(JC)2BIM 2018 Research School

Biological Network Inference with Sparse Graphical Models

Julien Chiquet

Fréjus, 4–8 June 2018

https://github.com/jchiquet/JC2BIM18

Outline

Outline

Statistical analysis of Networks

Different questions

Understanding the network topology

- Data = observed network
- Questions: central nodes? cluster structure? small-world property?

Inferring/Reconstructing the network

- Data = repeated signal observed at each node
- Questions: which nodes are connected?

Using the network

- Data = a given network + signal on nodes
- Questions: how the epidemic spreads along the network?

Each to be combined with

covariates, time, heterogeneous data set, missing data, ...

Automatic reconstruction of biological networks (1)

E. coli regulatory network

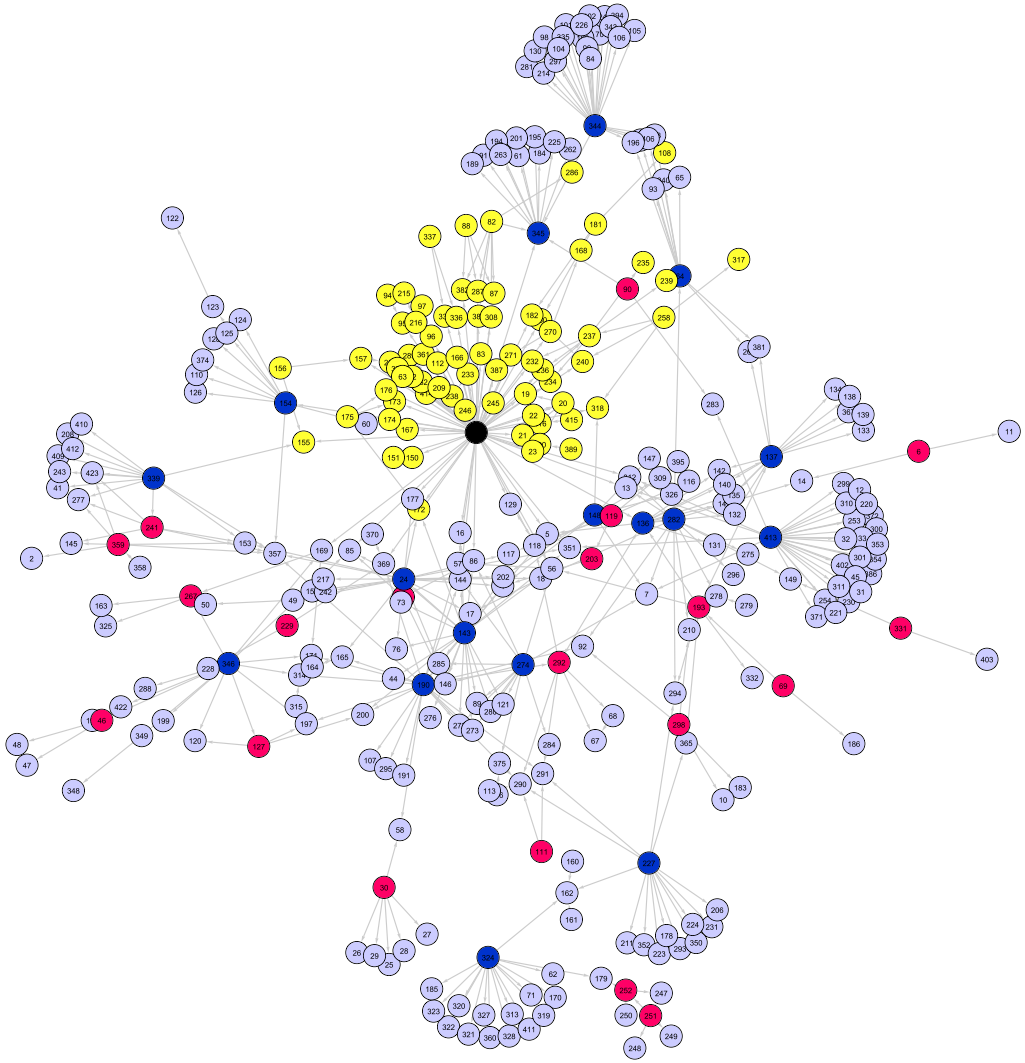
Target network

Relations between genes and their products

- highly structured
- always incomplete

Data and method

- transcriptomic data
- Gaussian graphical model with sparse methods



Automatic reconstruction of biological networks (2)

Microbial association network of the oak tree susceptible to the foliar fungal pathogen

