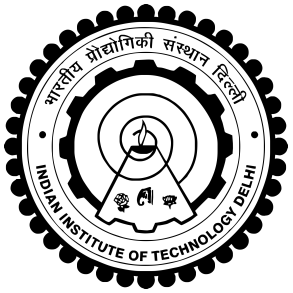
Causal Computational Models

for

Gene Regulatory Networks



Department of Computer Science and Engineering

Indian Institute of Technology Delhi

Sahil Loomba & Parul Jain

Supervised by Dr. Parag Singla & Dr. Sumeet Agarwal

July 2016

Abstract

Gene Regulatory Networks (GRNs) hold the key to understanding and solving many problems in biological sciences, with critical applications in medicine and therapeutics. However, discovering GRNs in the laboratory is a cumbersome and tricky affair, since the number of genes and interactions, say in a mammalian cell, are very large. We aim to discover these GRNs computationally, by using gene expression levels as a "time-series" dataset. We research and employ techniques from probability and information theory, theory of dynamical systems, and graph structure estimation, to establish causal relations between genes, on synthetic datasets. Furthermore, we suggest methods for global estimation of gene networks. Therefore, narrowing the space of genetic interactions to be looked at when discovering these GRNs in the lab.

Keywords: *gene regulation, network inference, causality, information theory, dynamical systems, random walk models*

Contents

Chapter 1

Introduction

A single biological cell, is the basic independent building block of all lifeforms. Colloquially speaking, a cell is essentially a "bag" or membrane, enclosing a number of "chemicals" which govern the lower-level working (birth, survival, death) of the cell. These chemicals include mostly water, and proteins. Proteins themselves are created by a broadly two-step process of DNA transcription and translation. The DNA, as is common knowledge, is the storehouse of cellular information. Those sections of the DNA which code for important biomolecules such as proteins, are called **genes**. When a gene gets "activated", the process of transcription and translation ensues, leading to production of the relevant protein, in certain quantities and at certain points in time depending on the level of gene activation. For a graphical representation, see Figure 1.1. Genes themselves get activated through their interaction with proteins called **transcription factors**. This regulation can be either activating, or inhibitory. Some genes also undergo self-regulation, that is they regulate their own expression. (Although we ignore self-regulation for the purpose of our analysis.) Thus eventually, the expression of a gene of importance (GOI) is controlled

by a usually large directed network of cascading gene interactions. It becomes very important to discover this network for the GOI not only so that we can model gene expression, but also so that we can better understand which regulatory pathways need to be targeted for therapeutics.

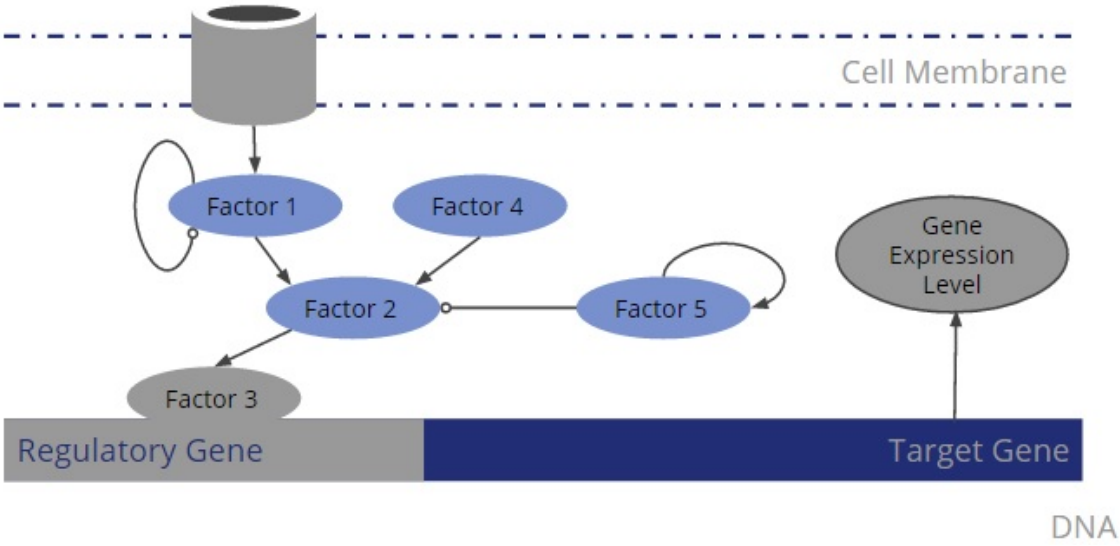


Figure 1.1: A cartoon representation of a Gene Regulatory Network. Each of the factors can be represented as nodes in a directed graph $G = \{V, E\}$

Since gene expression is temporal data, one could interpret the expression of genes connected in the GRN to be causally related signals. That is, iff for two genes X and Y a directed regulatory edge exists from X to Y, then there exists a causal relation $X \rightarrow Y$. The difficult problem of GRN discovery has thus been translated to another (albeit still difficult) problem of establishing causality between signals. We attempt to establish pairwise causality using various metrics, and thus arrive at conclusions as to which methods are favourable for this problem, and if at all establishing pairwise causality is sufficient for GRN discovery. We also attempt to extend our ideas from the pairwise causal to a global network estimation level.

Reliable techniques to predict GRNs can help us understand physiological dif-

ferences and predict/diagnose health issues. The biomarkers identified by such a method could be used to detect early onset of many diseases (Huntington, Alzheimer's etc). One could also imagine applications in therapeutics, since modelling of gene regulatory pathways and knockouts can allow us to design better-targeted drugs. Since the development of such a technique holds key to understanding multiple biological issues, there have been many attempts to solve the problem. Some of these are discussed in the sections that follow.

Chapter 2

Previous Work

As discussed, the problem of GRN discovery is not a new one. Some of the current literature which has inspired our work is briefly discussed in the subsections that follow.

2.1 GRNs and their Applications

This paper is the motivation for our work and provides a general discussion and perspective on gene regulatory networks [1]. GRNs represent statistically significant predictions of molecular interactions obtained from large-scale data. In other words, for genomes as large as that of humans, GRNs are of tremendous help in narrowing down **potential interactions** for which statistical support is available. Using prior knowledge about "partial" gene regulatory networks inferred from such observational data, we can have controlled experiments by establishing conditions that enhance molecular target processes to improve the signal strength. Since they consider the interaction structure between individual genes explicitly, GRNs could also be used as biomarkers, for diagnostic, predictive or prognostic purposes. As the number of

discovered GRNs will grow, the comparison of networks will allow us to learn about interaction changes across different physiological or disease conditions and enrich our biological and biomedical understanding of various phenotypes. Most importantly, GRNs will enable production of personalised medicine as condition specific GRNs are closer to the phenotype than genetic or epigenetic markers.

After detailed analysis of the various methods used for GRN detection, the paper argues that results of such technical comparisons depend crucially on the **studied conditions**, including: type of data (simulated or real), size of the network, number of samples, amount of noise, experimental design (observational, experimental, interventional), type of the underlying interaction structure (scale-free, random, small-world) and error measure (global, local), among others. Also, though experimental research work and biological databases used as ground truths have a huge number of recorded interactions, many of them might not be relevant to the biological conditions under investigation, which could affect truth assessment of the inferred network.

Keeping this in mind, parameters defining the data, used in the experiments for this project, were recorded in detail and underwent multiple perturbations so that the observations are not biased and results can be generalised. In addition, the experiments are performed on synthetic data for accurate evaluation of the performance with methods suggested in our work.

2.2 Discovering GRNs using Pairwise GC

This paper uses Granger Causality as the method of inference, but instead of using full model GC which considers all possible combinations, it considers pairs of genes

at a time and only includes significant ones in the inferred network [2]. GC (see Section 4.3) is a "statistical concept of causality based on prediction" proposed by the Nobel laureate Clive Granger in Economics. However, pairwise GC produces large number of false edges. Thus, the first part of the paper is devoted to detecting the issues with pairwise GC and then, using techniques to reduce the number of **spurious edges**. The paper uses various methods to reduce the number of false edges. It first compares the performance on synthetic networks with and without the correction methods to establish their effectiveness. Then, it uses the method on human HeLa dataset.

It uses a multivariate autoregressive model for data regression and uses F-statistic for significance. If the VAR model does not adapt to the data well, correlations between the variables cannot be captured properly and the GC detected is not reliable. There, model validation is done by checking model consistency, adjusted RSS and Durbin–Watson (whiteness) test. The model order is calculated through AIC or BIC and the network is inferred using both. Since BIC suits better where the data size is large, AIC was observed to produce better results on the HeLa data-set, which had a largest size of 47 only. Also, since there are ${}^nC_2$ pairs of genes for an n-gene network, there are as many p-values and hence the tests need correction, for which Bonferroni correction and Benjamini–Hochberg False Discovery Rate (FDR) controlling procedure are used.

On similar lines, our project uses pairwise GC as a method of inference. Since the size of time-series data is large (500, 1000), BIC is used for lag estimation. The paper concludes that though the accuracy of the method cannot be estimated for real data sets, the techniques used could definitely make GC a strong contender

for causality inference since correlation-based methods (such as Pearson correlation coefficient (PCC) and mutual information) cannot give direction of causality.

2.3 Information Theory for Causality Detection

This paper aims to provide a detailed overview of information theoretic approaches for measuring causal influence in multivariate time series, focusing on diverse approaches to entropy and mutual information estimation [3]. Natural phenomena emerge from complex systems which are composed of modules, interacting in a complex, often non-linear manner. The behavior of the system cannot be explained by a linear combination of the parts. Thanks to various databases, we now have big data representing the temporal dynamics of possibly interacting variables, but the methods which could make sense of the data are still being researched and developed. The paper suggests that information theory techniques are crucial to understanding such systems as they hold the key to causality detection which will help us understand the basic network structure of these system.

It explains causality under the Granger-Weiner framework and then extends it to non-linear systems, based on the information theoretic formulation of **transfer entropy**. First, it lists the prerequisites for using entropy or mutual information for measurement, such as continuity, differentiability and boundedness, among others. Describing all the information theory methods in detail, it discusses the conditions under which they will be best suited. Finally, it discusses the idea of Granger Causality in detail along with its non-linear analogs. Hence, this paper provided a reference ground for selection of methods for information theory methods.

It argues that for a good entropy estimator, the condition of consistency seems

to be important so that the method is applicable for the problem for a wide range of experimental conditions. This is essential as in case of biological systems, there are multiple genetic and epigenetic factors in play which control the data, making it biased in one way or another. Since truly generalised data is not feasible, consistency of method across conditions is desirable. With this, it agrees that conditional mutual information (or transfer entropy) is crucial to causality detection. Hence, we have used transfer entropy, suggested as the most promising method by the paper, along with mutual information for comparison of performance.

