Patrick E. Meyer, Frederic Lafitte, and Gianluca Bontempi. “MINET: An open source R/Bioconductor Package for Mutual Information based Network Inference”. In: *BMC Bioinformatics* 9 (2008). URL: <http://www.biomedcentral.com/1471-2105/9/461>.
- [75] Xiujun Zhang. *Website of Xiujun Zhang*. URL: <https://sites.google.com/site/xiujunzhangcsb/software/narromi>.
- [76] Marco Scutari. “Learning Bayesian Networks with the bnlearn R Package”. In: *Journal of Statistical Software* 35.3 (2010), pp. 1–22. DOI: 10.18637/jss.v035.i03.
- [77] Osiris Ríos et al. “A Boolean network model of human gonadal sex determination”. In: *Theoretical Biology and Medical Modelling* 12.1 (2015), pp. 1–18.
- [78] Jan Krumsiek et al. “Hierarchical differentiation of myeloid progenitors is encoded in the transcription factor network”. In: *PloS one* 6.8 (2011), e22649.
- [79] Anna Lovrics et al. “Boolean modelling reveals new regulatory connections between transcription factors orchestrating the development of the ventral spinal cord”. In: *PloS one* 9.11 (2014), e111430.
- [80] Clare E Giacomantonio and Geoffrey J Goodhill. “A Boolean model of the gene regulatory network underlying Mammalian cortical area development”. In: *PLoS Comput Biol* 6.9 (2010), e1000936.
- [81] Daniel Marbach et al. “Generating realistic in silico gene networks for performance assessment of reverse engineering methods”. In: *Journal of computational biology* 16.2 (2009), pp. 229–239.
- [82] Brendan McKay. *Graphs*. URL: <http://users.cecs.anu.edu.au/~bdm/data/graphs.html>.

-
- [83] Richard P Stanley. “Acyclic orientations of graphs”. In: *Discrete Mathematics* 5.2 (1973), pp. 171–178.
- [84] P Hanlon. “The chromatic polynomial of an unlabeled graph”. In: *Journal of Combinatorial Theory, Series B* 38.3 (1985), pp. 226–239.
- [85] Jeffrey M Wooldridge. *Econometric analysis of cross section and panel data*. MIT press, 2010.
- [86] Markus Kalisch and Peter Bühlmann. “Estimating high-dimensional directed acyclic graphs with the PC-algorithm”. In: *Journal of Machine Learning Research* 8.Mar (2007), pp. 613–636.