Learning and Visualizing Massive Bayesian Networks with FGES-Merge and BayeSuites

Moreno, J. L. Bernaola, N. Larrañaga, P. Bielza, C.

Computational Intelligence Group
Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

joseluis.moreno.rodriguez@upm.es

November 10, 2021

Introduction

- **Bayesian networks** (BNs) for learning the structure of **gene regulatory networks** (GRNs).
- GRNs are complex interregulatory relationships among genes (DNA, RNA, protein and complexes of these).
- **BayeSuites**, a new open-access visualization tool that facilitates the exploration of massive networks.

- **Goal**: GRN learnt with human brain data and using FGES-Merge

What is FGES-Merge?

- **Fast Greedy Equivalence Search Merge** (FGES-Merge), a new method for learning the structure of GRNs via merging locally learned BNs.
- It is based on the **FGES method** (Ramsey, 2017), an optimized version of **GES** (Meek, 1997).

Structure of FGES-Merge

1. For each gene X_i, we select its **most likely neighbours** as candidates for a **local subgraph** around X_i.
2. Each local subgraph is learned using a **modified version** of original FGES method.
3. FGES starts by greedily searching over the **space of edge additions** in the forward equivalent search (FES) step, and then in the backward equivalent search (BES) step the **space of edge deletions**.
4. Finally, we **merge** the local subgraphs by performing graph unions with **pruning** to satisfy the topological properties of GRNs.

Which nodes will belong to each of the smaller networks?

- Instead of calculating the pairwise mutual information (MI) between the nodes (Liu, 2016), we use the **Bayesian information criterion** (BIC) of adding the new edge from X_j to X_i:

$$BIC(X_i|X_j) = N \ln(\hat{\sigma}_e^2) + \lambda k \ln(N)$$

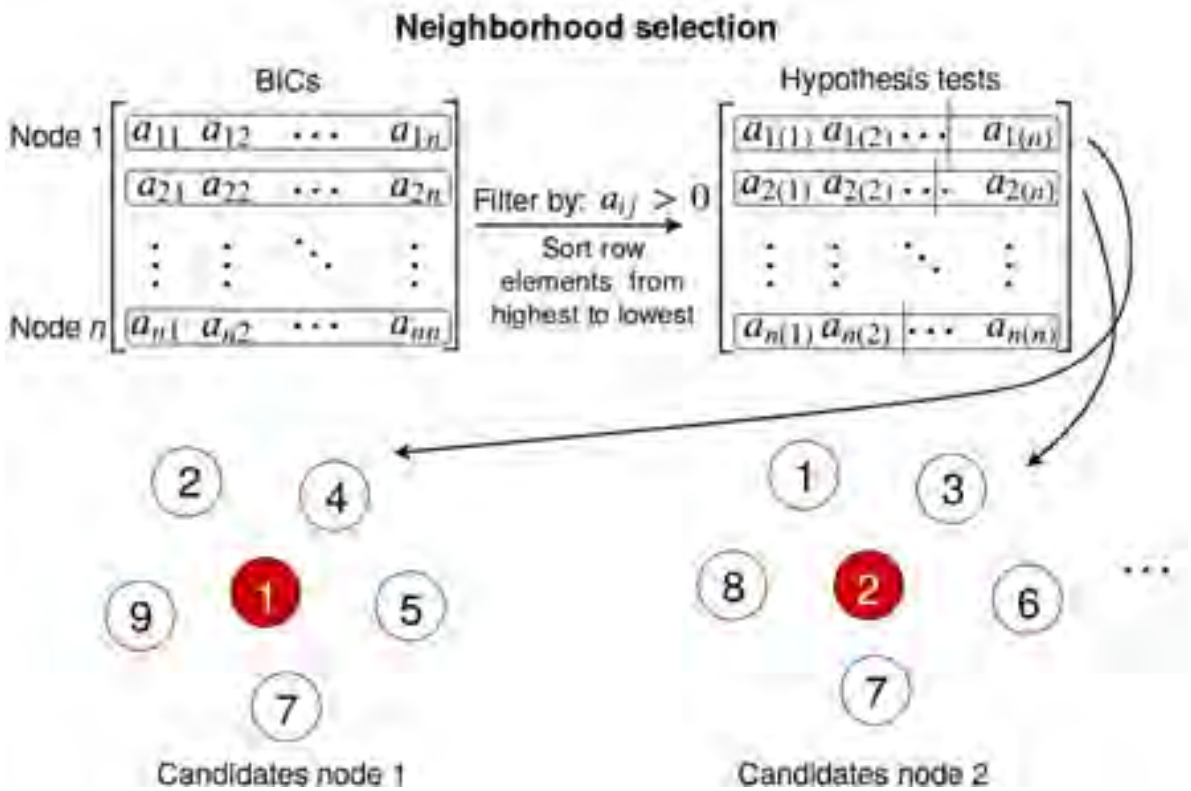
  - $\hat{\sigma}_e^2$: average of the sum of the squares of the residuals of regressing the child node against its parents.
  - N: number of instances.
  - k: number of parents.
  - λ: penalty hyperparameter.

