GRASMOS: Graph Signage Model Selection for Gene Regulatory Networks

Angelina Brilliantova, Hannah Miller, Ivona Bezáková

Rochester Institute of Technology, 102 Lomb Memorial Drive, Rochester, NY 14623
lb9849@rit.edu, hm@mail.rit.edu, ib@cs.rit.edu

Abstract

Signed networks (networks with positive and negative edges) commonly arise in various domains from molecular biology to social media. The edge signs – i.e., *the graph signage* – represent the interaction pattern between the vertices and can provide insights into the underlying system formation process. Generative models considering signage formation are essential for testing hypotheses about the emergence of interactions and for creating synthetic datasets for algorithm benchmarking (especially in areas where obtaining real-world datasets is difficult).

In this work, we pose a novel Maximum-Likelihood-based optimization problem for modeling signages given their topology and showcase it in the context of gene regulation. Regulatory interactions of genes play a key role in the process of organism development, and when broken can lead to serious organism abnormalities and diseases. Our contributions are threefold: First, we design a new class of signage models for a given topology, and, based on the parameter setting, we discuss its biological interpretations for gene regulatory networks (GRNs). Second, we design algorithms computing the Maximum Likelihood – depending on the parameter setting, our algorithms range from closed-form expressions to MCMC sampling. Third, we evaluated the results of our algorithms on synthetic datasets and real-world large GRNs. Our work can lead to the prediction of unknown gene regulations, novel biological hypotheses, and realistic benchmark datasets in the realm of gene regulation.

Introduction

Networks with positive and negative edges (*signed networks*) are ubiquitous across various domains. They work well for situations when the objects modeled as network vertices have positive and negative interactions. Accounting for edge types helps substantially with many important network-related problems, such as missing link prediction (Li, Fang, and Zhang 2017; Li et al. 2020), node ranking (Li, Fang, and Zhang 2019), network synchronization (Monteiro et al. 2022). Signed networks were successfully applied to model social interactions (trust/distrust in Epinions (Xu et al. 2019), epidemic spreading (Li et al. 2021), political interactions between US Congressmen (Thomas, Pang, and Lee 2006), and gene regulation (Mason et al. 2009).

One of the most interesting things about signed networks is the distribution of signs on the edges - i.e., *the signage*. Systems modeled by signed networks rarely exhibit random interactions between their elements - instead, they follow certain interaction patterns, that could provide insights into the underlying formation process. Generative models considering signage formation are essential for testing hypotheses about the emergence of interactions, as well as creating synthetic datasets for algorithm benchmarking (especially in areas where obtaining real-world datasets is difficult).

One of the fields affected by the scarcity of realistic signed network datasets is gene regulation. The diversity of cells, organs, and eventually organisms arises from regulatory interactions between genes through their biochemical products (proteins or RNAs). If a gene product increases the synthesis of a target gene product, the corresponding directed edge is modeled as positive; if it decreases, the edge is modeled as negative. The set of all regulatory interactions along with a set of genes is called a *gene regulatory network* (GRN). GRNs play a key role in the process of organism development, and when broken can lead to serious organism abnormalities and diseases (Alon 2019). Understanding gene regulation mechanisms as well as identifying drug interventions often rely on reconstructing GRNs from the dynamics of the corresponding gene transcripts (Zhang et al. 2017; Lopes-Ramos et al. 2020).

In this work, we designed a new class of signage models for gene regulatory networks along with a maximum-likelihood-based framework assessing their goodness-of-fit. Our signage models work for scenarios in which the graph topology is formed first and later refined with the edge signs. Motivated by underlying biological processes, our models involve latent non-overlapping groups of nodes and the edge signs are generated based on the endpoints' groups. A real-life example is forming social interactions in a closed community (e.g. dormitories) in which one gets acquainted with the people one lives and interacts with and later decides on their attitude to them. Some evolutionary hypotheses suggest a similar origin of gene regulation, with the genome structure defining the gene interactions in the course of species divergence (Bylino, Ibragimov, and Shidlovskii 2020; Wittkopp and Kalay 2012). Signage models facilitate the comparison of node interactions regardless of the underlying topology – this might be useful for graphs generated with different topologies but fol-

lowing the same signage model, allowing topology-agnostic comparisons and testing the plausibility of node interactions hypotheses.

We evaluated the fit of our models using maximum likelihood as an objective criterion, see, for example (Bezáková, Kalai, and Santhanam 2006). Most existing methods evaluating a model's goodness-of-fit rely on high-level network characteristics (degree distribution, diameter, frequency of specific subgraphs, etc.), which might be prone to overfitting and subjectivity in criteria selection especially when real-world data is scarce (as in the case of GRNs). In contrast, likelihood-based goodness-of-fit is a universal approach which does not rely on an arbitrarily chosen network or signage properties. The main challenge of this approach is the computation of the likelihood.

Contributions 1) We design a class of graph signage models with a latent node group assignment, which gets generated first, followed by the edge signage, the distribution of which is conditional on the node group assignment. 2) We pose a new optimization problem for modeling graph signage for gene regulatory networks, GRASMOS [1], which aims to select the model and its parameters that best fit the observed data using Maximum Likelihood. Our algorithmic framework can be used as a guide to design likelihood estimators for other models, facilitating direct comparison of their goodness-of-fit. 3) We analyze the GRASMOS problem space and identify four cases of parameter combinations that have different intuitions and induce different computational complexity of the likelihood computation. We give efficient algorithms for estimating the likelihood corresponding to the given parameters. Depending on the parameter class they belong to, our algorithms range from closed-form expression to MCMC sampling. 4) We obtain a 16-fold reduction in the MCMC algorithm runtime, by identifying parts of the datasets where MCMC can be replaced by exact computation and by accounting for symmetry in the parameter space. 5) We evaluate our framework on two real-world bacteria GRN datasets - *E.coli* and *B.subtilis* (both with thousands of nodes and edges), as well as on synthetic datasets of comparable size.

Related work Works of (Derr, Aggarwal, and Tang 2018; Jung, Park, and Kang 2020) designed generators of signed networks based on the structural balance theory, that capture the distribution of signed triangles. Such generators model a signage jointly with a topology, while in our work, the signage is formed on the basis of the existing topology. These generators do not appear to generate self-loops, while our work admits topologies with self-loops, which play an essential part in GRNs (Burda et al. 2011; Alon 2019). Other works designed algorithms generating a node embedding in signed networks and tested its performance for missing link prediction and node classification (Li, Fang, and Zhang 2017; Li et al. 2020). Such algorithms aim to have high predictive power while in our work we aim for a model with high explanatory power.