2.4 ARACNE

Considered to be state-of-the-art for inferring Gene Regulatory Networks, an Algorithm for the Reconstruction of Gene Regulatory Networks, or simply ARACNE, works on the metric of pairwise mutual information [4]. It defines an edge in the graph as an irreducible statistical dependency between gene expression profiles that cannot be explained as an artifact of other statistical dependencies in the network. It further propounds that there is no universally accepted definition of statistical dependencies in the multivariate setting.

Taking from literature on Markov networks, it represents a GRN as Markov network, and then defines clique potentials only till a pairwise level, ignoring higher order analysis. It then identifies candidate interactions by estimating pairwise gene expression profile **mutual information**. Interactions are then filtered in two steps. First, using an appropriate thresholding, computed for a specific p-value, in the null hypothesis of two independent genes. Second, indirect interactions are detected using the **Data Processing Inequality** (DPI) which says that if two genes g_1 and

g_3 can interact with each other only through a path via gene g_2 (that is, $g_1 \leftrightarrow ... \leftrightarrow g_2 \leftrightarrow ... \leftrightarrow g_3$), then $I(g_1, g_3) \leq min[I(g_1, g_2), I(g_2, g_3)]$.

The paper further claims (with proof) that if MIs can be estimated with no errors, then ARACNE reconstructs the underlying interaction network exactly, provided this network is a tree and has only pairwise interactions. It doesn't however comment on the validity of the assumption of correctness of estimation, in the first place. Thus, the real challenge of information theoretic measures lies in estimating probabilities, in the limit of poverty of data.

Chapter 3

Datasets

The ultimate aim of this project requires us to work on real datasets for modelling and testing. However, most real datasets are very large in the number of genes and interactions. Therefore, we decided to start with smaller synthetic datasets.

3.1 Steady State Data

We generated this data using the MATLAB Toolbox called **SysGenSIM** [6]. The toolbox first generates a GRN topology, depending on: (a) number of genes, (b) average degree (c) a topology model (random, scale-free, modular, etc.) and (d) length of time series (or rather, number of perturbation conditions, since this isn't exactly time-series data). We take these to be fixed parameters with respect to an experiment run, and use the scale-free topology. (See Figure 3.1 for an example network topology.) Then, it uses a non-linear ODE model given below, where the first term accounts for transcription, and the second term accounts for degradation.

$$\frac{\mathrm{d}\,G_g}{\mathrm{d}\,t} = Z_g^c \cdot V_g \cdot \theta_g^{syn} \cdot \prod_k \left(1 + A_{k,g} \frac{G_k^{h_{k,g}}}{G_k^{h_{k,g}} + (K_{k,g}/Z_k^t)^{h_{k,g}}}\right) - \lambda_g \cdot \theta_g^{deg} \cdot G_g$$

Where G_g is mRNA concentration of gene g which is the gene of interest, V_g is its basal transcription rate, and λ_g is the degradation rate constant. The G_k are expression levels of genes which have directed edges into node i.e., the genes that affect the expression of gene G_g. $K_{k,g}$ is the interaction strength, a Michaelis constant (non-negative, denoting how strongly does gene G_k affect G_g), $h_{k,g}$ is a cooperativity coefficient (and controls the extent of non-linearity of the interaction), and $A_{k,g}$ is an element of matrix A encoding the signed network structure representing the kind of effect (-1 for inhibitor, 1 for activator, 0 for no effect). The biological variance parameters θ_g^{syn} and θ_g^{deg} represent non-genetic additional biological noise in the transcription and degradation rates, respectively, and are sampled from a normal distribution with unit mean and user specified standard deviations. This is to ensure that the data is closer to what is observed in the real world with respect to error reporting. Z_g^c and Z_k^t are parameters that incorporate effects of DNA variants and represent the effect on the transcription rate of G_k when the variant is in its own promoter region or in the coding region of its regulatory gene respectively.

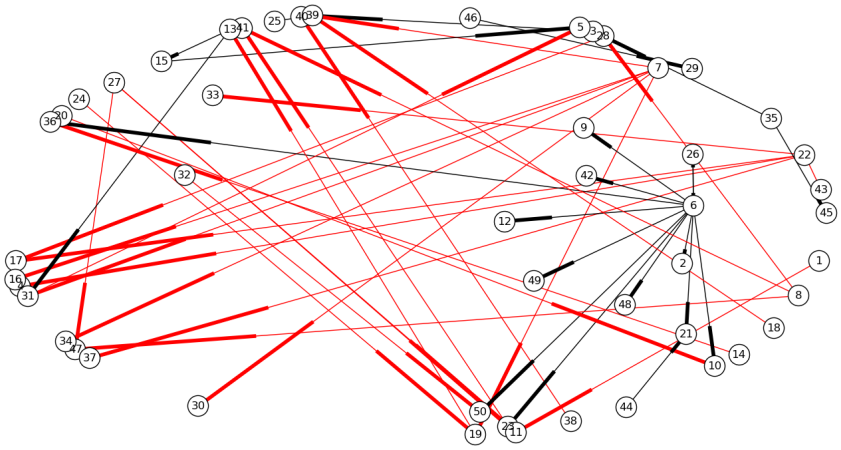
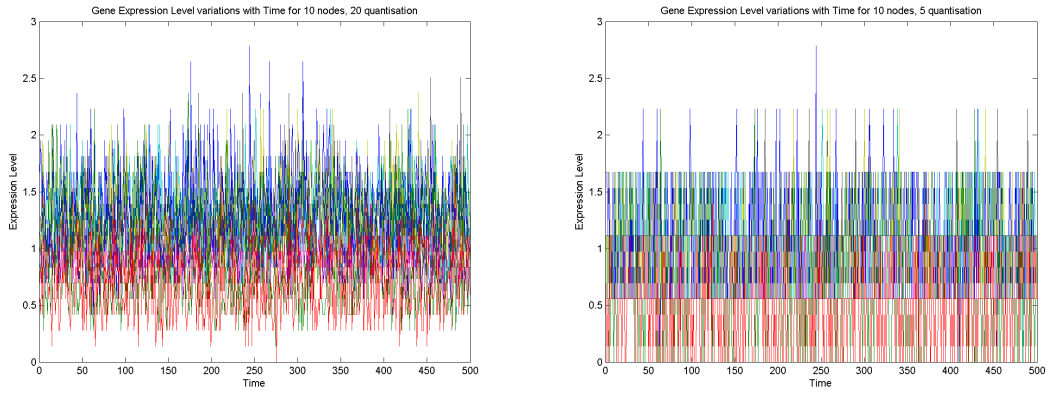


Figure 3.1: An example network topology with 50 nodes. Notice the scale-free distribution followed by the network: few nodes with high degree, many nodes with low degree.

The data output of SysGenSIM is a matrix of size $N \times T$, for the expression of N genes under T independent conditions. We have experimented with these datasets for $N \in \{10, 20, 50, 100\}$ and for $T \in \{500, 1000\}$. Moreover, every dataset was quantised to different levels of quantisation, $Q \in \{2, 5, 10, 20\}$ for the information theoretic techniques (namely Mutual Information and Transfer Entropy). See Figure 3.2 for a visualisation of the effect of quantisation on signal space. Also, as is standard practice, the data was normalised to zero mean and unit variance.



(a) $Q = 20$ (b) $Q = 5$

Figure 3.2: An example of signal quantisation for GRN with $N = 10$

3.2 Time Series Data

Since we need to model causal relationships, it is more sensible to use time-series data, because we need some temporal ordering of the individual dimensions of the signal vector. Thus, we have lifted datasets from the DREAM4 challenge, as also used in [5]. Since this dataset has been previously used in a paper on benchmarking GRN discovery methods, this allows us to compare those results with our own, as covered in Section 7.

DREAM4 datasets are also synthetic datasets, and have used the GeneNetWeaver software for generating their data. It consists of 5 networks each of sizes $N = 10$ and $N = 100$. A single experimental run on a network renders one time-series matrix of size $N \times T$, where $T = 21$ timesteps in this case. Additionally, multiple experimental runs have been done for every network, rendering a new time-series matrix each time. There were 5 and 10 runs respectively for the 10 and 100 sized networks. The dataset also contains the ground truth for every network. These

networks are sparse and possess the scale-free property. A time-series of just length 21 faces a fierce problem of scarcity of data. Therefore to reduce this issue, we use various methods to aggregate the data across all the experimental runs, instead of simply averaging the metric values obtained after every run. We briefly describe the method used for every pairwise metric technique, below:

- **Correlation and Mutual Information:** Every time-series was shifted according to the lag value τ. These shifted series were then simply stacked one after another to create a single time-series matrix before calculating the metric.

- **Granger Causality:** The time-series were stacked together as usual. Those points at the beginning of every time-series, whose regression terms included data from the previous time-series (first '$\tau + 1$' points), were ignored before carrying out the regression.

- **Transfer Entropy:** The conditional probability distributions were separately estimated for each time-series, and then they were all averaged, before being used for calculating entropy.

- **Convergent Cross Map:** The shadow manifold embedding vectors were separately found for each time-series, which were then augmented to form the aggregate shadow manifold, on which the nearest-neighbour algorithm was applied in the usual fashion.

Chapter 4

Pairwise Metrics

One of the two core approaches in our work is to look at signals in a pairwise fashion, and then use a metric which works as a thresholding function from "non-causal" to "causal" relation. Here, we work with primarily six methods (or *metrics*), first four of which treat gene expression levels (time-series signal) for a given gene G, as a random variable X, coming from an underlying probability distribution $P(X)$. For a summary of methods, see Figure 4.2.

4.1 Correlation

Correlations are used to measure the notion of co-dependence between two signals. More specifically, a linear correlation can be used to find existence of simple linear relationships between these signals. We use the unsigned value of the Pearson Correlation Coefficient (PCC) $\rho(X, Y)$ to quantify this linear correlation. In its most concise statistical form, PCC can be expressed as:

$$\rho(X, Y) = \frac{cov(X, Y)}{\sigma_X \sigma_Y}$$

Where $cov(X, Y)$ refers to the covariance between signals X and Y, and σ_X refers to the variance in signal X. This can be expanded further to express them in terms of probability distributions which govern the two signals, which have been assumed to be random variables. For visual acuity, we write only the unstandardised PCC, i.e. the covariance, in terms of probabilities.

$$cov(X, Y) = E[XY] - E[X]E[Y]$$

$$cov(X, Y) = \sum_{x,y} xyP(x, y) - \sum_{x} xP(x) \sum_{y} yP(y)$$

$$cov(X, Y) = \sum_{x,y} \big(P(x, y) - P(x)P(y)\big)xy \tag{4.1}$$

Some key mathematical properties of PCC are:

- It is symmetric in X and Y, which does not capture a sense of direction in causality. However, we make use of (maximum) shifted correlation, and on the basis of the sign of shift (positive/negative), we decide the direction of causality.