First modification

- Assumption: the level of expression of each gene is distributed as a **linear Gaussian** depending only on its parents (it is valid and gives good results).
- After calculating the BIC values (matrix order N^2), we **parallelized** this step (the calculations are independent of each other).

Second modification

- GRN topological property: number of edges going into each node is a Laplace distribution with an **upper limit**. Therefore, the set of parents of each node has an upper bound, so we decided to limit the possible size of the neighbourhood.
- The set of children can be very large for the hubs (nodes with a higher than average number of edges attached), so we assume that **each child will contain the hub in its neighbourhood**.
- This limitation makes each of the neighbourhood networks smaller and **speeding up** the algorithm.



1. **BIC matrix** is computed, where a_{ij} corresponds to the $\Delta BIC(X_i, X_j)$ of adding the edge.
2. The BICs are filtered: **positive values** and sorted from **highest to lowest** (each row is divided at the most likely point and take the left side as the **neighbours** for the next step).

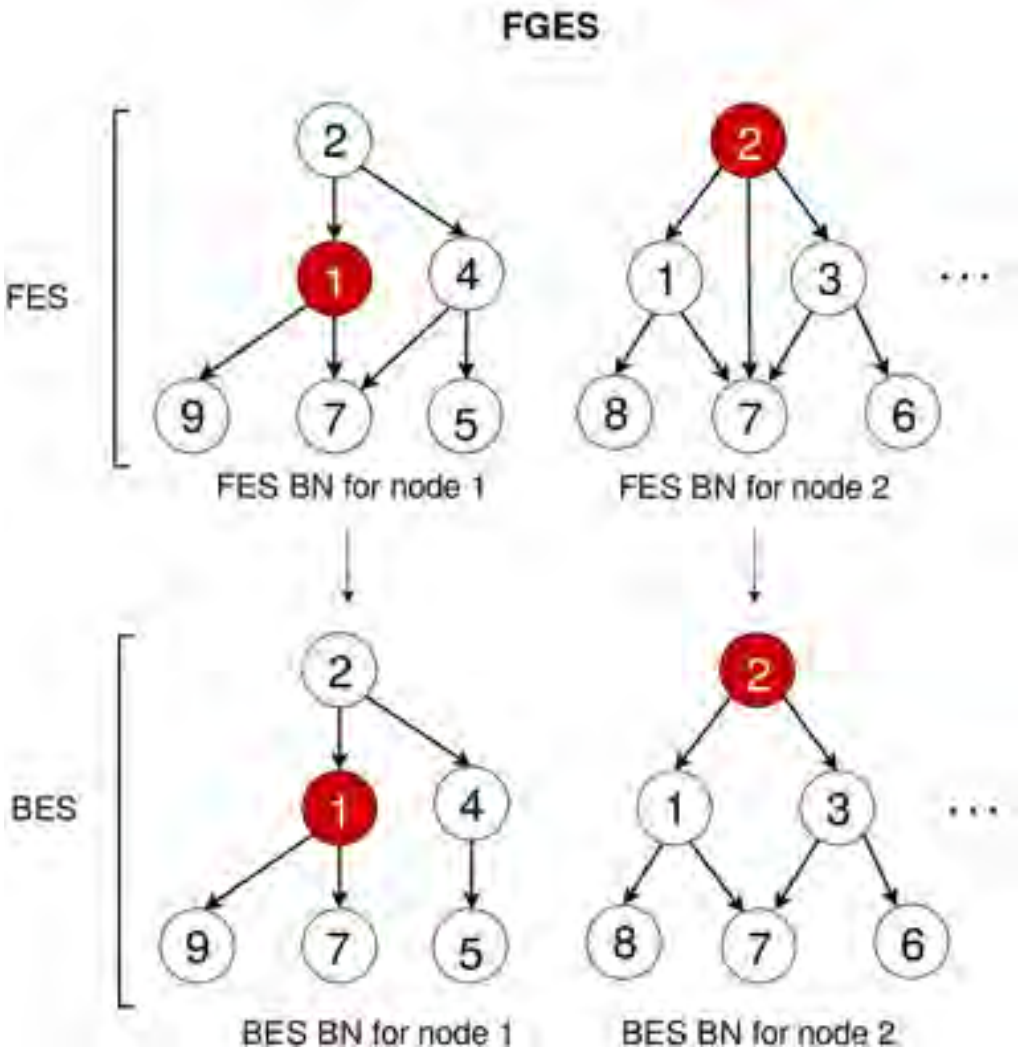
Forward Equivalence Search

- At each FES step, it adds the **best possible edge** to the structure of the BN according to the BIC difference from adding that edge. If we want to add an edge from Z to Y, it is given by:

$$\Delta BIC(Y, Z) = BIC(Y|\mathbf{Pa}(Y) \cup Z) - BIC(Y|\mathbf{Pa}(Y))$$

Backward Equivalence Search

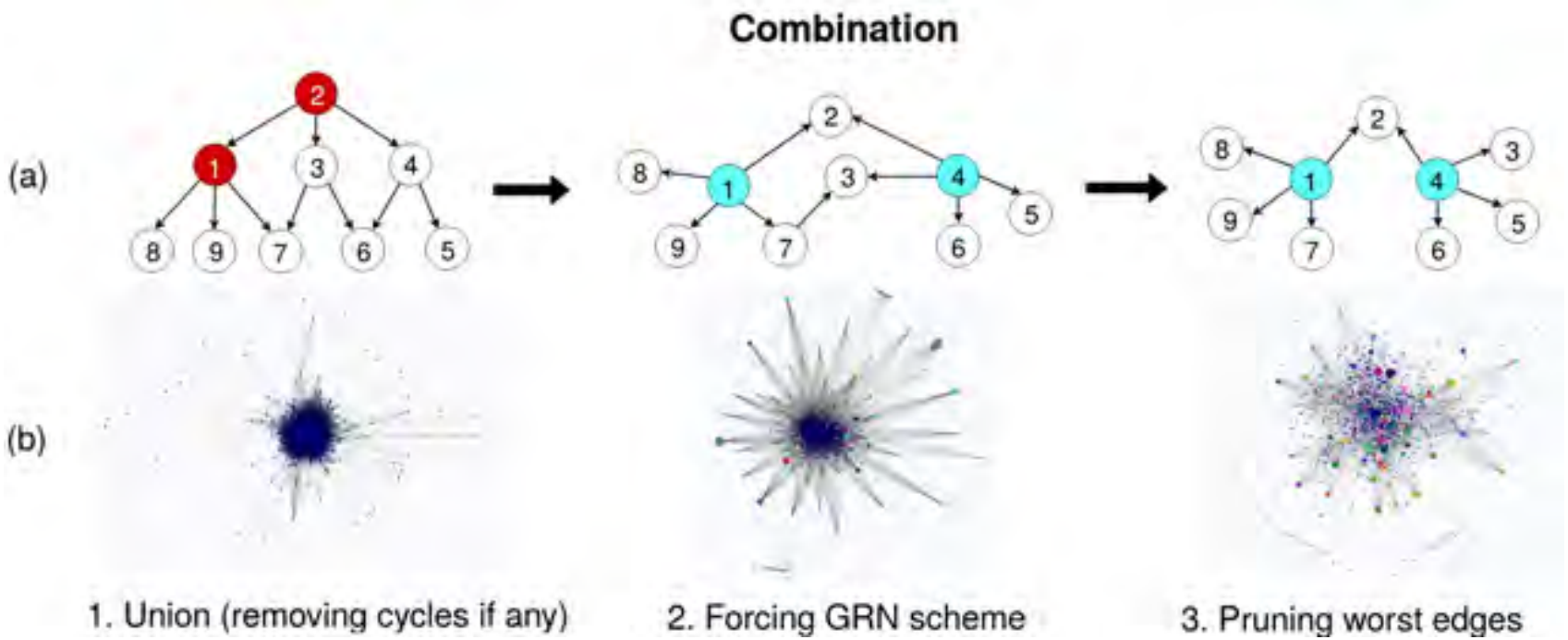
- Since the BIC is a local score, the new edge can only modify the score of some of the edges around it (**skip many computations**).
- Finally, we search the space of edge deletions in the BES step to end up with the **best scoring structure**.