[1] https://github.com/Restel/grasmos

Preliminaries

A *signed* graph $G^{\pm} = (V, E, \mathcal{A})$ consists of a directed graph $G = (V, E)$ with node set V and edge set $E \subseteq V \times V$, and a signage function $\mathcal{A} : E \to \{+, -\}$ determining a positive or a negative sign for each edge. Let $n = |V|$ be the number of vertices and $m = |E| = m_+ + m_-$, where m_+ and m_- is the number of positive and negative edges, respectively.

For any $v \in V$ let $\mathrm{out}_s(v)$ be the number of outgoing edges from v of sign $s \in \{+, -\}$, and define $\mathrm{in}_s(v)$ analogously for incoming edges to v. For a sign s, we define $\bar{s}$ as its complementary sign. The total number of outgoing edges from v is $\mathrm{out}(v) = \mathrm{out}_s(v) + \mathrm{out}_{\bar{s}}(v)$. Let $\mathrm{self}_s(v)$ represent the number of s-self-loops of node v.

Graph Signage Model

We propose a graph signage model, where the signs of the edges of a given directed graph $G = (V, E)$ are driven by a random latent node partition C and a parameter matrix ξ. In particular, let S be a set of symbolic node group labels, q be a distribution over S (i.e., $q : S \to [0, 1]$ where $\sum_{s \in S} q(s) = 1$), and ξ be an $|S| \times |S|$ matrix of probabilities $\xi_{x,y}$, where $x, y \in S$. The signage model first randomly creates a latent node partition $C : V \to S$ by assigning each node independently to one of the groups in S according to the distribution q. Then, for each edge $(u, v) \in E$, the model assigns the sign $+$ to this edge with probability $\xi_{C(u),C(v)}$, and the sign $-$ otherwise, obtaining a signed graph $G^{\pm}$. The parameters of the model are combined in the tuple $\Theta = (\xi, q)$.

We define $|S| \times |S|$ probability matrices $\mathcal{P}^+$ and $\mathcal{P}^-$ denoting the probabilities that, based on the node assignment of the end-points of an edge, the edge gets sign $+$ or $-$: For $s_1, s_2 \in S$, let $\mathcal{P}^+_{s_1,s_2} := \xi_{s_1,s_2}$ and $\mathcal{P}^-_{s_1,s_2} := 1 - \xi_{s_1,s_2}$.

Graph Signage Model Selection, GRASMOS

In this section, we formalize our optimization problem, GRASMOS, aimed at finding the parameters of the generative graph signage model with the best explanatory power of the observed data, based on maximum likelihood. Recall that the signage model only generates the signs of the edges, not the topology; in a sense, it "augments" the edges with signs based on the existing graph topology.

Formally, given a signed graph $G^{\pm} = (V, \mathcal{A})$, we are looking for such combination of parameters Θ that has the highest probability of generating edge signs $\mathcal{A}$, denoted $P(\mathcal{A}|\Theta)$. For a fixed node group assignment, C, we can find the corresponding probability of the signs as:

$$\mathcal{L}(\Theta|C) := P(\mathcal{A}|\Theta, C) = \prod_{(u,v) \in E} \mathcal{P}^{\mathcal{A}(u,v)}_{C(u),C(v)} \qquad (1)$$

Since the node group assignment is generated randomly, the probability of interest is a sum of conditional probabilities over all possible node assignments:

$$\mathcal{L}(\Theta) = P(\mathcal{A}|\Theta) = \sum_{C \in \mathcal{C}} P(\mathcal{A}|\Theta, C) \cdot P(C) \qquad (2)$$

Our model and the above expression are reminiscent of the well-known Ising model from statistical physics and its partition function (Ising 1925; Jerrum and Sinclair 1993). However, there are crucial differences between the two models:

the Ising model is defined for undirected graphs and its Hamiltonian does not capture our setting.

GRASMOS aims to find model parameters Θ_{MLE} with the highest probability (or likelihood) of realizing the edge signs $\mathcal{A}$. As is standard for the maximum likelihood approach, for numerical purposes we frame the optimization problem as finding the parameters Θ with the lowest negative log-likelihood:

$$\Theta_{\text{MLE}} = \arg\min_{\Theta}[-\log \mathcal{L}(\Theta)] \tag{3}$$

Analysis of the GRASMOS Parameter Space

GRASMOS is an optimization problem over an $(|S|^2+|S|-1)$-dimensional parameter space, since ξ is of dimensions $|S| \times |S|$ and q is determined by $|S|-1$ probabilities.

While our signage model is general, in this work we focus on modeling GRNs where $S = \{A, R\}$. Therefore, from now on we assume $S = \{A, R\}$ and, therefore, the optimization is over a 5-dimensional parameter space. For each parameter setting Θ, the likelihood computation is potentially an exponential summation over 2^n node group assignments. However, on closer scrutiny, it turns out that points in some areas of the problem space are easier to estimate than the rest and it is the relationship between ξ probabilities that determines the computational difficulty of the point estimation. In this section, we subdivide our main optimization problem (3) into several cases (models) according to the relationship between probabilities in ξ and provide intuition for each model. We present the models from the simplest to the most general.

Node-Oblivious model (NO) For the problem space points with identical ξ entries, the node group partition does not affect the likelihood of Θ, so we can estimate the likelihood of such points analytically. We call this case the *Node-Oblivious* model because the signs of edges are independent of the nodes' groups. Define $\xi = \xi_{A,A} = \xi_{A,R} = \xi_{R,A} = \xi_{R,R}$. Then, $\mathcal{P}^{+}_{s_1,s_2} = \xi$ and $\mathcal{P}^{-}_{s_1,s_2} = 1-\xi$ for every $s_1, s_2 \in S$. This model can be viewed as a signage analog of the well-studied Erdös-Rényi random graph model.

