IMA Journal of Complex Networks (2022) Page 1 of 10
doi:10.1093/comnet/xxx000

Computing Minimal Boolean Models of Gene Regulatory Networks

GUY KARLEBACH*
The Jackson Laboratory for Genomic Medicine
*Corresponding author: guy.karlebach@jax.org

PETER N ROBINSON
The Jackson Laboratory for Genomic Medicine

[Received on 19 April 2022]

Models of Gene Regulatory Networks (GRNs) capture the dynamics of the regulatory processes that occur within the cell as a means to understand the variability observed in gene expression between different conditions. Arguably the simplest mathematical construct used for modeling is the Boolean network, which dictates a set of logical rules for transition between states described as Boolean vectors. Due to the complexity of gene regulation and the limitations of experimental technologies, in most cases knowledge about regulatory interactions and Boolean states is partial. In addition, the logical rules themselves are not known a-priori. Our goal in this work is to present a methodology for inferring this information from the data, and to provide a measure for comparing network states under different biological conditions. We present a novel methodology for integrating experimental data and performing a search for the optimal consistent structure via optimization of a linear objective function under a set of linear constraints. We also present a statistical approach for testing the similarity of inferred network states under different conditions. Finally, we extend our methodology into a heuristic that can handle large datasets that are generated by single-cell RNA-Sequencing(scRNA-Seq). We demonstrate the effectiveness of these tools using a public scRNA-Seq dataset and the GRN that is associated with it. Our methodology will enable researchers to obtain a better understanding of the function of gene regulatory networks and their biological role.

Keywords: differential network state, state inference, minimal structure

1. Introduction

Maintenance of cellular functions requires the orchestration of many interleaving processes over time and space. A Gene Regulatory Network (GRN) is a set of genes such that the present state of their expression trajectory can be predicted from past states via the regulatory relationships between the genes. As a model that can display complex behaviour and at the same time is straightforward to specify, GRNs have been used to describe the regulation of processe as different as cell differentiation, circadian clocks and diauxic shift [3, 11, 18]. Consequently, many methods for reconstructing GRNs from experimental data at varying levels of detail have been proposed [5, 7, 17]. Arguably the most basic formulation, the Boolean network, describes gene expression states as Boolean values and changes in those levels as Boolean functions [9]. While the simplicity of this model imposes a certain level of abstraction, it also makes it applicable to a broader range of datasets and simplifies its analysis. Interestingly, despite the relative simplicity of Boolean networks, fitting a Boolean network to a gene expression dataset given a set of regulatory relationships is an NP-Hard problem [8]. In practice the exact regulatory relationships are not known, which can result in redundant regulators after fitting. A

© The author 2022.

possible remedy to this problem is impose non-redundant logic as a requirement in the solution of the fitting problem [7, 16]. However, this approach contradicts the widely accepted principle that a simpler model that provides the same degree of fit to the data is preferable to a more complex one [1, 12, 15]. In this paper we present a novel algorithm for fitting a Boolean network to a gene expression dataset that addresses the problem of redundant regulators. In addition, we provide a method to compute the statistical significance of difference between network states under different conditions. We apply our methodology to a gene expression dataset of single-cell RNA Sequencing from mouse embryonic midbrain.

2. Methods

Optimal Fit of Boolean Networks

A gene expression dataset consists of a N $\times$ M matrix where N corresponds to the number of genes whose expression level was measured and M corresponds to the number of experiments. In a typical dataset N $\gg$ M. The expression values in the matrix can be discrete or continuous, depending on the experimental technology that was used for generating the data, but higher values always represent a higher expression level. In order to map these values to Boolean values one needs to label each observation as belonging to a state of low or high expression. Since the proposed methodology is independent of the choice of a mapping, in the rest of this section we will assume that the mapping has already been applied to the data.