- Its value ranges between -1 and 1, with values close to zero implying low correlation and those close to ± 1 implying high (positive/negative) correlation.

- It captures linear relations. The square of the PCC is same as the coefficient of determination r^2 when trying to do a linear regression between the two signals. Thus, a value of zero means a lack of *linear* relationship.

- Clearly, there is no sense of predictability or causation, but mere co-dependence. Hence the common but appropriate adage "correlation does not imply causation".

4.2 Mutual Information

In the purview of statistical theory, Mutual Information or MI captures non-linear correlations between signals. In information theory, MI is a measure of how much extra information can a given signal Y provide about another signal X. In its most concise information theoretic form, MI can thus be expressed as:

$$I(X,Y) = H(X) - H(X|Y)$$

Where $H(X)$ refers to the entropy of X, and $H(X|Y)$ is the conditional entropy of X given Y. Expressing these quantities in terms of probability distributions of the two random variables, it can be shown that MI captures their mutual dependence. That is, it quantifies how close the joint distribution of the two signals is to the product of their marginal distributions.

$$I(X,Y) = E\left[log\left(\frac{P(x,y)}{P(x)P(y)}\right)\right]$$

$$I(X,Y) = \sum_{x,y} log\left(\frac{P(x,y)}{P(x)P(y)}\right)P(x,y)$$

$$I(X,Y) = \sum_{x,y} \Big(log\big(P(x,y)\big) - log\big(P(x)P(y)\big)\Big)P(x,y) \tag{4.2}$$

Looking at equations 4.1 and 4.2 together offers some insight into how correlation and mutual information convey the relationship between the signals X and Y. While the former creates a weighted sum of product of signal values, the latter generates a weighted sum of joint probabilities.

Some key mathematical properties of MI are:

- It is symmetric in X and Y, which does not capture a sense of direction in causality. However, we again use a shifted MI as a proxy for causal direction.

- Larger is the value of mutual information, higher is the mutual dependence between the two signals.
- It does not concern itself with the linearity of the signals involved, since it only worries about the probability distribution of the two signals. But evaluating these distributions accurately is significantly more challenging than calculating a simple shifted correlation.

4.3 Granger Causality

Given by the British economist Clive Granger, Granger Causality is a statistical test to determine whether a time signal can help forecast another time signal [7]. Thus, this test can be used to confirm only the idea of "predictive causality". A signal Y is said to Granger cause another signal X if the prediction of future values of X based on its own as well as Y's past values is "more accurate" than that based on X's past values alone.

Essentially, GC is an F-test, where the test statistic (an attribute of a sample) follows the F-distribution. This distribution is parametrised by two quantities d_1 and d_2, such that its random variate Z can be expressed in the form $Z = \frac{W_1/d_1}{W_2/d_2}$, where W_1 and W_2 are random variables following the chi-squared distribution parametrised by d_1 and d_2 degrees of freedom respectively. The random variate of a chi-squared distribution W can itself be expressed in the form $W = \sum_{i=1}^{d} X_i^2$, where X_is are independent normal random variables.

From here, it is easy to see how an F-test can be applied to estimate GC. We define predictability in the sense of how closely previous values can be used to fit the future value of a time signal. Thus, in the "restricted model" M_1, we have an

autoregression of X given by

$$x_t = a_0 + \sum_{i=1}^{m} a_i x_{t-i} + \epsilon_t$$

In the "unrestricted model" M_2 we append this autoregression with past values of Y as

$$x_t = a_0 + \sum_{i=1}^{m} a_i x_{t-i} + \sum_{i=1}^{q} b_i y_{t-i} + \epsilon_t$$

We can now define the F-statistic for determining goodness of fit as the following, using the Z representation given above, where RSS refers to the residual sum of squared errors, p_i refers to number of parameters of model i, and n is total number of data points:

$$F(X,Y) = \frac{\left(\frac{RSS_1 - RSS_2}{p_2 - p_1}\right)}{\left(\frac{RSS_2}{n - p_2 + 1}\right)}$$

$$F(X,Y) = \frac{\left(\frac{RSS_1 - RSS_2}{q}\right)}{\left(\frac{RSS_2}{n - (m+q+1)}\right)} \tag{4.3}$$

The null hypothesis for this test says that Y does not Granger cause X. Thus, larger is the value of this F-statistic, the more likely we are to reject this hypothesis. Thus, we reject it when $F(X,Y)$ is more than the critical value of the F-distribution parametrised by q and $n - (m + q + 1)$, for some desired false-rejection probability α like 0.05.

Some key mathematical properties of GC are:

- It is asymmetric in X and Y, which inherently captures a sense of direction in causality.
- Larger is the value of the F-statistic, higher is the predictive causality from Y to X.

- It assumes the causal relationship between two signals to be linear, since the autoregression is linear and not polynomic.

4.4 Transfer Entropy

In probability and information theory, Transfer Entropy refers to the amount of information transfer between two random processes. In words, it measures something similar to GC, that is, it quantifies the reduction in uncertainty of future values of X, by knowing past values of Y, given past values of X. In another sense, it is nothing but the conditional mutual information of X and Y, given past values of Y. If d is the lag assumed, then it is given by:

$$T(X,Y) = T_{Y\to X} = H(X_t|X_{t-1:t-d}) - H(X_t|X_{t-1:t-d}, Y_{t-1:t-d})$$

Assuming a lag of 1 unit in the simplest case, we can write $T(X,Y)$ as:

$$T(X,Y) = T_{Y\to X} = \sum_{x_{t+1},x_t,y_t} \Big[log\big(P(x_{t+1},x_t,y_t)P(x_t)\big) - log\big(P(x_t,y_t)P(x_{t+1},x_t)\big)\Big] P(x_{t+1},x_t,y_t) \tag{4.4}$$

- It is asymmetric in X and Y, which inherently captures a sense of direction in causality.

- Larger is the value of transfer entropy, higher is the information transfer between the two signals (or "processes").

- It does not concern itself with the linearity of the signals involved, since it only worries about the probability distribution of the two signals. But evaluating

these distributions accurately is significantly more challenging, and usually longer time series data is required.

- For auto-regressive processes, Transfer Entropy has been shown to reduce to Granger Causality.

Aside on Laplace Smoothing

One key issue with using gene expression time-series data, is that most real datasets are recorded over a very small span of time, say around a thousand time-steps. Depending on the domain of the time signal X, assuming it's discrete given some least count of measurement, one could imagine that evaluating probabilities over a larger domain/sample space would require a larger amount of data. Also, the more the number of variables in a joint distribution, the larger the sample space of the distribution becomes. Looking at the expressions for the four methods above, it becomes clear that the probability estimates from the data alone, may not be close to the true probabilities.

In such cases, we can do what is called "smoothing" of the data. Say we are given counts over a sample space $S = (s_1, s_2, ..., s_n)$ of size n as $(c_1, c_2, ..., c_n)$, total count C, one could find the probability distribution parameters as:

$$\theta_i = \frac{c_i + \alpha}{C + \alpha n}$$

If $\alpha = 0$, there is no Laplace Smoothing. In practice, α is kept at a small value less than or equal to 1. We keep it as unity for our analysis, and use this smoothing for mutual information and transfer entropy.

4.5 Convergent Cross Map

All four methods listed above, treat time-series signals as being random variables with some underlying probability distributions. However, there is another picture of looking at a time-series signal: as it being a part of some dynamical system. A dynamical system represents the temporal evolution of a signal in geometric space, often called the manifold representation of the system. The assumption is no longer of randomness, rather, of a deterministic system, which has the potential for showing unpredictable behaviour, often termed as "chaotic" behaviour.

Consider two time-series signals X and Y belonging to the same dynamical system represented by the manifold M of dimension d. Since we are dealing with non-linear biological systems, we assume A non-linear dynamical system. Mathematically, we can express the manifold as:

$$M(t) = [X(t), Y(t), ...]$$

We can now define something called a "shadow manifold" with respect to one of the signals, say $M_X(t)$ constructed by time-lagged (τ-lagged) values of X. We assume $M_X(t)$ to have a dimensionality E no less than that of M, that is, $d \leq E$, and given by:

$$M_X(t) = [X(t), X(t-\tau), ..., X(t-\tau(E-1))]$$

Note that for our implementation, we choose a time-lag interval of $\tau = 1$. Thus:

$$M_X(t) = [X(t), X(t-1), ..., X(t-E-1)]$$

Now, using Taken's embedding theorem, one can reconstruct M from M_X, since there exists a one-to-one correspondence between the true manifold M and the shadow manifold M_X (that is, they are diffeomorphic). Similarly, one can say that M_Y is diffeomorphic to M, and thus, the two shadow manifolds are diffeomorphic to one another. Because X and Y are dynamically coupled, points that are nearby on M_X will correspond temporally to points that are nearby on M_Y (see Figure 4.1). This enables us to estimate states across manifolds using Y to estimate the state of X and vice-versa using k-nearest neighbours [8]. With longer time series, the shadow manifolds become denser and the neighbourhoods shrink, allowing more precise cross-map estimates.

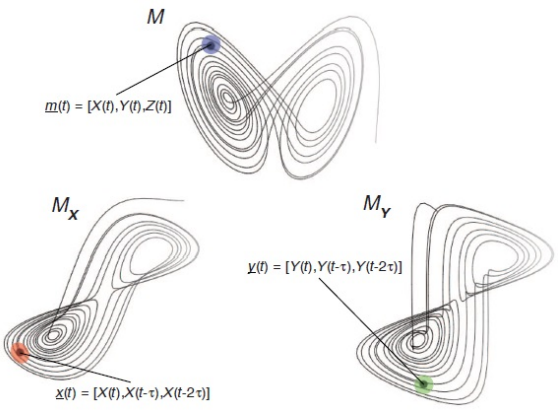


Figure 4.1: A sketch showing the attractor manifolds: true manifold M and shadow manifolds M_X and M_Y, with a diffeomorphism across all three [8].