Source-Consistent model (SC) In parameter subspace, where ξ matrix has pairwise identical probabilities $\xi_{A,A} = \xi_{A,R} =: \xi_{A*}$ and $\xi_{R,A} = \xi_{R,R} = \xi_{R*}$. The edge sign probabilities depend only on the source node's group. In the context of gene regulation, that means that a gene mostly performs the regulation of the same sign: an activator gene tends to activate, and a repressor gene represses. We call the parameter instances belonging to this case the *Source-Consistent model*. This observation allows us to reformulate (1) as a product of probabilities of signed outgoing edges over the set of vertices:

$$P(\mathcal{A}|\Theta, C) = \prod_{v \in V} \xi^{\text{out}_+(v)}_{C(v)*} \cdot (1 - \xi_{C(v)*})^{\text{out}_-(v)} \tag{4}$$

Target-Consistent model (TC) Similarly to the previous model, in the *Target-Consistent* model, the following edge probabilities are pairwise equal, i.e. $\xi_{A,A} = \xi_{R,A} =: \xi_{*A}$ and $\xi_{A,R} = \xi_{R,R} =: \xi_{*R}$ and the calculation of $P(\mathcal{A}|\Theta, C)$ is analogous to the SC model, using incoming edges.

Bi-Node-Consistent model (BNC) In the *Bi-Node-Consistent* model, the edge signs depend on the node group assignment of both the source and the target nodes and the probabilities in ξ can be arbitrary. In this parameter subspace, we can see ξ instances inducing assortativity: e.g., whether nodes from the same group tend to have positive edges between each other, and negative edges to the nodes from the other group (Mussmann et al. 2015), (Newman 2002). We reformulate (1) for the likelihood estimation of the BNC parameters as a product of the likelihood contributions of non-self-loop (outgoing) signed edges and self-loop signed edges for each node given node assignment C. Let $l(v, C)^{\text{out}}$ and $l(v, C)^{\text{self}}$ be the likelihood contribution of the outgoing and self-loop edges of node v to the Θ likelihood given C in (1). Thus,

$$l(v, C)^{\text{out}} = \prod_{u: u \neq v, (v,u) \in E} \mathcal{P}^{\mathcal{A}(v,u)}_{C(v),C(u)} \tag{5}$$

$$l(v, C)^{\text{self}} = \xi^{\text{self}_+(v)}_{C(v),C(v)} \cdot (1 - \xi_{C(v),C(v)})^{\text{self}_-(v)} \tag{6}$$

We rewrite (1) with respect to the contribution of each node to the overall likelihood:

$$P(\mathcal{A}|\Theta, C) = \prod_{v \in V} l(v, C)^{\text{out}} \cdot l(v, C)^{\text{self}} \tag{7}$$

Likelihood Estimation of the GRASMOS Model Parameters

In this section, we show how to estimate the likelihood of the GRASMOS model parameters, depending on which of the above-stated models they belong to. The methods we design range in their complexity depending on the computational difficulty of the problem: we provide a closed-form expression for the Node-Oblivious model, a polynomial-time exact algorithm for the source- and Target-Consistent models, and an MCMC sampling algorithm for the Bi-Node-Consistent model. We prove that the models (or, more precisely, their natural generalizations) are *self-reducible* (Jerrum, Valiant, and Vazirani 1986), which allows us to estimate the likelihood for a given Θ through a product of likelihood ratios of progressively smaller instances. For the BNC model, we use sampling to get an approximation of each of the likelihood ratios.

Node-Oblivious Model

For the NO model, from (1) and (2) we get $P(\mathcal{A}|\Theta, C) = P(\mathcal{A}|\Theta) = \xi^{m_+} \cdot (1-\xi)^{m_-}$. Therefore, the optimization problem (3) can be solved analytically: it is maximized for $\xi_{\max} := \frac{m_+}{m_+ + m_-}$. Since q does not contribute to $P(\mathcal{A}|\Theta)$, $\Theta_{\text{MLE}} = (\xi_{\text{MLE}}, q_{\text{MLE}})$, where ξ_{MLE} has all entries equal to $\xi_{\max}$ and q_{MLE} is arbitrary.

Likelihood Estimation through the Product of Likelihood Ratios

The total likelihood of Θ consists of the sum of 2^n likelihoods of Θ conditional on the node assignment (see (2)). We express $\mathcal{L}(\Theta)$ as the product of ratios of pairs of likelihoods on decreasingly smaller spaces of node group assignments:

the reduction in space is achieved by fixing the groups for some nodes.

Let $V = \{v_1, v_2, \ldots, v_n\}$. For $s_1, \ldots, s_j \in S$, we define $\mathcal{C}_j^{[s_1,\ldots,s_j]}$ as the set of node group assignments $C : V \to \{A, R\}$ such that $C(v_i) = s_i$ for every $i \leq j$ (vertices $v_1, \ldots, v_j$ have their group determined by $s_1, \ldots, s_j$). Notice that $\mathcal{C}_0$ is the set of all node group assignments (with no restrictions) and that $|\mathcal{C}_j^{[s_1,\ldots,s_j]}| = 2^{n-j}$ for any $s_1, \ldots, s_j$.

For our self-reducibility approach, let us fix a "master" node assignment $\tilde{C} : V \to \{A, R\}$ that gradually more and more vertices will adhere to. Let

$$Z_j := \sum_{C_j \in \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_j)]}} P(\mathcal{A}|\Theta, C_j) \cdot P(C_j|j), \quad (8)$$

where $P(C_j|j)$ is the probability that vertices $v_{j+1}, \ldots, v_n$ get the node assignment given by C_j. Notice that $P(C_j|j)$ is a probability distribution over the node group assignment subspace $\mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_j)]}$, and, therefore, Z_j is the likelihood of Θ restricted to this subspace.

Our overall goal is to estimate $\mathcal{L}(\Theta) = Z_0$. We do this via the following product of likelihood ratios:

$$\frac{Z_1}{Z_0} \cdot \frac{Z_2}{Z_1} \cdot \frac{Z_3}{Z_2} \ldots \cdot \frac{Z_n}{Z_{n-1}} = \frac{Z_n}{Z_0}. \quad (9)$$

Notice that $Z_n = P(\mathcal{A}|\Theta, \tilde{C})$ can be easily computed via (1). Therefore, if we estimate each of the ratios $\sigma_j := \frac{Z_j}{Z_{j-1}}$, we can compute Z_0 as follows:

$$P(\mathcal{A}|\Theta) = Z_0 = \frac{Z_n}{\prod_{j=1}^{n} \sigma_j} \quad (10)$$

For numerical stability, we will choose $\tilde{C}$ so that each ratio σ_j is reasonably far from 0.