1. We take the **neighbourhood candidates** from the neighbourhood selection and greedily add edges **until the BIC cannot improve** in the FES.
2. We perform any edge deletions that improve the score in the BES.
3. After obtaining the **globally optimal neighbourhood network** for the candidates, we repeat the process for each of the other nodes.

Main modification

- The calculation of the new possible edge additions are **parallelize adding simulated annealing** to choose which edges to add or remove.
  1. The set of all edges that have changed with the last addition is **divided across all processors** (calculating their scores in parallel).
  2. Sometimes our algorithm chooses **suboptimal additions and deletions** (instead of the maximum scoring one). That structure will be **pruned** later



1. **Merging the neighbourhood networks** with their union.
2. **Orienting the edges** from the hubs to their connected nodes.
3. **Prunning the worst performing edges** by their BIC score up to a size threshold.

What is BayeSuites?

- **Problem**: networks with thousands of nodes and edges.
- **BayeSuites** is a web framework for **learning, visualizing, and interpreting** massive Bayesian networks.

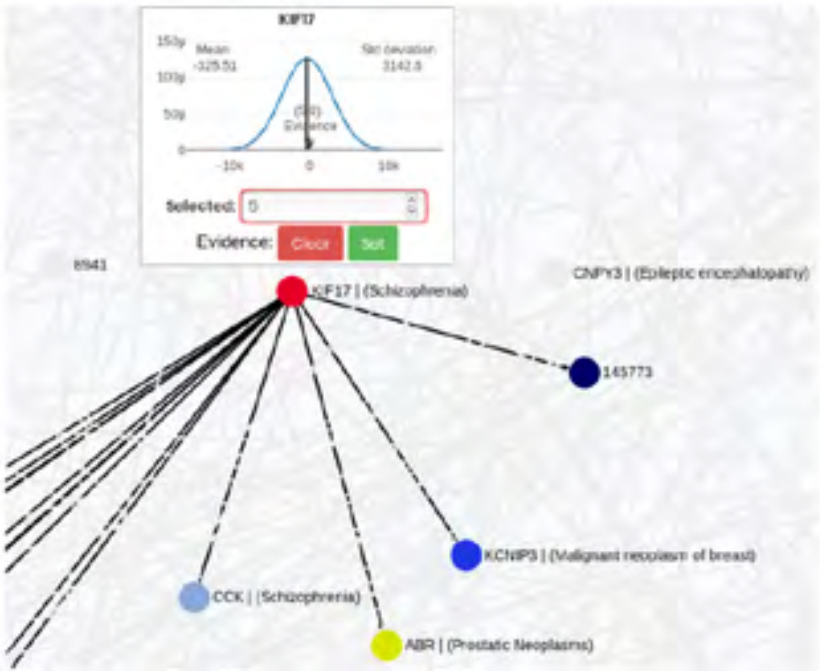
NeuroSuites

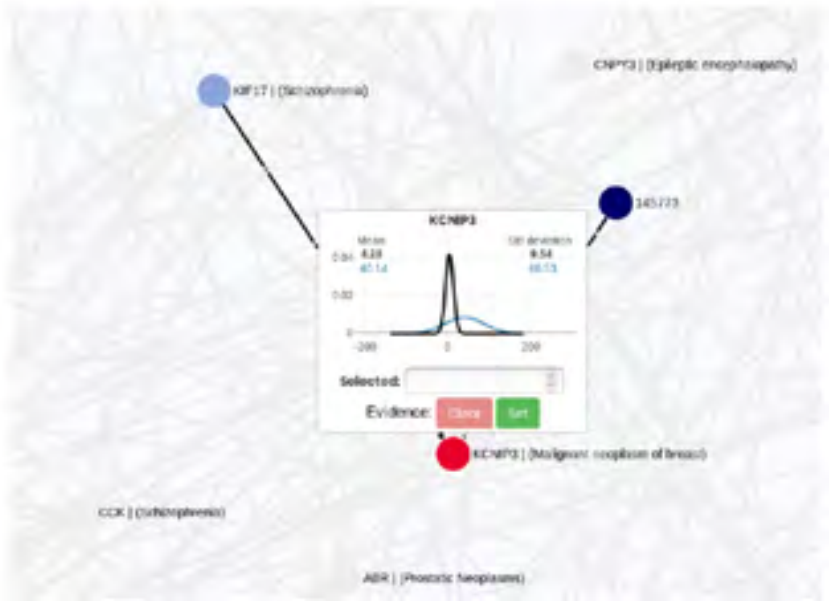
- BayeSuites allows the user to focus on **nodes of interest** and graphically perform the usual operations on BNs.
- It is embedded in the **NeuroSuites** platform (large-scale problems are common in some neuroscience topics).
  `https://neurosuites.com/morpho/ml_bayesian_networks`

- Fruchterman-Reingold layout or ForceAtlas2 layout (at the bottom left) are recommended for GRNs.