Thus, without loss of generality, if $\hat{X}_Y$ be the reconstruction of signal X from shadow manifold M_Y, then a qualitative degree of causality using CCM can be given by the PCC between the two. That is, to test for $X \to Y$, we find

$$C(X, Y) = \rho(X, \hat{X}_Y) \tag{4.5}$$

Note that although the direction of causality might appear counterintuitive,

CCM says that if $X \rightarrow Y$, then a signature of X must be contained in Y. Some mathematical properties of CCM are:

- It is asymmetric in X and Y, which inherently captures a sense of direction in causality.
- Larger is the value of correlation between original and reconstructed signals, higher is the degree of causality between the two signals.
- It does not assume linearity of the system.
- Longer is the time-series, larger is the "library size", denser is the manifold, smaller is the neighbourhood, and higher is the precision in establishing causality.
- Although not entirely a substitute for other metrics like GC, CCM has been shown to work well for weakly coupled systems.

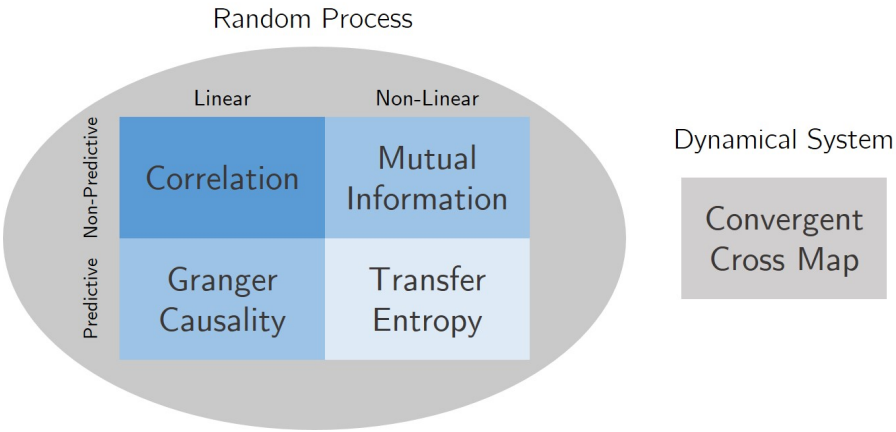


Figure 4.2: A succinct graphical summary of the methods used for estimating pairwise causality between signals X and Y.

Chapter 5

Intrinsic Graph Estimation (IGE)

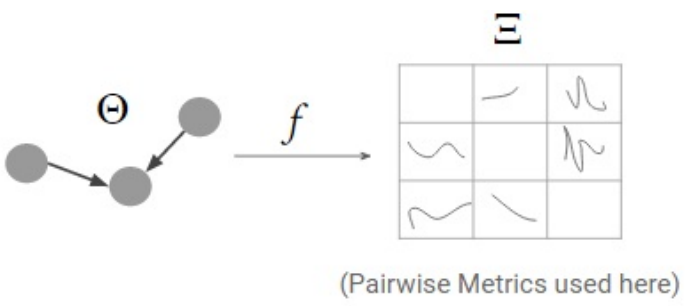
The other of the two core approaches in our work is to look at global causal computational models of a gene regulatory network. Clearly, using just pairwise metrics for determining the entire causal model is a naive strategy, wherein since all pairwise metrics are directly proportional to the strength of causality, essentially this simple algorithm operates: (1) sort ${}^{n}C_2$ edges by metric value in decreasing order, (2) choose top-k edges and output as graph G. However, one could imagine a more sophisticated algorithms to come up with this ordering. One which also looks at higher order interactions between the nodes of the graph, by taking the original pairwise metric matrix as an input. ARACNE is one simple yet powerful algorithm which looks at small higher order interactions (like the DIP inequality), however, we'd like to look at all possible interactions in the graph. Taking forward from the work of Noda et al. [10], we present the method of intrinsic graph estimation.

Let us call the given pairwise metric matrix Ξ (called the observation matrix), and the underlying graph structure (what we have to discover) represented by its adjacency matrix Θ. One can now imagine a mapping f which captures how ob-

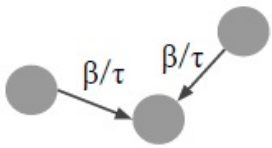
served data (gene expression modality such as one or more of the pairwise metrics) rises from the network structure (see Figure 5.1a). That is:

$$f : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$$

$$\Theta \mapsto f(\Theta) = \Xi$$



(a) f maps from graph structure to observation matrix



(b) Random walk model for modelling network interactions

Figure 5.1: Schematics of intrinsic graph estimation for single-attribute observation matrix

Now, we are in a position to explicate the network interactions. The assumption is that the observation matrix can be obtained as a weighted linear contribution of multiple-order interactions in the network structure. That is,

$$\xi_{ij} = \underbrace{c_i}_{\text{zero-order}} + \overbrace{c_{ij}\theta_{ij}}^{\text{first-order}} + \underbrace{\sum_{k\in V} c_{ij}^{k}\theta_{ik}\theta_{kj}}_{\text{second-order}} + \overbrace{\sum_{k,l\in V} c_{ij}^{kl}\theta_{ik}\theta_{kl}\theta_{lj} + \dots}^{\text{higher-order interactions}}$$

Also, every observation datum has some additive natural noise, which could be Gaussian.

$$t_{ij} = \xi_{ij} + \epsilon \text{ where } \epsilon \sim N(\mu, \sigma^2)$$

Ultimately, the objective of intrinsic graph estimation (IGE) would be do find the constants c. To do that, we need to have a notion of error which we would like to minimise, so that these constants best fit the observed data for given structure Θ. This can simply be the squared error, given by the following, where ρ are the parameters of f (or the constants of the interaction model described above):

$$J(\rho, \Theta) = \sum_{i,j \in V, i \neq j} \Big(t_{ij} - [f(\Theta)]_{ij} \Big)^2$$

However, notice that the structure Θ is not exactly given. Therefore, we must estimate the structure simultaneously with the parameters of the model. This is a typical EM-Algorithm style setting where we can have the following:

$$\text{E step: } \Theta = f^{-1}(\Xi)$$

$$\text{M step: } \rho = \underset{\rho}{\operatorname{argmin}}\, J(\rho, \Theta^m)$$

Before we go on to describe the algorithm, it's important to explicate which mathematical functions are a good candidate for f. To understand that, we will describe this problem by a **random walk model** on the graph Θ. If $\beta \in \mathbb{R}$ be the transition probability in a short interval $1/\tau$ (see Figure 5.1b), then (check for yourself that) the probability matrix for this interval can be written simply as $I_n - \frac{\beta}{\tau} L(\Theta)$, where the digraph Laplacian is given by

$$L(\Theta) = \begin{bmatrix} \sum \theta_{1k} & -\theta_{12} & \dots & -\theta_{1n} \\ -\theta_{21} & \sum \theta_{2k} & \dots & -\theta_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -\theta_{n1} & -\theta_{n2} & \dots & \sum \theta_{nk} \end{bmatrix}$$

Now, if we look at the τ-step probability matrix, then on considering the continuous time limit of the random walk, we get the transition probability matrix as:

$$\lim_{\tau\to\infty}\left(I_n - \frac{\beta}{\tau}L(\Theta)\right)^{\tau} = e^{-\beta L(\Theta)} \text{ ; where } e \text{ refers to the matrix-exponential map}$$

Now, by appending a positive multiplicative term $\alpha \in \mathbb{R}_+$, one can compare this to the interaction model described above, and thus a good candidate function f can be:

$$f(\Theta;\alpha,\beta) = \alpha e^{-\beta L(\Theta)} \text{ and } \rho = \{\alpha,\beta\}$$

And f^{-1} can be given by the matrix-logarithmic map, that is:

$$[log\Xi]_{ij} = [log(\alpha I) - \beta L(\Theta)]_{ij} = \begin{cases} log\alpha - \beta\sum_{k\in V}\theta_{ik}, & i = j \\ \beta\theta_{ij} & i \neq j \end{cases}$$

Thus, graph structure can be estimated as:

$$\theta_{ij} = \frac{[log\Xi]_{ij}}{\beta} \text{ ; for } i \neq j$$

Given the mathematical background above, we describe this in Algorithm 1. Note that while estimating the parameters Θ and ρ, we also need to update the diagonal elements of Ξ, which are otherwise undefined.

Algorithm 1 Intrinsic Graph Estimation for Single-attribute Data

Input: Metric matrix $T \in \mathbb{R}^{n \times n}$, maximum iterations k'

$\Xi = T + rI_n$ ▷ Where r is such that $|\Xi| \neq 0$ so that matrix-log exists

for $m = 1$ to $n(n-1)$ **do** ▷ Iterating over best networks for m edged networks

  for $k = 1$ to k' **do**

    $$[\hat{\Theta}]_{ij} = \begin{cases} 1, & |[log\Xi]_{ij}| \geq \zeta_m \text{ and } i \neq j \\ 0, & \text{otherwise} \end{cases}$$ ▷ Thresholding to select top-m edges

    $\rho^m = \text{argmin}_\rho \, J(\rho, \Theta^m)$ ▷ Find optimal parameters $\rho = (\alpha, \beta)$

    $\Xi = T + diag(\alpha^m e^{-\beta^m L(\Theta^m)})$ ▷ Update diagonal elements of metric matrix

$\hat{m} = \text{argmin}_{m \in \{1,2,\ldots n(n-1)\}} \, J(\rho^m, \Theta^m)$

return $\hat{\Theta}^m$

5.1 Incorporating a Multi-attribute Observation Matrix

At times, one might be dealing with a multi-attribute observation matrix. That is, the graph structure could be giving rise to more than one observable phenomenon. Say in our case, it could be "correlative" phenomenon, "Granger causal" phenomenon, "CCMic" phenomenon, etc. How do we account for all of these observations from the same structure matrix Θ? One could imagine taking a weighted linear combination of all these features to form a single-attribute matrix, like we had in the original case. However, this proposes a critical issue. Up until now, IGE had a great advantage: it was a **parameterless model**, which means it didn't require any parameter tuning whatsoever. However, feature weighting would re-

quire us to tune these weights prior to applying IGE, that would amount to learning them under a supervised learning paradigm, which brings us back to the issue of paucity of data. To keep our methods parameterless and unsupervised, we propose the following extension.