Theorem 1. *There exists $\tilde{C}$ such that $\sigma_j \geq 1/2$ for every $j \in \{1, \ldots, n\}$.*

Proof. We will define $\tilde{C}$ inductively. First, notice that the ratio σ_j does not rely on the entire $\tilde{C}$ but only on its restriction to $\{v_1, \ldots, v_j\}$. For $j \in \{1, \ldots, n\}$, suppose $\tilde{C}(v_1), \ldots, \tilde{C}(v_{j-1})$ have been chosen so that $\sigma_i \geq 1/2$ for every $i < j$. We will show how to choose $\tilde{C}(v_j)$ so that $\sigma_j \geq 1/2$ by considering the two possible cases: $\tilde{C}(v_j)$ can be either an activator or a repressor. For $s \in \{A, R\}$, define Z_j^s using (8) with $[\tilde{C}(v_1), \ldots, \tilde{C}(v_{j-1}), s]$. Let $\sigma_j^s := \frac{Z_j^s}{Z_{j-1}}$. Recall that $P(C|j-1) = q^{A(C,j)}(1-q)^{n-j-A(C,j)}$, where $A(C, j) := |\{i \mid C(V_i) = A,\ i \geq j\}|$ stands for the number of activators among $v_j, \ldots, v_n$ in C. Notice that

$$\begin{aligned} Z_{j-1} &= \sum_{C \in \mathcal{C}_{j-1}^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_{j-1})]}} P(\mathcal{A}|\Theta, C) \cdot P(C|j-1) \\ &= Z_j^A q + Z_j^R (1-q). \end{aligned}$$

Therefore, $q\sigma_j^A + (1-q)\sigma_j^R = 1$. This equation cannot hold if both $\sigma_j^A < 1/2$ and $\sigma_j^R < 1/2$. Therefore, choosing $\tilde{C}(v_j) := \arg\max_{s \in \{A,R\}} \sigma_j^s$ ensures that $\sigma_j = \max\{\sigma_j^A, \sigma_j^R\} \geq 1/2$. □

The proof builds $\tilde{C}$ constructively but the algorithmic efficiency is unclear (it involves summations over exponentially many terms). In the following sections, we will show how to construct $\tilde{C}$ and compute all the $\sigma_j^* := \max\{\sigma_j^A, \sigma_j^R\}$ (and thus $\mathcal{L}(\Theta)$) efficiently.

Source-Consistent and Target-Consistent Models

We show how to estimate the ratio of likelihoods σ_j^* for the SC models in linear time. Our algorithms rely on the fact that, in the SC model, the group assignment of any node, $\tilde{C}(v_j)$, only affects the probabilities of edges connected to v_j. Therefore, the calculation of σ_j^* depends only on node v_j and its incident edges. The proof and the analogous computation for the TC model are included in the full version (Brilliantova, Miller, and Bezáková 2022).

Theorem 2. *When Θ belongs to the Source-Consistent model, for every $j \in \{1, \ldots, n\}$ and every $\tilde{C} : V \to \{A, R\}$:*

$$\sigma_j^* = \max\{\frac{1}{q + \alpha_j(1-q)}, \frac{\alpha_j}{q + \alpha_j(1-q)}\},$$

where

$$\alpha_j := (\frac{\xi_{R*}}{\xi_{A*}})^{\text{out}_+(v_j)} \cdot (\frac{1-\xi_{R*}}{1-\xi_{A*}})^{\text{out}_-(v_j)}.$$

Corollary 3. *For the Source-Consistent model, the likelihood of Θ, $\mathcal{L}(\Theta)$, can be calculated in $\mathcal{O}(n+m)$ time.*

MCMC Sampling for Likelihood Estimation of the Bi-Node-Consistent Model

In contrast to the SC and TC models, in which an edge sign depends only on one endpoint and the ratio of likelihoods σ_j^* can then be computed analytically, we are not aware of any analytical approach for σ_j^* in the BNC model where both edge end-points influence the sign of the edge. The main difficulty arises from the fact that the calculation of σ_j^* under BNC might depend on the colors of the remaining $n - j - 1$ nodes, which, at that point, are still unassigned in the course of the algorithm. Instead of exact computation, we estimate each σ_j^* via a Markov Chain Monte-Carlo (MCMC) sampling of node group assignments on the subspace corresponding to Z_j, with first j vertices assigned to groups.

Suppose $\tilde{C}(v_1), \ldots, \tilde{C}(v_j)$ has been already defined. Let us define $w_j(C_j) := P(\mathcal{A}|\Theta, C_j) \cdot P(C_j|j)$ as the weight of the node group assignment $C_j \in \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_j)]}$. To compute the likelihood $Z_j = \sum_{C_j \in \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_j)]}} w_j(C_j)$, we will randomly generate C_j, with probability proportional to its weight. Therefore, the stationary distribution of the node group assignments should be $\mu_j(C_j) := w_j(C_j)/Z_j$. To obtain this stationary distribution, we use the Metropolis-Hasting technique (Metropolis et al. 1953; Hastings 1970), with state space $\Omega_j := \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_j)]}$ and the Markov chain transitions $\tau : \Omega_j \times \Omega_j \to [0, 1]$ defined as follows. Let C_j be the current state.

- Choose a uniformly random $z \in \{j+1, \ldots, n\}$. Let C_j' be identical C_j, except $C_j'(v_z) = \bar{C}_j(v_z)$ (the assignment of v_z is opposite in C_j and C_j').

- With probability $\min\{1, \frac{w_j(C'_j)}{w_j(C_j)}\}$ move to state C'_j. Otherwise, stay at C_j.

Therefore, $\tau(C_j, C'_j) = \frac{1}{(n-j)} \min\{1, \frac{w_j(C'_j)}{w_j(C_j)}\}$ and $\tau(C_j^{(1)}, C_j^{(2)}) = 0$ for all other $C_j^{(1)}, C_j^{(2)}$ where $C_j^{(1)} \neq C_j^{(2)}, C_j^{(1)} \neq C_j'^{(2)}$ (the self-loops $\tau(C_j, C_j)$ correspond to the remaining probability, so that τ is a stochastic matrix).

The underlying transition graph is analogous to the hypercube with $n - j$ dimensions, and, therefore, the state space is connected (i.e., we can get from every state to every other state using transitions of the Markov chain). This Markov chain is also aperiodic due to the presence of self-loop transitions and, therefore, it has a unique stationary distribution. The Metropolis-Hastings technique ensures that this stationary distribution is exactly the distribution μ_j, i.e., proportional to the weights $w_j(C_j)$.