A trajectory of a Boolean network is a sequence of states such that each state except for the first state in the sequence is derived from the previous state using a set of Boolean functions. Each Boolean function determines the value of exactly one gene, and its inputs are the states of any number of genes in the network. Usually, it is assumed that the number of inputs is small compared to the total number of genes. The regulatory relationships of a Boolean network can be illustrated as edges from inputs to outputs in a directed graph, called a regulation graph. A gene that has an outgoing edge to another gene is referred to as a regulator, and a gene with an incoming edge as a target (a gene can be both a regulator and a target). A steady state is a state that repeats itself in a trajectory indefinitely unless perturbed by external signals, i.e. signals that are not part of the network. In a typical gene expression dataset the experiments correspond to a single time point, and therefore the network is assumed to be in a steady state in each experiment. For simplicity of description we assume in the rest of this section that the network is in steady state, however the algorithm presented here is applicable to any type of data.

Discrepancies between a dataset and a network model occur when the Boolean values in an experimental dataset do not agree with any network trajectory due to experimental noise. This presents a difficulty if the network model is not known a-priori, since enumerating all possible networks is infeasible. Formally, let $C_{g_i,j}$ denote the Boolean value of gene g_i in experiment j, and let e_{g_1,g_2} denote a directed edge between genes g_1 and g_2. We say that the data contains a discrepancy if for some gene g and two experiments i_1 and i_2, $C_{g_j,i_1} = C_{g_j,i_2}$ for all genes g_j such that an edge $e_{g_j,g}$ exists, but $C_{g,i_1} \neq C_{g,i_2}$. It follows from the network's determinism that at least one of the experiments i_1 or i_2 does not agree with any network trajectory.

Assuming that $P \neq NP$, there does not exist a polynomial time algorithm for resolving all the discrepancies with the minimal number of changes. Therefore, either a heuristic that finds a local optimum or an algorithm that may not terminate in a reasonable amount of time must be used instead. Another difficulty is that a strict subset of the regulation graph may provide a solution with the same number of changes to the expression dataset, and so the structure of the network itself needs to be considered in the search for the optimal solution. This brings another level of complexity to the already difficult problem.

An Algorithm for the Optimal Minimal Network

The in-degree of nodes in the regulation graph is usually assumed to be small compared to the number of genes or the number of experiments. If we assume that it is a constant value in terms of computational complexity, we can define a set of constraints on the values that have to be changed in order to remove all discrepancies from the data. Let $C_{i,j}$ denote the Boolean input value of gene i at experiment j, and let $B_{g_i,j}$ equal 1 if the value of gene g_i in experiment j was flipped in the solution, and otherwise 0. Then for every experiment j and for every gene g_{k+1} with regulators $g_1, g_2, ..., g_k$ and for every Boolean vector $(w_1, w_2, ..., w_k)$, $w_j \in \{0,1\}$, if the output of the Boolean function that determines the value of g_{k+1}, $I(w_1, w_2, ..., w_k)$, is 1, the following constraint must hold:

$$\begin{aligned}\sum_{r=1}^{k}(C_{rj}\cdot(w_r+(1-2\cdot w_r)\cdot B_{g_r,j}) \\ +(1-C_{rj})\cdot((1-w_r)+(2\cdot w_r-1)\cdot B_{g_r,j})) \\ +C_{k+1,j}\cdot B_{g_{k+1},j}+(1-C_{k+1,j})\cdot(1-B_{g_{k+1},j}) \\ <(2-I(w_1,w_2,...,w_k))\cdot(k+1)\end{aligned} \tag{2.1}$$

This constraint means that if the output variable $I(w_1, w_2, ..., w_k)$ was set to 1, whenever the inputs $w_1, w_2, ..., w_k$ appear in the solution the output (the value of g_{k+1}) must be 1. Similarly, if $I(w_1, w_2, ..., w_k)$ is set to 0 the following constraint must hold:

$$\begin{aligned}\sum_{r=1}^{k}(C_{rj}\cdot(w_r+(1-2\cdot w_r)\cdot B_{g_r,j}) \\ +(1-C_{rj})\cdot((1-w_r)+(2\cdot w_r-1)\cdot B_{g_r,j})) \\ +C_{k+1,j}\cdot(1-B_{g_{k+1},j})+(1-C_{k+1,j})\cdot B_{g_{k+1},j} \\ <(I(w_1,w_2,...,w_k)+1)\cdot(k+1)\end{aligned} \tag{2.2}$$

By requiring that under these constraints the following sum is minimized:

$$\sum{}^{i\in 1,...,N}_{j\in 1,...,M} B_{ij}$$

we can use a branch and bound algorithm for 0/1 integer programming to find a solution that fits the data with a minimal number of changes and construct a new dataset with values $D_{ij} = ((C_{ij} + B_{ij}) \bmod 2), i \in 1..N, j \in 1..M$.