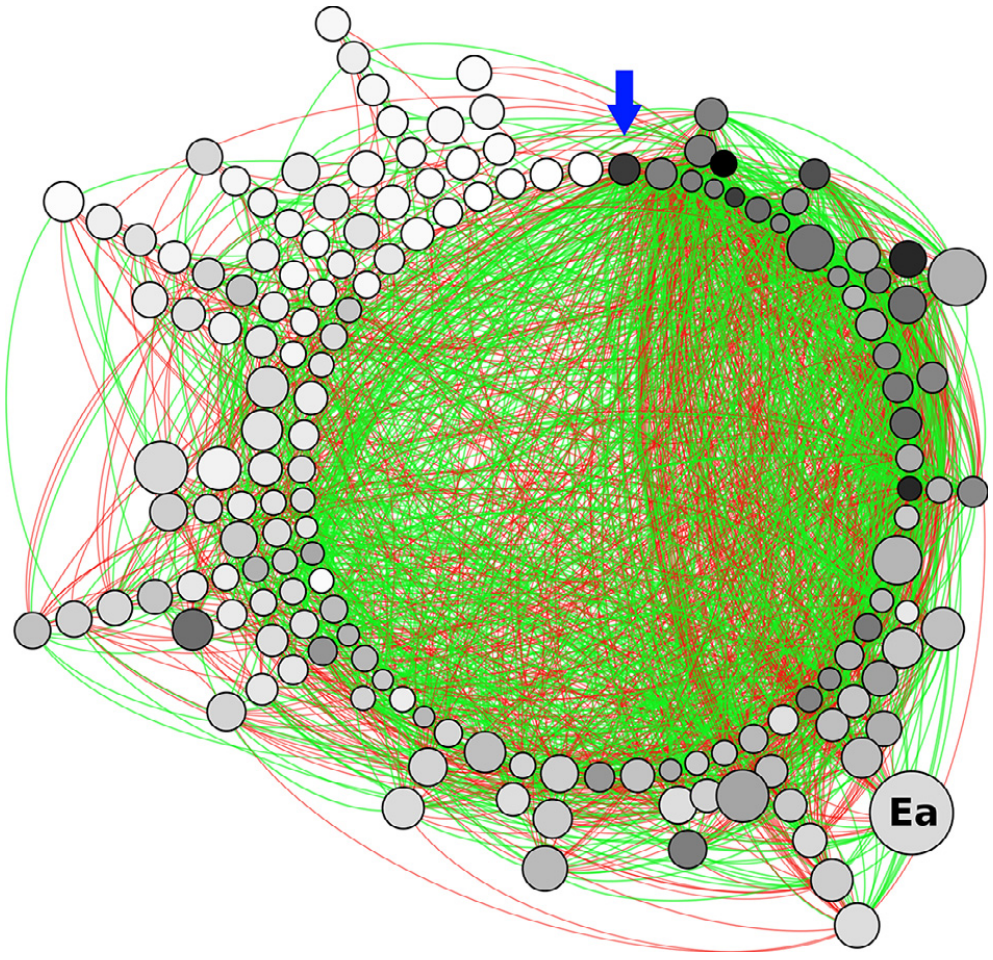
Target network

Relations between microbial species (bacterial or fungal)

- highly structured
- represents co-abudancies

Data and method

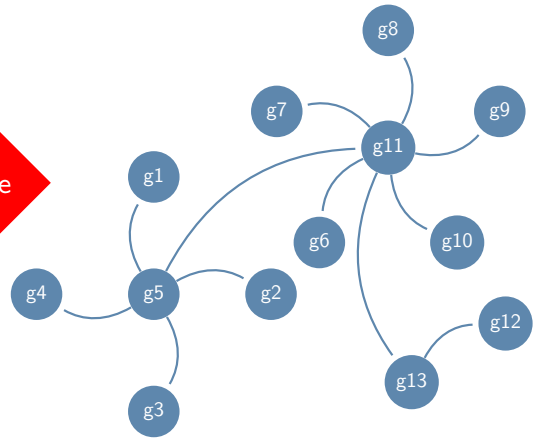
- OTUs table/abundances
- correlation + test/threshold



Vacher et al., Advances in Ecological Research

A challenging problem

Inference



$\approx$ 10s/1,000s assays

$\approx$ 1,000s/1,000,000s features

Model point of view

1. Nodes (genes, OTUS, ...)
   - fixed variables
2. Edges (biological interactions)
   - use (partial) correlations or others fancy statistical concepts
3. Data (intensities, counts)
   - a tidy $n \times p$ dat matrix

$\rightsquigarrow$ Quantities and goals well defined

Data point of view: non classical statistics

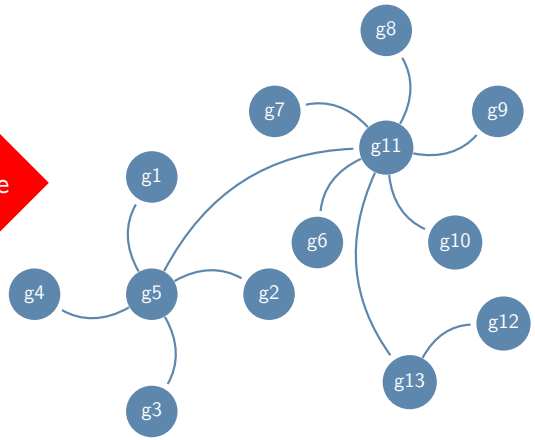
- (Ultra) High dimensionality ($n < p$, $n \ll p$)
- Heterogeneous data

Biological point of view: not well defined goals and questions

- What interaction? Direct? Indirect? Causal?
- Whole network? Subnetwork? Groups of key actors?