Following the earlier outline (see (9) and (10)), we will be estimating $\sigma_j = \frac{Z_j}{Z_{j-1}}$. Recall that, for numerical stability, we wanted $\sigma_j \geq 1/2$. We showed that $\tilde{C}(v_j) = \arg\max_{s \in \{A,R\}} \sigma_j^s$ yields $\sigma_j = \max\{\sigma_j^A, \sigma_j^R\} \geq 1/2$. We will estimate both σ_j^A and σ_j^R simultaneously by generating samples from $\Omega_{j-1} = \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_{j-1})]}$. A sample C_{j-1} with $C_{j-1}(v_{j-1}) = X$ will contribute to the σ_j^X computation, for $X \in \{A, R\}$.

Let C_{j-1} be a random sample from Ω_{j-1}, drawn according to μ_{j-1}. Define $f_A : \Omega_{j-1} \to \{0, 1\}$ as the indicator function that the corresponding node assignment assigned v_j as an activator: $f_A(C) = 1$ if and only if $C(v_j) = A$. Then,

$$\begin{aligned}
&\mathbf{E}_{\mu_{j-1}}[f_A(C_{j-1})] \\
&= \sum_{C \in \Omega_{j-1}} \mu_{j-1}(C) f_A(C) = \sum_{C \in \Omega_{j-1} : C(v_j) = A} \frac{w_{j-1}(C)}{Z_{j-1}} \\
&= \frac{1}{Z_{j-1}} \sum_{C \in \mathcal{C}_j^{[\tilde{C}(v_1),\ldots,\tilde{C}(v_{j-1}),A]}} P(\mathcal{A}|\Theta, C) \cdot P(C|j-1) \\
&= \frac{q Z_j^A}{Z_{j-1}} = q\sigma_j^A.
\end{aligned}$$

Therefore, we can use the expectation of f_A to estimate σ_j^A (and, similarly, we can define f_R and estimate σ_j^R since $\mathbf{E}_{\mu_{j-1}}[f_R(C_{j-1})] = (1-q)\sigma_j^R$). The accuracy of the estimate increases with the number of samples: we will draw k independent samples $C_j^{(1)}, C_j^{(2)}, \ldots, C_j^{(k)}$ and compute the average $f_A(C_j)$, namely $\frac{1}{k}\sum_{i=1}^{k} f_A(C_j^{(i)})$. We use the same set of samples for the average f_A and the average f_R computation (each sample contributes to either f_A or f_R). Since the expectation $\mathbf{E}_{\mu_{j-1}}[f_A(C_{j-1})]$ corresponds to the probability of drawing a sample with v_j assigned as an activator, we draw such sample with probability $\frac{qZ_j^A}{Z_{j-1}}$, and we draw a sample with v_j as a repressor with probability $\frac{(1-q)Z_j^R}{Z_{j-1}}$. Since these two types of samples cover the state space, at least one of the types will be drawn with probability $\geq 1/2$. This means that we will very likely have a sufficient number of samples of at least one of the types, which will allow us to estimate the corresponding expected value closely. Since we do not know which of the types is more likely, we generate k samples and estimate both the average f_A and the average f_R. Then, we choose the larger average and define the corresponding A or R as $\tilde{C}(v_j)$. Then, divide the larger average by q or $(1-q)$ as appropriate, obtaining an estimate on σ_j.

As a side remark, this σ_j might potentially be different than $\max\{\sigma_j^A, \sigma_j^R\}$ but we still have $\sigma_j \geq 1/2$ since $q\sigma_j^A \geq 1/2$ or $(1-q)\sigma_j^R \geq 1/2$.

Next, we discuss the accuracy of this estimate. Suppose we aim to be within a $(1+\varepsilon)$-factor of the true $\mathcal{L}(\Theta)$, for some small ε. Since we have n self-reducibility steps (i.e., n ratios σ_j to estimate, see (9) and (10)) and the estimated quantities are larger than $1/2$, we can follow the derivation in (Jerrum 2003)[page 26] almost verbatim to obtain the desired number of samples per estimate. However, this number of samples is rather high for the datasets of our size, so we (successfully) employ convergence heuristics for the estimates.

Type	# nodes	# edges	# + edges	# − edges
Regulon	1922	4265	2256	2000
SubtiWiki	2563	5283	3436	1847
Synthetic	2000	3127*	NA**	NA**

Table 1: Network characteristics of real-world and synthetic datasets. * The median over 40 generated topologies from DSF ** Varied substantially based on Θ-generator

Algorithmic Speed-Up of MCMC

We combine the MCMC approach with exact calculation to achieve significant speed-up for real-world datasets. GRNs, like many other real-world networks, are not dense: they usually have a few highly connected nodes (called *hubs* in network science) and many low-degree nodes. For such low-degree nodes, the use of MCMC is unnecessary and computationally wasteful: as soon as all of their adjacent vertices are assigned, the ratio of likelihoods for such nodes can be calculated analytically. For such v_j, this theorem shows how to compute σ_j^* exactly in time proportional to v_j's degree. The proof is in the full version (Brilliantova, Miller, and Bezáková 2022).

Theorem 4. *Let $C : V \to \{A, R, \text{undefined}\}$ be a partial node assignment. Reorder the vertices so that $v_1, \ldots, v_{j-1}$ are assigned by C to either A or R, and all other vertices are undefined. Suppose that all of v_j's neighbors are already assigned by C. Then σ_j^* can be calculated as:*

$$\sigma_j^* = \max\{\frac{q\beta_j^A}{q\beta_j^A + (1-q)\beta_j^R}, \frac{(1-q)\beta_j^R}{q\beta_j^A + (1-q)\beta_j^R}\}, \quad (11)$$

$$\beta_j^X(C) = \textstyle\prod_{i<j:(v_i,v_j)\in E} \mathcal{P}_{C(v_i),X}^{\mathcal{A}(v_i,v_j)} \prod_{i<j:(v_j,v_i)\in E} \mathcal{P}_{C(X,v_i))}^{\mathcal{A}(v_j,v_i)}.$$

Empirical Evaluation

Data Collection and Setup

Synthetic datasets To validate that our algorithms can correctly identify the GRASMOS parameters, we created synthetic datasets with known Θ, resembling real-world GRNs

by their network parameters and size. The characteristics of all synthetic and real-world datasets we used are shown in Table 1. Our evaluation pipeline had two steps: generation and reconstruction. During the generation part, we generated network topologies from the Directed Scale-Free (DSF) model, known to capture the characteristics of GRNs well (Van den Bulcke et al. 2006), then using several different Θ parameters we sampled the node group partition and assigned $+$ and $-$ signs to the edges.