However, this formulation still ignores the possible existence of multiple optimal solutions that correspond to different network structures, for if after resolution of the discrepancies only a strict subset of inputs uniquely determines a function's output then the edges in the regulation graph that correspond to the rest of the inputs can be removed. In order to choose the solution that results in the smallest network, for every gene g_i and each one of its targets g_j, we create another Boolean variable R_{ij}, that is added to the right hand side of additional constraints created for g_j. These constraints are similar to (1) and (2) but with all subsets of regulators removed and with their R_{ij} variables in the r.h.s. If the R_{ij} are equal to 1 then given a solution that satisfies the the full constraints, the new constraints will be satisfied as well. If

the new constraints can be satisfied without setting R_{ij} to 1, then the edge from g_i to g_j in the regulation graph is redundant. Therefore, all the variables R_{ij} have weight $\frac{1}{|E|+1}$ in the objective function, where $|E|$ is the number of edges in the regulation graph. The number of variables in the resulting 0/1 integer linear programming is $M \cdot N + \sum_{i=1}^{N} 2^{indegree(g_i)} + |E|$. Once the corrected values D_{ij} are obtained from the optimal solution, we remove redundant regulators to obtain the inferred network structure.

Comparing Groups of Inferred Network States

Given the optimal network and its inferred states, we now wish to compute the probability of seeing the observed difference between two groups of states, referred to as cases and controls, by chance. In order to do that, we compute the difference between the expected between-group state distance and the expected within-group state distance, where the two groups are cases and controls. A distance between two binary states is simply the sum of non-identical entries.

$$\frac{\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M}\sum_{k=1}^{N}|D_{ki}-D_{kj}|}{M\cdot(M-1)} - \tag{2.3}$$

$$\frac{\sum_{i\in group1}\sum_{j\in group1,j\neq i}\sum_{k=1}^{N}|D_{ki}-D_{kj}|}{|group1|\cdot|group1-1|\cdot 2} -$$

$$\frac{\sum_{i\in group2}\sum_{j\in group2,j\neq i}\sum_{k=1}^{N}|D_{ki}-D_{kj}|}{|group2|\cdot|group2-1|\cdot 2}$$

In order to generate a value from the null distribution we sample a number of random states equal to the number of states in the data and apply the network logic to find steady states. We then compute the same statistic (3) by randomly assigning the patient and healthy labels from the data to these states.

A Heuristic for Single Cell Datasets

We propose the following heuristic for scRNA-Seq data:
1. Cluster the input states into K clusters.
2. Solve the problem exactly for the set of cluster centers, where fractional values in the centers are rounded to the closer bit, to obtain a solution H^*. Set the set of regulators E to the one derived from H^*.
3. Initialize the full solution H to the empty set. For input state $i \in 1..M$, if it conflicts with a state in H^* with respect to E, change it incrementally to match the entries of the state in H^* that is closest to it, until all discrepancies are resolved, and add it to the full solution H. Otherwise, if it does not conflict add it to H and H^* without change.
4. Return H

Since H^* is optimal, it is free from discrepancies. H is built incrementally such that after every addition of a state it is free from discrepancies, and therefore it is also free of discrepancies. Similarly, step 3 preserves the set of regulators inferred in step 2, and therefore it does not increase the number of regulators. This heuristic is designed with the nature of single-cell data in mind, where clusters of cells will have a similar network state, and thus it is likely that each distinct state that is present in the optimal solution will already be included in H^*. Note that the order by which changes are applied to the conflicting state in step 3 may vary. For example, one may wish to order the changes according to the number of discrepancies that they resolve after the last change, or to choose the order based on the

network structure. Similar considerations can be applied to the order by which conflicting states are selected for addition.