We concatenate individual observation matrices to form a multi-attribute observation matrix, and thus can rewrite what we wrote for the single-attribute case as below, defining map g as (see Figure 5.2):

$$g : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n\times s}$$

$$\Theta \mapsto g(\Theta) = \Xi$$

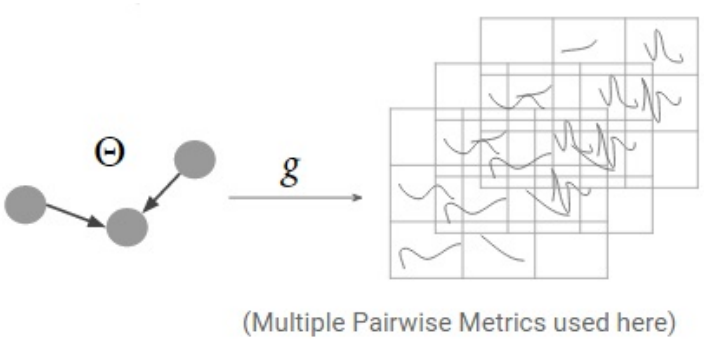


Figure 5.2: Schematic of intrinsic graph estimation for multi-attribute observation matrix

We can write the interaction model as:

$${}^{q}\xi_{ij} = {}^{q}c_i + {}^{q}c_{ij}\theta_{ij} + \sum_{k\in V} {}^{q}c_{ij}^{k}\theta_{ik}\theta_{kj} + \sum_{k,l\in V} {}^{q}c_{ij}^{kl}\theta_{ik}\theta_{kl}\theta_{lj} + \dots$$

$${}^{q}t_{ij} = {}^{q}\xi_{ij} + \epsilon$$

The error function can be written as:

$$\tilde{J}(\rho, \Theta) = \sum_{1 \leq q \leq s} \left(\sum_{i,j \in V, i \neq j} \left({}^{q}t_{ij} - {}^{q}[g(\Theta)]_{ij} \right)^2 \right)$$

Now however, one might ask which function is a good candidate for g. So as to not abandon the advantages of using an elegant random walk model, let us try to incorporate the function f we defined above, and define g as:

$$g(\Theta) = cat\big({}^{1}f(\Theta), {}^{2}f(\Theta), \ldots, {}^{s}f(\Theta)\big) = \Xi$$

We now define the inverse map h:

$$\Theta = h\big({}^{1}f^{-1}({}^{1}\Xi), {}^{2}f^{-1}({}^{2}\Xi), \ldots, {}^{s}f^{-1}({}^{s}\Xi)\big)$$

There are more than one choices for the function h, as long as it as an aggregate-of-sorts of the network structures predicted by a single-attribute matrix alone. Thus, we have chosen the logical-and operator, while ensuring that the number of edges is limited to m in the m^{th} iteration.

$h(x_1, x_2, \ldots, x_s; m) = \{x_1^{m'} \wedge x_2^{m'} \cdots \wedge x_r^{m'}$: for smallest $m' > m$ such that number of edges is $m\}$

The final algorithm is described below in Algorithm 2.

5.2 Penalty on high connectivity - Regularisation

When a plot of the variation of error for the best Θ^m versus m is made, it is realised that the error values for highly connected networks is quite low, compared to those of sparser networks, for most observation matrices. (See Figure 5.3 below.) Thus, our current error function mostly tends to favour denser networks, which could be

Algorithm 2 Intrinsic Graph Estimation for Multi-attribute Data

Input: Metric matrix $T \in \mathbb{R}^{n \times n \times s}$, maximum iterations k'

$\Xi = T + rI_n$ ▷ Where r is such that $\forall q \in \{1, 2, \ldots, s\}$, $|{}^q\Xi| \neq 0$ so that matrix-log exists

for $m = 1$ to $n(n-1)$ **do** ▷ Iterating over best networks for m edged networks

  for $k = 1$ to k' **do**

    $m' = m$

    while true **do**

      $\hat{\Theta} = ones(n, n)$ ▷ Initialised to all 1s

      for $q = 1$ to s **do**

        $$[{}^q\hat{\Theta}]_{ij} = \begin{cases} 1, & |[log\Xi]_{ij}| \geq \zeta_{m'} \text{ and } i \neq j \\ 0, & \text{otherwise} \end{cases}$$ ▷ Thresholding to select top-m' edges

        $\hat{\Theta} = \hat{\Theta} \wedge {}^q\hat{\Theta}$

      if $sum(\hat{\Theta}) = m'$ **then**

        break

      else

        $m' = m' + 1$

    $\rho^m = \text{argmin}_\rho \, \tilde{J}(\rho, \Theta^m)$ ▷ Find optimal parameters $\rho = (\alpha, \beta)$

    for $q = 1$ to s **do**

      ${}^q\Xi = {}^qT + diag({}^q\alpha^m e^{-{}^q\beta^m L(\Theta^m)})$ ▷ Update diagonal elements of metric matrix

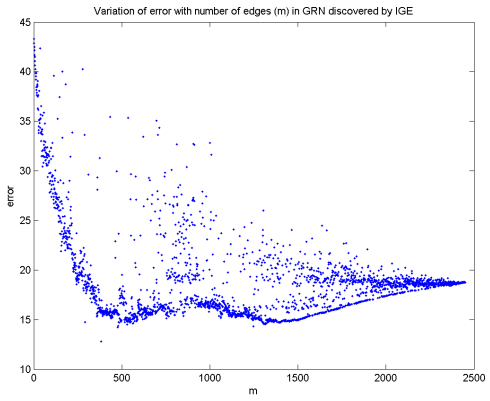
$\hat{m} = \text{argmin}_{m \in \{1,2,\ldots n(n-1)\}} \, \tilde{J}(\rho^m, \Theta^m)$

return $\hat{\Theta}^m$

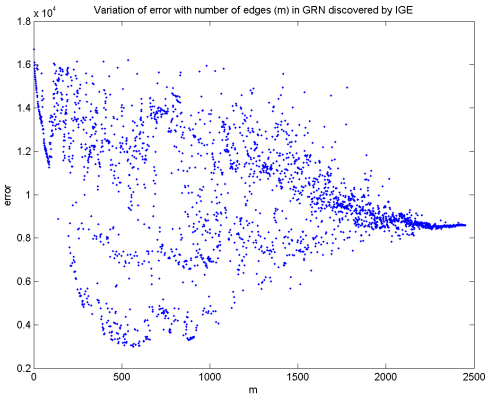
problematic for natural biological networks like GRNs, which are quite sparse. Thus, we can introduce a simple regularisation to the error function, as below:

$$\hat{J}(\rho, \Theta) = J(\rho, \Theta) + \lambda \sum_{i,j}^{n} \theta_{ij}$$

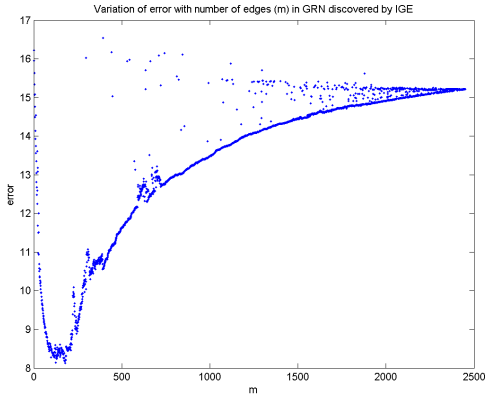
Notice, however, that this forces the introduction of a parameter λ to an otherwise parameterless model, which reduces the elegance of our model for better accuracy. Thus, perhaps an entirely different error function which naturally favours sparser networks can be imagined, which keeps IGE parameterless.



(a) Correlation



(b) Granger Causality



(c) Convergent Cross Map

Figure 5.3: Variation in error with number of edges for IGE in dataset: $N = 50$, $T = 1000$, 50 true edges

5.3 Hard Vs. Soft Thresholding

A key observation while running the IGE algorithm was that on a hard thresholding to select top-m edges, the inner for-loop converged very quickly, in a couple of iterations. This could imply that a hard edge-selection forces the optimisation algorithm to get stuck in a so-to-speak discretised variable space. Therefore, we decided to add another tweak to the algorithm, by employing soft edge selection. We use the smooth sigmoid function centred at ζ_m instead of the unit step function:

$$[\hat{\Theta}]_{ij} = \frac{1}{1 + exp(-|[log\Xi]_{ij}| + \zeta_m)}$$

And eventually, once the inner for-loop has converged, we use the old unit step function to obtain the graph estimate.

Chapter 6

Pagerank-Based Intrinsic Graph Estimation

Intrinsic graph estimation is essentially a problem of figuring out the underlying graph structure which gives way to some manifested graph link observations (that is, find the inverse map, f^{-1}). Now, the forward map f itself is unknown. The most critical property of f is that it captures higher order interactions between the nodes of the GRN, which is what the observations manifest themselves from. This idea can be captured by undertaking a random walk on the graph structure Θ.

6.1 Random Walk Models and Pagerank

Given a graph $G = \{V, E\}$, a random walk refers to the sequence of nodes visited, starting at a particular node $i \in V$ and then transitioning to a new (neighbouring) node $j \in V$, depending on the probability of transition P_{ij}. Say the random walk could be the sequence $X = \{4, 1, 17, \dots\}$, where X is a random variable. A popular random walk model involves treating the sequence of visited nodes as forming a

Markov chain, where the transition probabilities i $P(X_t = i/X_{t-1} = j)$ depend only on the current node. An entity of interest for random walks is the stationary distribution $P(X_\infty = i)$, which signifies the probability of being at a node i at $t = \infty$. In a crude sense, this distribution could be considered as a measure of centrality for node i in graph G.

A powerful model which takes the idea of "importance of nodes" a step further is the Pagerank Algorithm [11]. This algorithm also estimates some kind of a stationary distribution for a random walk on a given (un)weighted graph G, by figuring out the transition probabilities through a "voting" system. The descriptive algorithm is given below in Algorithm 3.

Algorithm 3 Pagerank for Weighted Graphs

1: **Input:** Weighted graph $G = \{V, E\}$ where $|V| = n$ and $W \in \mathbb{R}^{n \times n}$ is the weight matrix, Convergence threshold ϵ, Damping factor $d = 0.85$

2: $R_0 = \{0\}^n$, $R_1 = \{1\}^n$ ▷ Initialise ranks

3: $t = 1$

4: **while** $|R_t - R_{t-1}| > \epsilon$ **do**

5: $\quad t = t + 1$

6: $\quad$ **for** $i = 1$ to n **do**

7: $$R_t(V_i) = (1 - d) + d * \sum_{V_j \in In(V_i)} \frac{W_{ji}}{\sum_{V_k \in Out(V_j)} W_{jk}} R_{t-1}(V_j)$$

8: **return** R_t

Essentially, a sorting of the Pageranks sorts the nodes in decreasing order of "significance". Note, however, that we do not wish to find the importance of nodes. Rather we wish to find important real links in the graph (reverse map f^{-1}) and

important manifested relationships between nodes (forward map f) . Therefore, we need to convert the original graph G into a certain dual version called G', before running a Pagerank on it to find these significant links/link-relationships.