For illustrative purposes, in the Results section below we present our validation for the Source-Consistent model: We chose several Θ (we refer to them as Θ-generators) from this model and generated corresponding signages for our graph. During the reconstruction part, we tested multiple candidate Θ (Θ-candidates) belonging to the SC, TC, and NO models and checked the proportion of samples in which the candidate with the best likelihood Θ_{MLE} matched the Θ-generator. We also calculated the L_1-norm between the Θ-generator and Θ_{MLE}. The parameters within the Θ-candidates varied between 0.1 and 0.9, with an increment step of 0.1. We tested 4 different Θ-generators in the generation part and for each of 1458 of Θ-candidates in the reconstruction part. To account for stochasticity we repeated the pipeline 10 times. Tests for additional Θ-generators can be found in the full version (Brilliantova, Miller, and Bezáková 2022).

Real-world GRNs We used two public databases: RegulonDB (Santos-Zavaleta et al. 2019) and SubtiWiki (Pedreira, Elfmann, and Stülke 2022; Flórez et al. 2009). Both databases contain experimentally validated information about gene regulatory interactions and their type (activation/repression) for bacteria species: Regulon for *Escherichia coli* and SubtiWiki for *Bacillus subtilis*. We only left the entries associated with transcriptional gene regulation. From both datasets, we filtered out regulation edges of unknown types and duplicated edges. Regulatory interactions can be context-dependent: under different conditions, the same gene can either activate or repress the target gene (Ong and Corces 2011). Our model does not account for such scenarios, so we had to filter out duplicated edges.

For fitting the GRASMOS parameters of real-world GRNs, we: used a fine-grained exploration of parameters varying in $[0.1, 0.9]$ with an increment of 0.05 belonging to the SC, TC, and NO models; and, for complexity reasons, we evaluated the BNC model subspace using a coarse-grid with each parameter varying in $\{0.25, 0.5, 0.75\}$. For Θs belonging to multiple models, we compared the results from these approaches, made sure that they were consistent, and identified a good candidate to run a "refined" BNC search with 432 additional Θ-candidates in the vicinity (searching through Θs roughly within ± 0.125 from the identified candidate).

We estimated $\mathcal{L}(\Theta)$ of the BNC parameters using a parallel implementation on the RIT research computing cluster. We used up to 432 nodes with each core estimating a likelihood of a single Θ-candidate via MCMC sampling. Each core is equipped with Intel®Xeon®Gold 6150 CPU @ 2.70GHz. The RAM upper limit for our computation was 2048 MB.

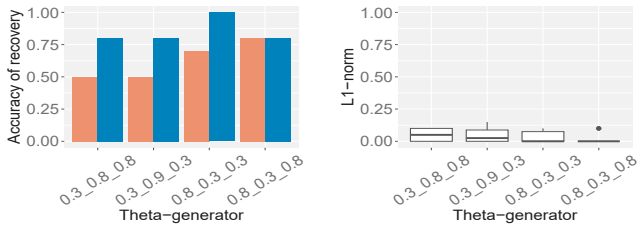

Figure 1: Left: The absolute accuracy of Θ (red) and ξ-generators (blue) recovery. Right: The average and the lower and the upper-quartiles of the L_1-norm between the Θ-generator and the best Θ-candidate. In both figures, Θ-generators have the form of (ξ_{A*}, ξ_{R*}, q)

.

Results and Discussion

Synthetic Data

For synthetic datasets, we tested how well our framework based on the total likelihood reconstructs the Θ values that were used to generate an edge signage instance. We found that, overall, it was able to reconstruct the Θ-generator well. Figure 1(left) shows the percentage of test cases when the Θ-generator equaled the reconstructed Θ_{MLE} exactly, and when the ξ-portion of Θ and Θ_{MLE} were identical, respectively. Reconstruction of the ξ-parameters was particularly successful. Moreover, even when Θ and Θ_{MLE} differed, they were very close in their L_1-norm, see Figure 1(right).

Results on Real-World GRNs

Table 2 presents 5 best Θ candidates per model for the SubtiWiki and Regulon datasets. For both datasets, the BNC parameters had the best performance among all models. However, some of the top BNC parameters belonged to the SC model for SubtiWiki. BNC's winning score is not surprising as it generalizes the other models. However, when another model overtakes BNC (as in the case of the SC model and Subtiwiki), that might lead to biological hypotheses about the underlying processes, which is our main reason for considering the simpler models. The source-consistency hypothesis corresponds well to the existence of *operons* – groups/clusters of bacterial genes that are co-located on the DNA strand and controlled by the same gene-regulator (Salgado et al. 2001). Interestingly, in the case of Regulon, none of the top-5 Θ belonged to the SC model. Moreover, all of the top parameters implied that genes tend to activate the genes from groups other than they belong to. This observation might corroborate the existence of feed-forward transcriptional control with interchanging edge signs suggested for certain GRNs (Sasse and Gerber 2015) and raises the question of whether the bacteria species corresponding to these two datasets indeed have different gene regulation mechanisms, or the discrepancies should be explained by other factors.

We assessed the accuracy of the MCMC sampling for the likelihood estimation of those Θ instances for which we have the exact solution (i.e., Θ in the SC, TC, and NO models). We aimed to have a $(1 + 1/n)$-multiplicative accuracy, i.e., at most 0.2% likelihood error for our datasets. We used

model	ξ_{AA}	ξ_{AR}	ξ_{RA}	ξ_{RR}	q	$\mathcal{L}(\Theta)$
BNC	0.99	0.99	0.20	0.15	0.40	557.12
BNC/SC	0.99	0.99	0.15	0.15	0.40	557.71
BNC	0.99	0.99	0.20	0.15	0.50	558.71
BNC/SC	0.99	0.99	0.15	0.15	0.50	558.93
BNC	0.99	0.99	0.15	0.20	0.40	559.24
SC	0.95	0.95	0.10	0.10	0.45	581.71
SC	0.95	0.95	0.10	0.10	0.50	581.88
SC	0.95	0.95	0.10	0.10	0.40	582.42
SC	0.95	0.95	0.10	0.10	0.55	582.91
SC	0.95	0.95	0.15	0.15	0.45	583.49
NO	0.65	0.65	0.65	0.65	NA	1484.93
TC	0.65	0.70	0.65	0.70	0.95	1485.03
TC	0.70	0.65	0.70	0.65	0.05	1485.03
TC	0.60	0.65	0.60	0.65	0.05	1485.07
TC	0.65	0.60	0.65	0.60	0.95	1485.07
TC	0.65	0.70	0.65	0.70	0.90	1485.19