Results

Analysis of the Gene Regulatory Network for Midbrain Dopaminergic Neurons

In order to test our algorithm we use the mouse embryo scRNA dataset of LaManno et al. [10] and the midbrain dopaminergic (mDA) neuron developmental GRN that was described by [2]. To obtain the gene counts we used the scRNA R package [13]. We used the R package Seurat [14] to inspect the data and to screen cells that had less than 500 or more than 4,000 features. After filtering, the dataset contained 1,631 cells (experiments). We applied SAVER [6] to impute expression values for the network genes. To obtain Boolean values, we map values smaller or equal to the median expression value to Boolean 0, and all other counts to 1. The number of clusters K used in the heuristic was set to 50. The clustering algorithm that we used was k-means as implemented in R, with all other parameters at their default values. The optimal solution flips 5,997 Boolean values, which corresponds to a noise level of approximately 17.5% of the input values. The observed number of discrepancies with respect to the original network structure is 8,331, approximately 24.3% of the input values.

Minimal Network Structure and Logic

Figure 1 shows a multidimensional scaling of the samples using the binary distance between them after network fitting. Each circle corresponds to a distinct network state. The circles form several clusters which correspond to states that are close to one another in the state space. Using a null distribution generated using 1,000 sampled network steady-states, we obtain a p-value of less than $10^{-}3$ when comparing every pair of adjacent clusters using the hypothesis test described in Methods. We propose that each cluster corresponds to a distinct network function, and cells in the same cluster thus share a phenotype induced by the network. Using the inferred logic, perturbations to drive the network to each of these states could be derived, and phenotypic differences under these perturbations could be further studied. Examination of the optimized network structure shows that fitting has resulted in a sparser network structure, as illustrated in Figure 2. After fitting, only 43 of the original 51 edges remained in the network, indicating that some of the interactions do not take place under the conditions induced in the experiment. Figure 3 shows the original network, where edges that were removed after fitting are highlighted in gold. In order to characterize the complexity of the regulatory logic, we examined for all genes the correlation between the number of expressed regulators and the expression of the target. The number of expressed regulators was far from perfectly correlated with the regulatory logic's output when the genes had multiple regulators affecting its value, reflected by low absolute values for the Kendall's tau statistic, with a median of 0.13 and a mean of 0.24. Pitx3 had the highest correlation value (0.57) among genes that had more than 2 regulators. These results suggest that the network's logic is not a monotone function of the number of regulators.

Noise Distribution Across Network Nodes

The noise level of different genes in the model is of interest since assumptions about this parameter are often made in models of gene expression. Therefore, we examined for each gene the number of input values of each type (Boolean 0 and 1) that were flipped in the optimal fit. The distribution of noise across the genes was non-uniform (Fig. 4, Kolmogorov-Smirnov p-value $1.2 \cdot 10^{-8}$) and was not

associated with an expressed state (Boolean 1) or non-expressed state(Boolean 0). This suggests that modeling assumptions that assign an equal level of noise for all genes may lead to wrong conclusions. Further research into the nature of the differences in noise levels between the different genes in this type of data could provide the basis for further improvement of modeling methods.

Conclusion

We propose a new algorithm for fitting a Boolean network model to gene expression data that improves on previous approaches by minimizing the size of the network that fits the data optimally. We further present a heurisitic that allows the analysis of large datasets, which is imperative for the analysis of large datasets such as single-cell RNA sequencing data. Using a known network structure and a dataset of scRNA-Seq measurements, we demonstrated our algorithm by inferring the network structure and its state in different cells. Inspection of the structural properties of the inferred network show that fitting prunes redundant regulators, which stresses the importance of sparsity constraints in the search algorithm. Furthermore, the inferred logic was diverse and showed that constraints on the logic functions should be avoided. By sampling random states and applying the inferred network logic we found that different groups of cells are likely to arise from distinct regulatory trajectories. The regulatory relationships between transcription factors and their targets as given by combining the model and expression data can enabled the selection of targets for perturbation experiments to validate phenotypes induced by the network.