6.2 Link-Pagerank - Dual Graph Construction

Consider the graph $G = \{V, E\}$ where $|V| = n$ and $|E| = n(n-1)$, whose dual graph is $G' = \{V', E'\}$. Note that we assume G to be a fully-connected weighted directed graph (with no self-edges), having a weight matrix $W \in \mathbb{R}^{n \times n}$. We construct G' and run Pagerank through the following steps:

1. For every directed edge $e_{ij} \in E$ between nodes $i, j \in V$, add a node e'_{ij} to V'. Let us call these nodes in G' as linknodes.

2. For ever pair of linknodes (e'_{ij}, e'_{jk}), add a directed edge $\hat{e}_{ijk}$ to E'. Let us call these edges in G' as metalinks. (See Figure 6.1a.)

3. For every metalink $\hat{e}_{ijk} \in E'$, we assign a weight $W'_{ijk} = W_{ij} * s_j * W_{jk}$, where s_j is the Pagerank score of node $j \in V$. (See Figure 6.1b.)

4. Run Pagerank on G' and output Pagerank-matrix $R \in \mathbb{R}^{n \times n}$ such that Rij is rank of the linknode e'_{ij}.

The intuition for Step 2 is that we would like to connect those linknodes together which share a node.

For Step 3, the setting of weights is a very critical part of the algorithm. The weight of an edge signifies the amount of co-dependence between the two nodes it connects. We thus want a metalink to have a high (low) weight if the linknodes

it joins have a high (low) weight in the original weighted graph G, hence the term $W_{ij} * W_{jk}$. Also, the co-dependence becomes more (less) important if the common node j is itself highly (lowly) significant. Therefore, we also append a score term s_j to the product, which is nothing but the node j's score when Pagerank is run on the original graph G.

Now, when Pagerank is run on G' in Step 4, it gives a ranking of linknodes, which in the primal form would imply a ranking of edges in G.

6.3 Iterative Pagerank for IGE

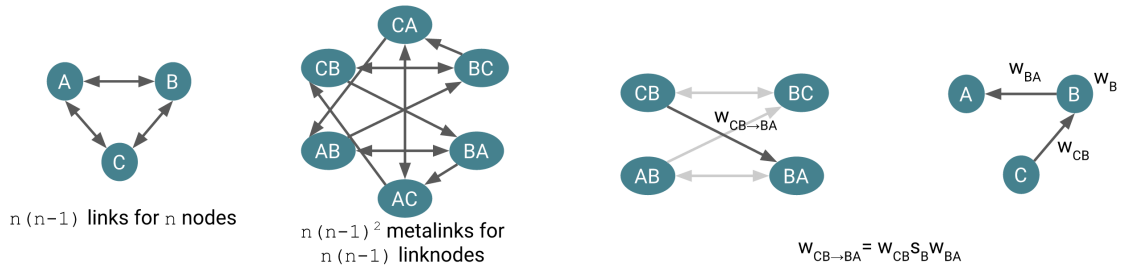
As described in the above sections, the link-Pagerank algorithm gives us links (or relationships) between nodes, after incorporation of global weighted-connectivity information. Therefore, it seems like a good candidate for the forward mapping f.Figuring out the inverse mapping f^{-1} is intuitively not as straight-forward, but the same algorithm can be used for this purpose. (See Figure 6.1c.)

Say we have the adjacency matrix Θ and the observation matrix Ξ. Now, one can imagine an EM-style iterative coordinate optimisation, in Θ-space and Ξ-space, till convergence, starting from the initial "given" Ξ estimate. However, we need an additional step in the inverse map direction ($\Xi \mapsto f^{-1}(\Xi) = \Theta$), that of soft-thresholding the matrix recovered from link-Pagerank. This step enforces this matrix to behave like an adjacency matrix, so that at convergence we have a good estimate of the mappings f and f^{-1}. The entire algorithm is described in Algorithm 4.

Although we don't present a theoretical validation for the algorithm here, we provide an empirical analysis on our datasets in Section 7.

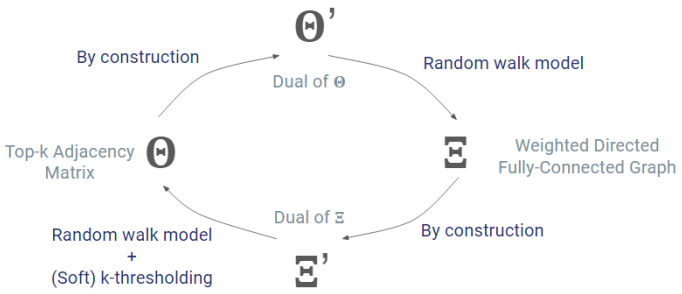
6.4 Pagerank Weighting for IGE

The idea of using a random walk model like Pagerank, can also be imagined in a simpler construction for graph estimation which incorporates some notion of a global significance of links. Consider the original observation matrix Θ_0. This accounts for only pairwise observations. Now, consider the link rank-matrix Θ_1, obtained after one iteration of link-Pagerank. This could be considered as a one-step improvement of Θ, in terms of global significance. Therefore, the new rank-weighted observation matrix $\Theta = \Theta_0 \circ \Theta_1$ (where $\circ$ is the pairwise product operator), when sorted in decreasing order, could indeed be the intrinsic graph structure. Again, we do not present a theoretical validation for the algorithm here, but we provide an empirical analysis in Section 7.



(a) Dual Graph Construction

(b) Setting Weights in G'



(c) Outline of Iterative-Pagerank IGE

Figure 6.1: Graphical Sketch for Iterative-Pagerank Intrinsic Graph Estimation

Algorithm 4 Iterative Pagerank for IGE for Single-attribute Data

Input: Metric matrix $\Xi \in \mathbb{R}^{n \times n}$, Convergence threshold ϵ

for $m = 1$ to $n(n-1)$ **do** ▷ Iterating over best networks for m edged networks

$\Xi_0 = \{0\}^{n \times n}$, $\Xi_1 = \Xi$

$t = 1$

while $|\Xi_t - \Xi_{t-1}| > \epsilon$ **do** ▷ Frobenius norm may be used

$R_\Theta = LinkPagerank(\Xi_t)$

$[\Theta^m]_{ij} = \dfrac{1}{1 + exp(-|[logR_\Theta]_{ij}| + \zeta_m)}$ ▷ Soft-thresholding to induce an m-node adj-mat

$t = t + 1$

$\Xi_t = LinkPagerank(\Theta^m)$

return Θ

Chapter 7

Experiments and Results

7.1 Time Lag and Amount of Past Information for a Causal Relation

Two important parameters in establishing causality between two signals concerns with the time scale over which causality operates, that is, the **"time lag"** and the **"amount of past information"** to be considered. The former refers to how far back in time the arrow of causality operates. That is, say $X \to Y$, then a time lag of τ means that $Y(t)$ is directly influenced by $X(t-\tau)$. The latter refers roughly to the amount of past information from the causing variable that is used to determine the caused variable. That is, say $X \to Y$, then past information amount η means that $Y(t)$ is directly influenced by $X(t-\tau), X(t-\tau-1), \cdots, X(t-\tau-\eta-1)$

Note that for correlation, mutual information and transfer entropy, the amount of past information used is fixed at 1 $\{X(t-\tau) \to Y(t)\}$, 1 $\{X(t-\tau) \to Y(t)\}$, and 2 $\{X(t-\tau), X(t-\tau-1) \to Y(t-\tau)\}$ respectively. While Granger causality and convergent cross map have a fixed time lag of 1 and 0 respectively. Thus

eventually, there is only a single hyperparameter for all our methods, and we call it the "max lag". This value was set to 10, such that the optimum lag chosen does not exceed it. For correlation, mutual information and transfer entropy, the optimum lag was assumed to be "optimistic", in the sense that the lag giving the largest metric value for that pair of signals, was chosen for them. For convergent cross map, we use a small standard value of amount of past information of (2 or 4). For Granger Causality, an information theoretic criterion, called the Bayesian Information Criterion (BIC) was used [9] to find the amount of past information to be used. BIC trades off the model complexity with the likelihood of the model, and is given by:

$$BIC = -2 \cdot ln(L) + k \cdot ln(n)$$

Where k is number of parameters, and n is number of data points. For GC, this can be written in terms of RSS as:

$$BIC = n \cdot ln(RSS/n) + k \cdot ln(n)$$

And then, the past information amount which gives the smallest BIC is chosen as the optimal one.

7.2 Performance Metrics

The observations are interpreted using four metrics of performance:

- **Precision** is defined as the fraction of retrieved data that is accurate. It is measured as the ratio of number of correct edges detected (*true positive*) to

the total number of edges in the inferred network (*test positive*).

$$precision = \frac{true\ positive}{test\ positive}$$

Higher the value of precision, more is the "goodness" of the inferred network.

- **Recall** is the fraction of accurate data that was retrieved by the method. It is measured as the ratio of edges detected correctly (*true positive*) to the number of edges in the real network (*condition positive*).

$$recall = \frac{true\ positive}{condition\ positive}$$

It denotes the coverage of the network by the method.

- **F-score** can be interpreted as a weighted average of the precision and recall, where an F-score reaches its best value at 1 and worst at 0. High precision could also be obtained due to a small number of edges getting successfully detected with most of the true edges getting rejected (*false negatives* or Type II error). Similarly, high recall may be obtained by selecting all the possible edges, most of which will be spurious (*false positives* or Type I error). Therefore, F-score is considered a more composite measure since it takes both aspects into consideration. The formula used here is the traditional *balanced F-score* and is the harmonic mean of precision and recall.

$$F\text{-}score = \frac{2 * precision * recall}{precision + recall}$$

- **Receiver Operating Characteristic (ROC)** is a standard measure for the performance of a binary classifier, in our case, whether the edge under consideration is a part of the real network or not. The curve is created by plotting the *true positive rate* (TPR, also known as precision) against the *false*

positive rate (FPR, also known as fallout). False positive rate is the fraction of edges reported to be true but were actually false. When using normalised units, the area under ROC curve (often referred to as simply the AUC, or AUROC) is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one, where the higher ranked sample is considered to be positive. Therefore, a "random" classifier would have an AUROC of 0.5, and a perfect classifier would have an AUROC of 1. This is the primary measure used in the results below.