model	ξ_{AA}	ξ_{AR}	ξ_{RA}	ξ_{RR}	q	$\mathcal{L}(\Theta)$
BNC	0.70	0.80	0.20	0.15	0.50	1127.74
BNC	0.70	0.80	0.25	0.10	0.50	1128.75
BNC	0.70	0.80	0.20	0.10	0.50	1129.03
BNC	0.70	0.80	0.25	0.15	0.50	1129.74
BNC	0.75	0.75	0.20	0.15	0.50	1129.84
SC	0.75	0.75	0.15	0.15	0.50	1130.33
SC	0.75	0.75	0.20	0.20	0.50	1130.45
SC	0.75	0.75	0.15	0.15	0.55	1130.51
SC	0.75	0.75	0.20	0.20	0.45	1130.78
SC	0.75	0.75	0.20	0.20	0.55	1130.91
NO	0.53	0.53	0.53	0.53	NA	1277.51
TC	0.55	0.50	0.55	0.50	0.60	1412.39
TC	0.50	0.55	0.50	0.55	0.40	1412.39
TC	0.55	0.50	0.55	0.50	0.55	1412.40
TC	0.50	0.55	0.50	0.55	0.45	1412.40
TC	0.55	0.50	0.55	0.50	0.65	1412.44

Table 2: Five best Θs per model for SubtiWiki (top) and Regulon (bottom) dataset. The lower $\mathcal{L}(\Theta)$, the better.

this target accuracy for our computation of the number of needed samples. In most cases, our MCMC computations were within the target accuracy. In those cases when we were off by more than $(1 + 1/n)$, the corresponding Θ-candidates had $\mathcal{L}(\Theta)$ far from $\mathcal{L}(\Theta_{\text{MLE}})$, and in the coarse search, the Θ-candidate with the smallest $\mathcal{L}(\Theta)$ was close to $\mathcal{L}(\Theta_{\text{MLE}})$. This meant that despite having more inaccurate $\mathcal{L}(\Theta)$ estimates for these (few) Θs than we hoped for, the coarse search eliminated them from consideration and thus eliminated the inaccuracies.

As for the running time, the analytical calculation of some vertices (Section Algorithmic speed-up of MCMC) and the reduction of the parameter space by half (discussed in the full version of this paper) resulted in 16x actual speed-up for BNC Θ parameters compared to the naive approach. Due to the parallelization of the grid search, the total runtime of fitting the BNC parameters was around 2 days for Regulon and 4 days for the SubtiWiki.

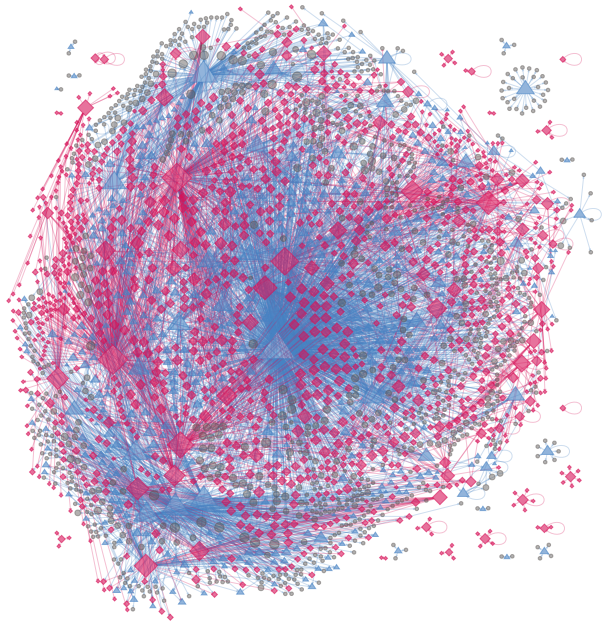

Figure 2: SubtiWiki signed GRN, with added node colors according to the "master" node assignment $\tilde{C}$ corresponding to Θ_{MLE} (SC model). Red diamonds - repressors, blue triangles - activators in $\tilde{C}$, gray circles = vertices where the assignment of $\tilde{C}$ is ambiguous, corresponding to probability exactly $1/2$ of being activator or repressor in the corresponding node assignment subspace.

Conclusion

In this work, we present a novel Maximum-Likelihood-based graph signage model selection problem for gene regulation, GRASMOS, developed a fitting framework for the problem and showcased its usage for two gene regulatory networks of $B.subtilis$ and $E.coli$. Our graph signage models and the model selection framework opened up a plethora of directions for future research. Among them: 1) Does the high explanatory power of the Θ_{MLE} translate into high predictive power? For our models to have high predictive power, we need to be able to reconstruct the hidden node assignment. We used the master node assignment $\tilde{C}$ (visually shown for SubtiWiki dataset in Figure 2) to estimate the likelihood, but how is it related to the "optimal" node assignment $C_{MLE} = \arg\max_{C \in \mathcal{C}} \mathcal{L}(\Theta|C)$? 2) Is our framework robust when there are missing edges? What is the percentage of the missing edges it can handle?

Additionally, we acknowledge that the signage can depend on the topology. A natural next step for this work is to compare the likelihood of models that generate the signage jointly with the topology, and an approach that uses existing topology generators followed by our signage model.

Acknowledgements

The authors thank the anonymous reviewers for their insightful comments and suggestions and acknowledge Research

Computing at the Rochester Institute of Technology for providing computational resources and support. The authors are partially supported by NSF awards DUE-1821459 and DUE-1819546.

References

Alon, U. 2019. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. CRC Press.

Bezáková, I.; Kalai, A.; and Santhanam, R. 2006. Graph Model Selection using Maximum Likelihood. In *Proceedings of the 23rd International Conference on Machine learning*, 105–112.

Brilliantova, A.; Miller, H.; and Bezáková, I. 2022. GRASMOS: Graph Signage Model Selection for Gene Regulatory Networks. *arXiv preprint arXiv:2211.09642*.

Burda, Z.; Krzywicki, A.; Martin, O. C.; and Zagorski, M. 2011. Motifs Emerge from Function in Model Gene Regulatory Networks. *Proceedings of the National Academy of Sciences*, 108(42): 17263–17268.