References

1. Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Springer Series in Statistics*, pages 199–213. Springer New York, 1998.
2. Ernest Arenas, Mark Denham, and J. Carlos Villaescusa. How to make a midbrain dopaminergic neuron. *Development*, 142(11):1918–1936, June 2015.
3. L. Geistlinger, G. Csaba, S. Dirmeier, R. Kuffner, and R. Zimmer. A comprehensive gene regulatory network for the diauxic shift in saccharomyces cerevisiae. *Nucleic Acids Research*, 41(18):8452–8463, July 2013.
4. LLC Gurobi Optimization. Gurobi optimizer reference manual, 2021.
5. R. F. Hashimoto, S. Kim, I. Shmulevich, W. Zhang, M. L. Bittner, and E. R. Dougherty. Growing genetic regulatory networks from seed genes. *Bioinformatics*, 20(8):1241–1247, February 2004.
6. Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John I Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. Saver: gene expression recovery for single-cell rna sequencing. *Nature Methods*, 15(7):539–542, 2018.
7. Guy Karlebach and Ron Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9(10):770–780, September 2008.
8. Guy Karlebach and Ron Shamir. Constructing logical models of gene regulatory networks by integrating transcription factor dna interactions with expression data: An entropy-based approach. *Journal of Computational Biology*, 19(1):30–41, January 2012.
9. S.A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467, March 1969.
10. Gioele La Manno, Daniel Gyllborg, Simone Codeluppi, Kaneyasu Nishimura, Carmen Salto, Amit Zeisel, Lars E. Borm, Simon R.W. Stott, Enrique M. Toledo, J. Carlos Villaescusa, Peter Lönnerberg, Jesper Ryge,

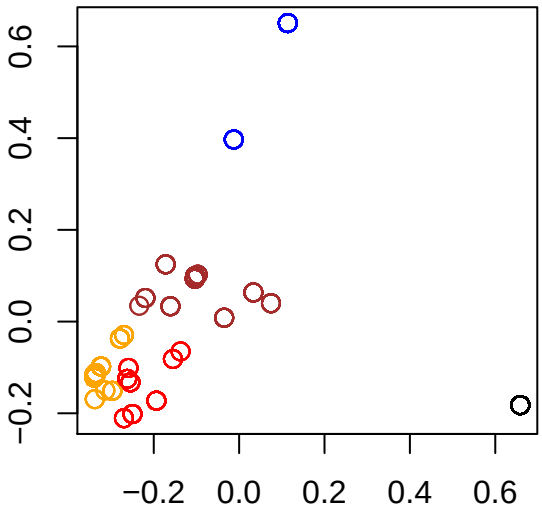

FIG. 1. Multidimensional scaling of network states

Roger A. Barker, Ernest Arenas, and Sten Linnarsson. Molecular diversity of midbrain development in mouse, human, and stem cells. *Cell*, 167(2):566–580.e19, October 2016.

11. Takayuki Ohara, Timothy J. Hearn, Alex A.R. Webb, and Akiko Satake. Gene regulatory network models in response to sugars in the plant circadian system. *Journal of Theoretical Biology*, 457:137–151, November 2018.

12. Jorma Rissanen. A universal prior for integers and estimation by minimum description length. *The Annals of Statistics*, 11(2), June 1983.

13. Davide Risso and Michael Cole. *scRNAseq: Collection of Public Single-Cell RNA-Seq Datasets*, 2020. R package version 2.4.0.

14. Rahul Satija, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology*, 33:495–502, 2015.

15. Gideon Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2), March 1978.

16. Roded Sharan and Richard M. Karp. Reconstructing boolean models of signaling. *Journal of Computational Biology*, 20(3):249–257, March 2013.

17. Yoli Shavit, Boyan Yordanov, Sara-Jane Dunn, Christoph M. Wintersteiger, Tomoki Otani, Youssef Hamadi, Frederick J. Livesey, and Hillel Kugler. Automated synthesis and analysis of switching gene regulatory networks. *Biosystems*, 146:26–34, August 2016.

18. Simon N Willis and Stephen L Nutt. New players in the gene regulatory network controlling late b cell differentiation. *Current Opinion in Immunology*, 58:68–74, June 2019.