Layout options

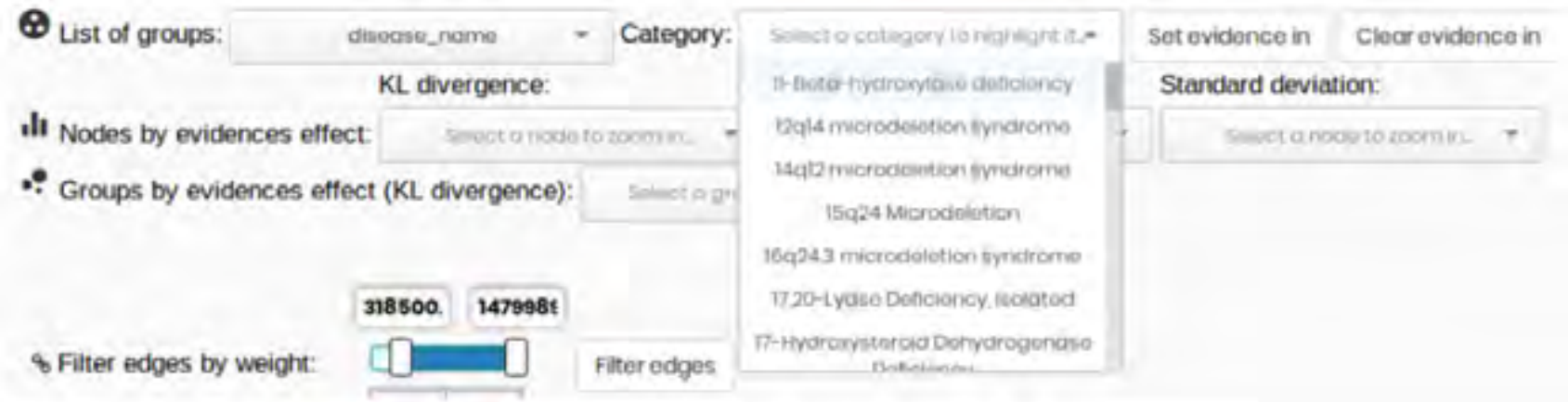
Inference





1. Set a specific value for a node or a group of nodes (evidence variables $\mathbf{E}$).
2. Visualize how the distributions of query nodes Q have changed (with respect to the original ones) and obtain the $p(Q|\mathbf{E})$.

Options


List of groups:
disease_name
Category:
Set evidence in
Clear evidence in
KL divergence:
Standard deviation:
Nodes by evidences effect:
Groups by evidences effect (KL divergence):
11-Beta-hydroxylase deficiency
12q14 microdeletion syndrome
14q32 microdeletion syndrome
15q24 Microdeletion
16q24.3 microdeletion syndrome
17,20-Lyase Deficiency, Isolated
17-Hydroxysteroid Dehydrogenase
318500.
1479981
% Filter edges by weight:
Filter edges

Conclusion

- We have reviewed the problem of **reconstructing GRNs** from gene expression data, restricting the expression levels to be **linear Gaussian distributions** to be able to learn a BN.
- **FGES-Merge** is **fast** (paralellization of parts of the algorithm) and it gets good results respecting the **topological properties** of real GRNs.
- **NeuroSuites** incorporates the learning algorithm FGES-Merge, the inference algorithms and the visualization tool **BayeSuites**.

Source code

- Our version can be found at `https://gitlab.com/mmichiels/fges_parallel_production`

Acknowledgments

- Spanish Ministry of Science and Innovation through the PID2019-109247GB-I00 project.
- The BBVA Foundation (2019 Call) through the "Score-based nonstationary temporal Bayesian networks. Applications in climate and neuroscience" project.
- The BBVA Foundation's grants (2020 Call) for Scientific Investigation Teams SARS-CoV-2 and COVID-19through the "Outcome prediction and treatment efficiency inpatients hospitalized with Covid-19 in Madrid: A Bayesian network approach" project.

Learning and Visualizing Massive Bayesian Networks with FGES-Merge and BayeSuites

Moreno, J. L. Bernaola, N. Larrañaga, P. Bielza, C.

Computational Intelligence Group
Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

joseluis.moreno.rodriguez@upm.es

November 10, 2021