7.3 Pairwise Random Variable based Techniques

On steady state data

For pairwise metrics on steady-state data, experiments were run over all combinations of:

- Number of genes: $N \in \{10, 20, 50, 100\}$
- Length of time-series: $T \in \{500, 1000\}$
- Quantisation levels: $Q \in \{2, 5, 10, 20\}$
- Pairwise metrics: $metrics \in \{CO,\ GC,\ MI,\ TE,\ CCM\}$

However, after hundreds of experimental runs, it was realised that larger networks gave the poorest results. Additionally, the change of quantisation level changes the performance of the information theory methods. One might expect there to be a sweet-spot value of Q where neither the sample space is too large, nor too much quantisation error is induced in the calculations. Figuring an optimal Q is thus very

crucial. Eventually, we stick to including results only for $N \in \{50, 100\}$, and $Q = 20$ (quantisation done, naturally, only for probabilistic methods of transfer entropy and mutual information).

Dataset	CO	GC	MI	TE	CCM
N50S500	0.85008	0.48317	0.50385	0.60557	0.75741
N50S1000	0.92128	0.51899	0.43182	0.6201	0.83638
N100S500	0.82886	0.46239	0.60465	0.52031	0.76961
N100S1000	0.86609	0.55725	0.47071	0.65968	0.63700

Table 7.1: AUROCs for Steady State data using Pairwise Metrics

Clearly, simple shifted correlation performs the best amongst all the metrics, followed by CCM. (Note that the CCM results here are with $E = 4$.) This is probably because of the fact that the data is just steady-state data, with no sense of a temporal ordering between the "dimensions" of the data (i.e. $X(t)$ doesn't immediately follow $X(t-1)$; in fact the nomenclature is rather misleading for steady state data and we should just represent X as a vector $\{x_0, x_1 \cdots\}$). Hence, methods which assume this temporal relationship tend to make mistakes. Another thing to note is that an increase in the data length (steady state conditions) had an expected improvement in the performance of pairwise metrics.

On time series data

Since for each of the two DREAM4 GRNs we have 5 expression simulations, we aggregate our metrics as an average for the purpose of further analysis. Over here too, interestingly, correlation and CCM perform the best, followed by GC. (Note

that the CCM results here are with $E = 2$.) For the smaller network, CCM performs the best, and one can say that it is able to exploit the temporal ordering of data dimensions to make better predictions about causality. For the larger one, correlation is best, followed very closely by GC, for which a similar conclusion could be drawn.

Dataset \ Metric	CO	GC	MI	TE	CCM
1	0.68267	0.59467	0.43511	0.69867	0.66756
2	0.64105	0.60980	0.49578	0.54561	0.76182
3	0.69422	0.63378	0.49778	0.59111	0.58133
4	0.71728	0.62338	0.41858	0.63836	0.76723
5	0.73504	0.86111	0.47115	0.58974	0.86538
Average	0.69405	0.66455	0.46368	0.61270	0.72866

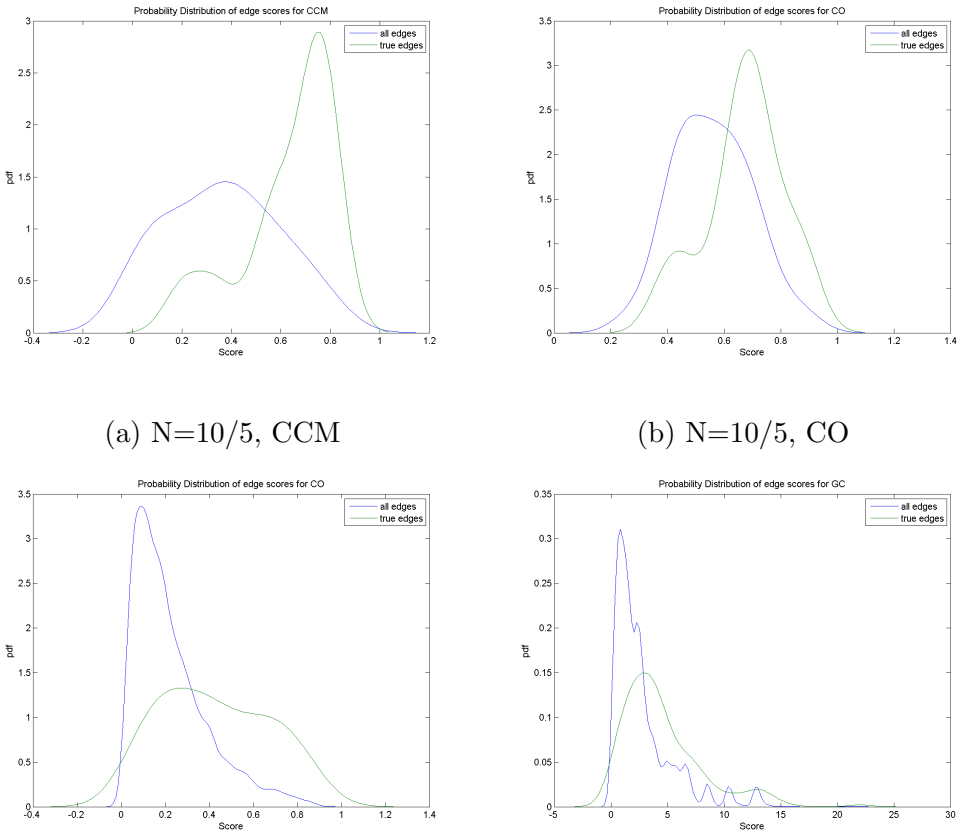
Table 7.2: AUROCs for DREAM4 Time Series ($N = 10$) data using Pairwise Metrics

Dataset \ Metric	CO	GC	MI	TE	CCM
1	0.74943	0.73735	0.55882	0.49309	0.68174
2	0.67353	0.63091	0.60383	0.55015	0.61816
3	0.70322	0.60340	0.48884	0.53302	0.65147
4	0.68245	0.64466	0.51648	0.52689	0.60043
5	0.71630	0.62661	0.51397	0.55264	0.59193
Average	0.70499	0.64859	0.53639	0.53116	0.62874

Table 7.3: AUROCs for DREAM4 Time Series ($N = 100$) data using Pairwise Metrics

One can also see the discriminatory power of these metrics in their score distribu-

tions, as plotted in Figure 7.1. The more significantly different and right-inclined is the distribution of true edges vs. all possible edges, the better is the classifier. The surprising conclusion still remains, that correlation performs the best. We explore reasons for this in Section 7.6.



(a) N=10/5, CCM

(b) N=10/5, CO

(c) N=100/1, CO

(d) N=100/1, GC

Figure 7.1: Score Distributions for Pairwise Metrics on DREAM4

7.4 IGE on DREAM4 time series datasets

Since IGE is computationally very expensive, we choose the smaller 10-node DREAM4 dataset to test it. When we merely replicate the original IGE formulation of Noda et al., using a single-attribute observation matrix without any regularisation, for all the pairwise techniques the AUROC surprisingly lowers when we use IGE, instead

of the naive top-k selection strategy. The networks are denser, which can be tackled using regularisation. Soft-thresholding does improve the results from regular IGE, but they still remain lower than the pairwise metrics. Clearly, the problem in the current algorithm is more than just this:

- The current function f, is the matrix exponential map. Its inverse function, which is critical to estimating Θ, is the matrix logarithmic map. Although the former exists and is always well-defined, the *log* of a real matrix exists if and only if it's invertible (i.e., determinant $\neq 0$). Even if the *log* exists, it is not unique, unless its eigenvalues lie in the strip $\{z \in \mathbb{C} | -\pi < Im(z) < \pi\}$. The unique *log* is called the principal logarithm. However, since it may not be unique, the inverse map could map to an entirely different Θ. Thus, one possible direction of further work is to choose another function which satisfies the random walk model.

- We have employed MATLAB's $fminsearch$ function to solve the unconstrained optimisation problem in minimising the error function. Despite increasing the option of maximum function evaluations and maximum number of iterations, a few times the function returns without reaching the optima. This is possibly because the error landscape is more complicated that a simple convex function, and/or the starting point of initialisation of parameters ρ (which is being drawn from a random number between 0 and 1) is too far off from the minima.

We therefore went on to suggest Page-ranked versions of IGE which could potentially solve these issues. On using them as a preliminary stance, we obtain performances very close to the pairwise metrics, AUROCs being sometimes more or

less than the latter. Hence, a lot still remains to be done, and the theory of random walk models needs to explored deeper so as to tap into these ideas in a more sound manner.

7.5 ARACNE on large synthetic steady state datasets

In order to compare our results with those of popular network finding methods, we tested the performance of ARACNE on our datasets. It works by estimating MI values, thresholding on the basis of p-value, and then using DPI to further remove false edges. To create ROCs, we generate the complete MI matrix for all networks by setting p-value as 1 and DPI value as 0, indicating that no pruning is done. Then, we varied DPI from 0 to 1, with a step size of 0.01, maintaining the same p-value of 1, so that the parameter being tuned is the DPI tolerance.

Dataset	ARACNE	CO	CCM
N50S500	0.78728	0.85008	0.75741
N50S1000	0.81242	0.92128	0.83638
N100S500	0.73914	0.82886	0.76961
N100S1000	0.74119	0.86609	0.63700

Table 7.4: AUROCs for Steady State data using ARACNE

It isn't surprising that two of our pairwise metrics are doing better than ARACNE, since the latter is only a minor global-level improvement to the MI metric using DPI.

7.6 State-of-the-art methods on DREAM4 time series datasets

The DREAM4 dataset has been popularly used for benchmarking many network inference methods. In one such study done by Young et al. they enclose the performance metrics of many popular methods (including ARACNE), including the one they have proposed themselves (ScanBMA) [5]. It's intriguing to note that 3 of our pairwise metrics are competitive with these methods on this dataset, with shifted correlation being the best one for the larger 100-sized GRN.

Method	N=10	N=100
LASSO	0.731	0.643
ARACNE	0.668	0.589
CLR	0.681	0.699
MRNET	0.709	0.701
ScanBMA	**0.740**	0.657
CO	0.694	**0.705**
GC	0.665	0.649
CCM	0.729	0.629

Table 7.5: Comparison of average AUROCs for State-of-the-art vs. Pairwise Metrics on DREAM4

To explore possible hypotheses as to why correlation is, surprisingly, almost always the best method, we make note of the following points.