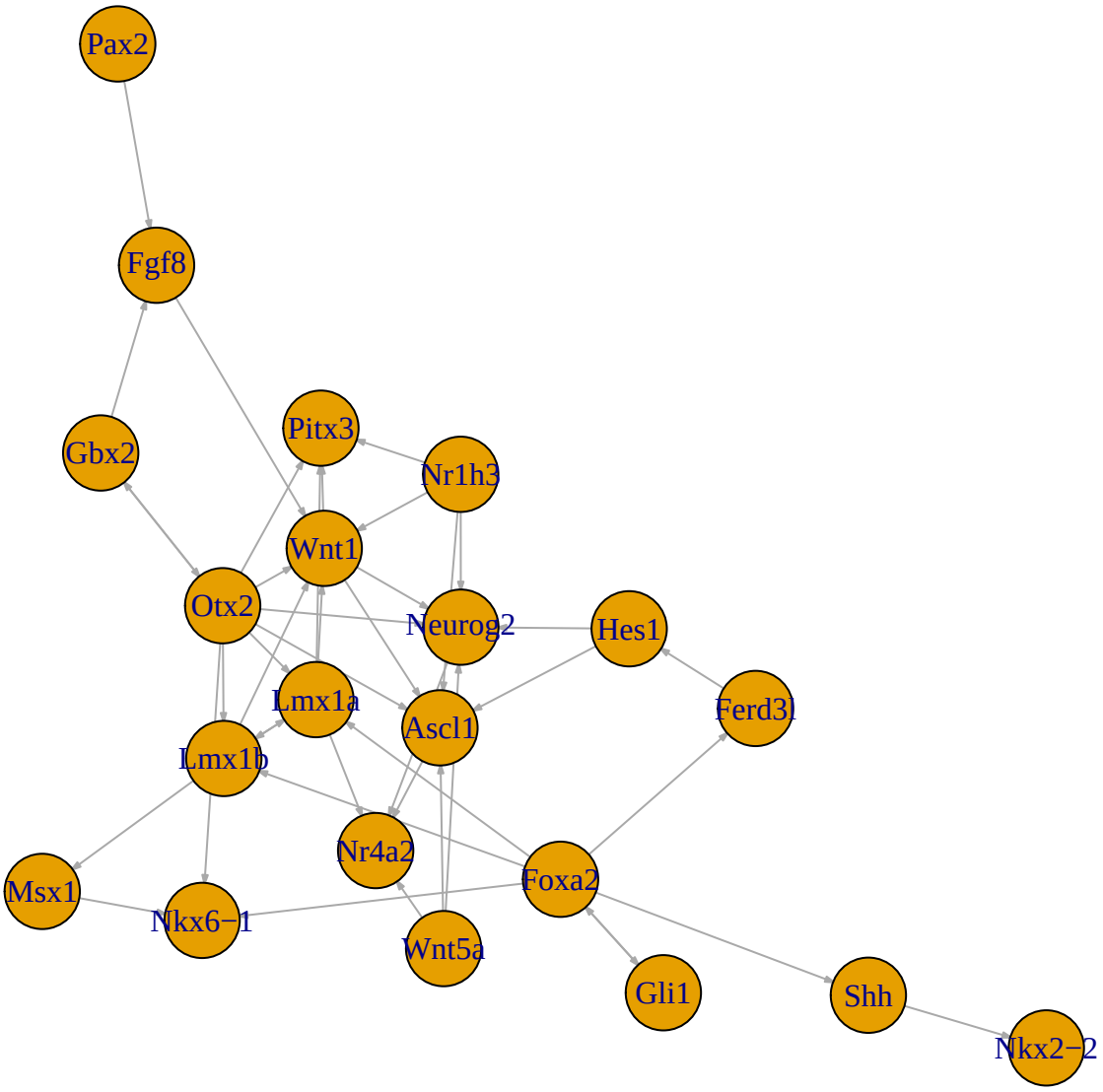

FIG. 2. Optimized network structure

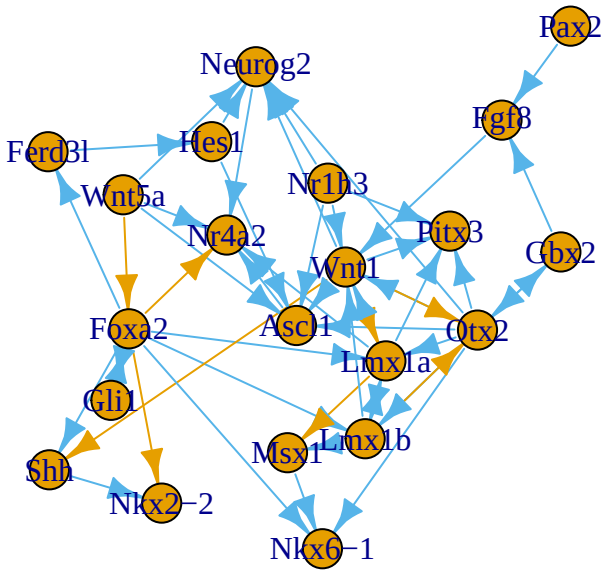