Bylino, O. V.; Ibragimov, A. N.; and Shidlovskii, Y. V. 2020. Evolution of Regulated Transcription. *Cells*, 9(7): 1675.

Derr, T.; Aggarwal, C.; and Tang, J. 2018. Signed Network Modeling based on Structural Balance Theory. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 557–566.

Flórez, L. A.; Roppel, S. F.; Schmeisky, A. G.; Lammers, C. R.; and Stülke, J. 2009. A Community-curated Consensual Annotation that is Continuously Updated: the *Bacillus subtilis* Centered Wiki, SubtiWiki. *Database*, 2009. [Database version: 4, download date: 06/29/2022].

Hastings, W. K. 1970. Monte Carlo Sampling Methods using Markov Chains and their applications. *Biometrika*, 57(1): 97–109.

Ising, E. 1925. Beitrag zur theorie des ferromagnetismus. *Zeit. Physik*, 31.

Jerrum, M. 2003. *Counting, Sampling and Integrating: Algorithms and Complexity*. Birkhäuser Basel. ISBN 978-3-7643-6946-0.

Jerrum, M.; and Sinclair, A. 1993. Polynomial-time Approximation Algorithms for the Ising Model. *SIAM Journal on computing*, 22(5): 1087–1116.

Jerrum, M.; Valiant, L. G.; and Vazirani, V. V. 1986. Random Generation of Combinatorial Structures from a Uniform Distribution. *Theor. Comput. Sci.*, 43: 169–188.

Jung, J.; Park, H.-M.; and Kang, U. 2020. BalanSiNG: Fast and Scalable Generation of Realistic Signed Networks. In *EDBT*, 193–204.

Li, H.-J.; Xu, W.; Song, S.; Wang, W.-X.; and Perc, M. 2021. The Dynamics of Epidemic Spreading on Signed Networks. *Chaos, Solutions & Fractals*, 151: 111294.

Li, X.; Fang, H.; and Zhang, J. 2017. Rethinking the Link Prediction Problem in Signed Social Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Li, X.; Fang, H.; and Zhang, J. 2019. Supervised User Ranking in Signed Social Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 184–191.

Li, Y.; Tian, Y.; Zhang, J.; and Chang, Y. 2020. Learning Signed Network Embedding via Graph Attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 4772–4779.

Lopes-Ramos, C. M.; Kuijjer, M.; Glass, K.; DeMeo, D.; and Quackenbush, J. 2020. Regulatory Networks of Liver Carcinoma Reveal Sex Specific Patterns of Gene Regulation. *Cancer Research*, 80(16 Supplement): 6569–6569.

Mason, M. J.; Fan, G.; Plath, K.; Zhou, Q.; and Horvath, S. 2009. Signed Weighted Gene Co-expression Network Analysis of Transcriptional Regulation in Murine Embryonic Stem Cells. *BMC Genomics*, 10(1): 1–25.

Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; and Teller, A. H. 1953. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(1087).

Monteiro, H. S.; Leifer, I.; Reis, S. D.; Andrade Jr, J. S.; and Makse, H. A. 2022. Fast Algorithm to Identify Minimal Patterns of Synchrony through Fibration Symmetries in Large Directed Networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 32(3): 033120.

Mussmann, S.; Moore, J.; Pfeiffer, J.; and Neville, J. 2015. Incorporating Assortativity and Degree Dependence into Scalable Network Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Newman, M. E. 2002. Assortative Mixing in Networks. *Physical review letters*, 89(20): 208701.

Ong, C.-T.; and Corces, V. G. 2011. Enhancer function: New Insights into the Regulation of Tissue-specific Gene Expression. *Nature Reviews Genetics*, 12(4): 283–293.

Pedreira, T.; Elfmann, C.; and Stülke, J. 2022. The Current State of Subti Wiki, the Database for the Model Organism *Bacillus subtilis*. *Nucleic Acids Research*, 50(D1): D875–D882.

Salgado, H.; Santos-Zavaleta, A.; Gama-Castro, S.; Millán-Zárate, D.; Díaz-Peredo, E.; Sánchez-Solano, F.; Pérez-Rueda, E.; Bonavides-Martínez, C.; and Collado-Vides, J. 2001. RegulonDB (version 3.2): Transcriptional Regulation and Operon Organization in *Escherichia coli* K-12. *Nucleic Acids Research*, 29(1): 72–74.

Santos-Zavaleta, A.; Salgado, H.; Gama-Castro, S.; Sánchez-Pérez, M.; Gómez-Romero, L.; Ledezma-Tejeida, D.; García-Sotelo, J. S.; Alquicira-Hernández, K.; Muñiz-Rascado, L. J.; and Peña-Loredo, P. 2019. RegulonDB v 10.5: Tackling Challenges to Unify Classic and High Throughput Knowledge of Gene Regulation in *E. coli* K-12. *Nucleic Acids research*, 47(D1): D212–D220. [Database release: 10.9, download date: 06/29/2021].

Sasse, S. K.; and Gerber, A. N. 2015. Feed-forward Transcriptional Programming by Nuclear Receptors: Regulatory Principles and Therapeutic Implications. *Pharmacology & Therapeutics*, 145: 85–91.

Thomas, M.; Pang, B.; and Lee, L. 2006. Get out The Vote: Determining Support or Opposition From Congressional Floor-debate Transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 327–335.

Van den Bulcke, T.; Van Leemput, K.; Naudts, B.; van Remortel, P.; Ma, H.; Verschoren, A.; De Moor, B.; and Marchal, K. 2006. SynTReN: a Generator of Synthetic Gene Expression Data for Design and Analysis of Structure Learning Algorithms. *BMC Bioinformatics*, 7(1): 1–12.

Wittkopp, P. J.; and Kalay, G. 2012. Cis-regulatory Elements: Molecular Mechanisms and Evolutionary Processes Underlying Divergence. *Nature Reviews Genetics*, 13(1): 59–69.

Xu, P.; Hu, W.; Wu, J.; and Du, B. 2019. Link Prediction with Signed Latent Factors in Signed Social Networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1046–1054.

Zhang, Z.; Lei, A.; Xu, L.; Chen, L.; Chen, Y.; Zhang, X.; Gao, Y.; Yang, X.; Zhang, M.; and Cao, Y. 2017. Similarity in Gene-regulatory Networks Suggests that Cancer Cells Share Characteristics of Embryonic Neural Cells. *Journal of Biological Chemistry*, 292(31): 12842–12859.