Firstly, shifted correlation is a very simple technique, with only a single point of

past information taken at a τ-lag being used to determine the current signal value: $X(t-\tau) \to Y(t)$. Whereas both GC and CCM take a usually larger amount of past information: $\{X(t-1), \cdots X(t-E)\} \to Y(t)$ and $\{X(t), \cdots X(t-E-1)\} \to Y(t)$ respectively. Thus, possibly GC and CCM tend to overfit the data assuming a larger past contribution. To test this, we do a "max lag analysis", wherein we vary the max lag parameter and plot variation in AUROC with it (see Figure 7.3). Although the variation in GC is not as significant, we see that the AUROC is high for low max lag value for GC and CCM. In fact, it is the best for CCM at the smallest possible amount of past information of 2 (when $E = 1$) and (almost) best for GC at the same value of 2 (when $E = 2$). Moreover, another reason to believe in this overfit-hypothesis is that when we observe the precision-recall curves at very low recall, we see correlation very significantly outperforming GC and CCM in terms of the precision. A trend which is sustained as the precision falls with rise in recall (see Figure 7.2).

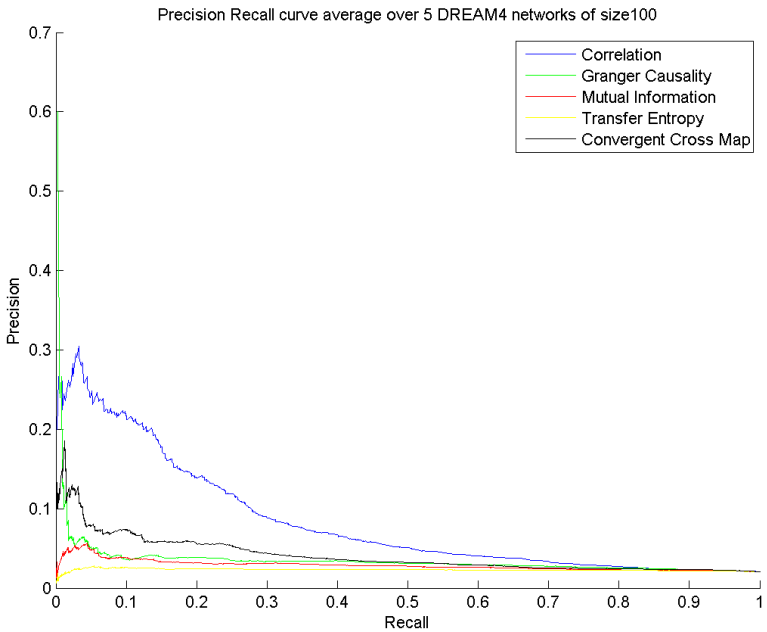


Figure 7.2: Precision-Recall Curve for Pairwise Metrics on DREAM4 size 100

Secondly, while we have fixed the hyperparameter E for CCM for all pairs of signals, we allow a free reign over the time-lag parameter for correlation (since it can be shifted by max lag amount for every pair of signals separately). This could possibly offer some "undue advantage" to correlation, thus letting it perform better. To confirm this, we enforce the same max lag value to be the actual lag across all pairs of signals, and plot the variation in AUROC with that in max lag for correlation in Figure 7.3. We indeed see a consistent decline in the performance of correlation. Although even then, correlation at the minimum max lag beats GC and CCM, bringing more credibility to the first hypothesis of a potential overfit.

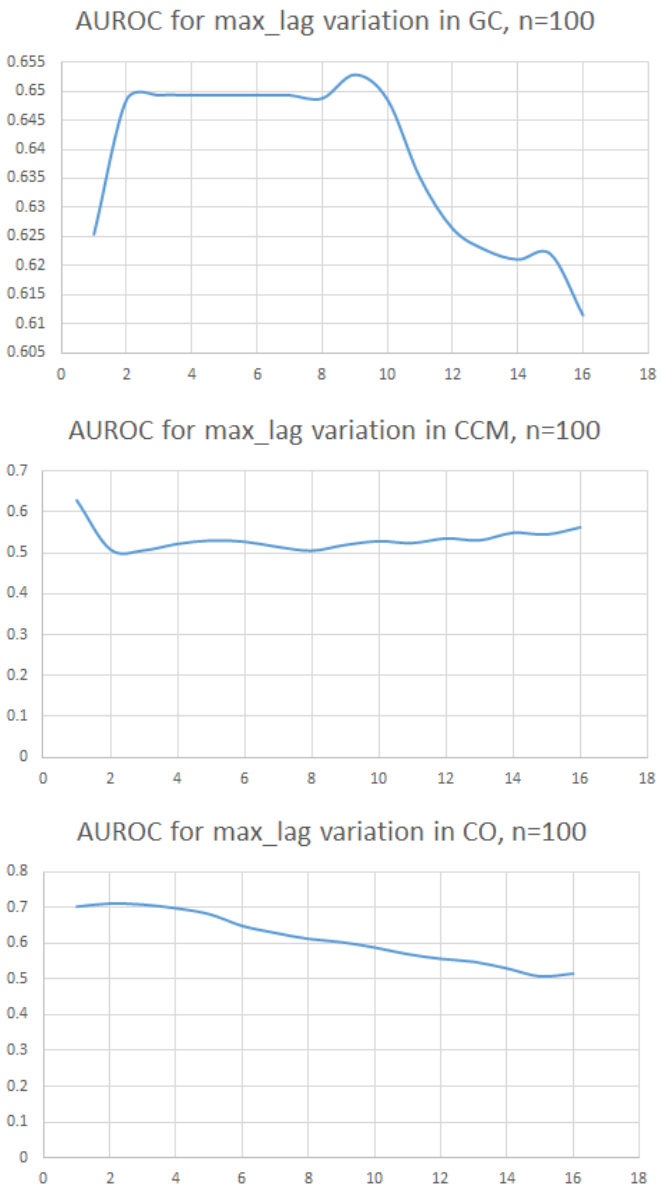


Figure 7.3: Max Lag Variation for Pairwise Metrics on DREAM4

Chapter 8

Summary and Future Work

This project was an attempt to review current literature, spanning domains of gene regulatory networks, biological network inference and causality detection and estimation. We started out with investigating methods which would look at pairwise levels of interaction to detect and quantify the existence of a causal relation. After a thorough review, we employed five techniques for this.

Correlation, Granger Causality, Mutual Information and Transfer Entropy are four methods which treat time-series signals as a random variable, following some underlying probability distribution. While the first two work for linear systems, the latter two for non-linear, GC and TE can principally distinguish between the two directions of causality. However, TE and MI suffer from the poverty of data, since they make use of joint probabilities over a large sample space. Surprisingly, the ultimate winner amongst these techniques was simple shifted correlation. Additionally, one method which seems to outrun all of these (but correlation) was of convergent cross map. This metric assumes signals as belonging to a dynamical system. It is indifferent to the (non-)linearity of the system, and respects the direction

of causality as well. Our experiments show significant measures in performance, making it the most suitable candidate to detect and estimate pairwise causality, alongside correlation. There is still scope for running **further diagnostics**, as to what kind of relationships are better captured by what kind of pairwise metrics. Say, one metric could be picking up links of high-degree nodes, while another could be picking up links of low-degree nodes. After a preliminary analysis, we observed that when we look only at the correctly detected edges, links with high-degree out-nodes are better detected by CO, followed by CCM, and barely any by GC. For links with high-degree in-nodes, we see an opposite trend. More such analyses could help generate **ensemble models** for higher performance across different datasets.

In all likelihood, the problem of GRN discovery is underconstrained, if we stick to looking at only pairwise interactions. Thus, we moved to a global optimisation scheme of intrinsic graph estimation. However, in its simplest single-attribute form, it gives results slightly poorer than pairwise techniques. There is a need to improve this unsupervised algorithm, particularly in terms of deciding a functional map f with a unique inverse. We hypothesise the use of Page-rank for this, outlining some key ideas and algorithms, and this is an interesting branch for further exploration in network inference for GRNs.

Furthermore, GRNs consist of genes influencing behaviour of other genes, both upstream and downstream. One could imagine characterising a local vicinity of influence over which these methods of causality are employed. Moreover, various global properties of biological networks, such as their scale-free nature, average path lengths, observance of Data Processing Inequality as in ARACNE, etc. can also help in constraining the causal computational model. Although we have moved towards a

method of global optimisation, a **trade off between local and global structures** is expected to improve the results many fold.

And finally, one of the primary conclusions of our project is that simple shifted correlation seems to outweigh most of the sophisticated state-of-the-art methods of GRN discovery. While this shows the overfitting nature of added model sophistication, it also exposes the simplicity of synthetic datasets that are being used for benchmarking gene network inference. Therefore, we strongly believe that the ultimate test for these methods lies in applying them to real biological datasets. **Because our algorithms and analyses can only be as good as (and rather peculiar to) our data.**

Bibliography

[1] Frank Emmert-Streib, Matthias Dehmer, and Benjamin Haibe-Kains, *Gene regulatory networks and their applications: understanding biological and medical problems in terms of networks*, Front Cell Dev Biol., 2014, 2: 38

[2] Gary Hak Fui Tam, Chunqi Chang, and Yeung Sam Hung, *Gene regulatory network discovery using pairwise Granger causality*, IET Syst. Biol., 2013, Vol. 7 Iss. 5, pages 195–204

[3] Katerina Hlaváčková-Schindler, Milan Palušb, Martin Vejmelkab and Joydeep Bhattacharyaa, *Causality detection based on information-theoretic approaches in time series analysis*, Physics Reports 441 (2007) 1– 46

[4] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera and Andrea Califano, *ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context*, BMC Bioinformatics 2006, 7(Suppl 1):S7

[5] Young, William C., Adrian E. Raftery, and Ka Y. Yeung *Fast Bayesian inference for gene regulatory networks using ScanBMA*, BMC systems biology 8.1 (2014): 47

[6] Andrea Pinna, Nicola Soranzo, Ina Hoeschele and Alberto de la Fuente, *Simulating systems genetics data with SysGenSIM*, Bioinformatics, Vol. 27 No. 17, 2011, pages 2459–2462

[7] C. W. J. Granger, *Investigating Causal Relations by Econometric Models and Cross-spectral Methods*, Econometrica, Vol. 37 No. 3, 1969, pages 424-438

[8] George Sugihara et al., *Detecting Causality in Complex Ecosystems*, Science, Vol. 338, 2012, pages 496-500

[9] Gideon Schwarz, *Estimating the Dimension of a Model*, The Annals of Statistics, Vol. 6 No. 2, 1978, pages 461-464

[10] Noda, Hino, Tatsuno, Akaho and Murata, *Intrinsic graph structure estimation using graph Laplacian*, Neural Comput. 2014, 26(7), pages 1455-1483

[11] S. Brin and L. Page, *The anatomy of a large-scale hypertextual Web search engine*, Computer Networks and ISDN Systems 1998, 30(1–7)