FIG. 3. Network structure before optimization

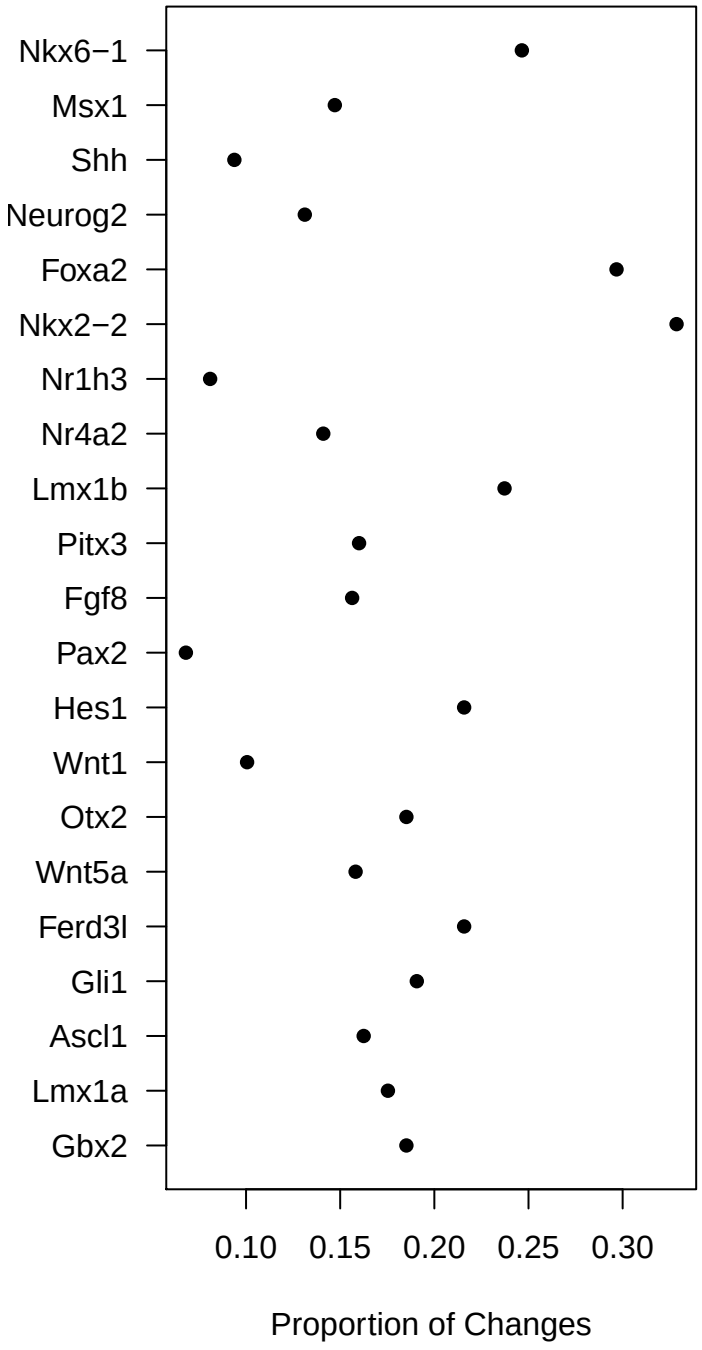

FIG. 4. Noise distribution across genes