A challenging problem

Inference



$\approx$ 10s/1,000s assays

$\approx$ 1,000s/1,000,000s features

Model point of view

1. Nodes (genes, OTUS, ...)
   - fixed variables
2. Edges (biological interactions)
   - use (partial) correlations or others fancy statistical concepts
3. Data (intensities, counts)
   - a tidy $n \times p$ dat matrix

$\rightsquigarrow$ Quantities and goals well defined

Data point of view: non classical statistics

- (Ultra) High dimensionality ($n < p$, $n \lll p$)
- Heterogeneous data

Biological point of view: not well defined goals and questions

- What interaction? Direct? Indirect? Causal?
- Whole network? Subnetwork? Groups of key actors?

Outline

Part I

sparse Gaussian Graphical Models

1. Network and data modeling
2. Network inference with GGM
3. A tour of the `huge` package assessing GGM approach

Outline

Outline

Canonical model settings

Biological microarrays in comparable conditions

Notations

1. a set $\mathcal{P} = \{1, \ldots, p\}$ of p variables:
   these are typically the genes (could be proteins);
2. a sample $\mathcal{N} = \{1, \ldots, n\}$ of individuals associated to the variables:
   these are typically the microarray (could be sequence counts).

Basic statistical model

This can be view as

- a *random vector* X in $\mathbb{R}^p$, whose jth entry is the jth variable,
- a n-size sample $(X^1, \ldots, X^n)$, such as X^i is the ith microarrays,
  - could be independent identically distributed copies (steady-state)
  - could be dependent in a certain way (time-course data)
- assume a parametric probability distribution for X (Gaussian).

Canonical model settings

Biological microarrays in comparable conditions

Notations

1. a set $\mathcal{P} = \{1, \ldots, p\}$ of p variables:
   these are typically the genes (could be proteins);
2. a sample $\mathcal{N} = \{1, \ldots, n\}$ of individuals associated to the variables:
   these are typically the microarray (could be sequence counts).

Basic statistical model

This can be view as

- a *random vector* X in $\mathbb{R}^p$, whose jth entry is the jth variable,
- a n-size sample $(X^1, \ldots, X^n)$, such as X^i is the ith microarrays,
  - could be independent identically distributed copies (steady-state)
  - could be dependent in a certain way (time-course data)
- assume a parametric probability distribution for X (Gaussian).

Canonical model settings

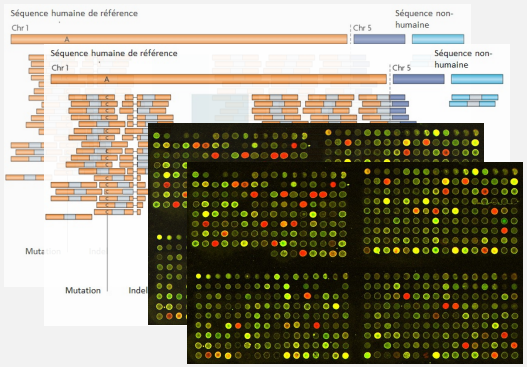
Biological microarrays in comparable conditions

Notations

1. a set $\mathcal{P} = \{1, \ldots, p\}$ of p variables;

The data

Stacking $(X^1, \ldots, X^n)$, we met the usual individual/variable table $\mathbf{X}$



stacked in

$$\mathbf{X} = \begin{pmatrix} x_1^1 & x_1^2 & x_1^3 & \ldots & x_1^p \\ \vdots & & & & \\ x_n^1 & x_n^2 & x_1^2 & \ldots & x_n^p \end{pmatrix}$$

- a n-size sample $(X^1, \ldots, X^n)$, such as X^i is the ith microarrays,
  - could be independent identically distributed copies (steady-state)
  - could be dependent in a certain way (time-course data)
- assume a parametric probability distribution for X (Gaussian).

Outline

sparse Gaussian Graphical Models

Modeling relationship between variables (1)

Independence

Definition (Independence of events)

Two events A and B are independent if and only if

$$\mathbb{P}(A, B) = \mathbb{P}(A)\mathbb{P}(B),$$

which is usually denoted by $A \perp\!\!\!\perp B$. Equivalently,

- $A \perp\!\!\!\perp B \Leftrightarrow \mathbb{P}(A|B) = \mathbb{P}(A)$,
- $A \perp\!\!\!\perp B \Leftrightarrow \mathbb{P}(A|B) = \mathbb{P}(A|B^c)$

Example (class vs party)

	party	
class	Labour	Tory
working	0.42	0.28
bourgeoisie	0.06	0.24

	party	
class	Labour	Tory
working	0.60	0.40
bourgeoisie	0.20	0.80

Table: Joint probability (left) vs. conditional probability (right)

Modeling relationship between variables (1)

Independence

Definition (Independence of events)

Two events A and B are independent if and only if

$$\mathbb{P}(A, B) = \mathbb{P}(A)\mathbb{P}(B),$$

which is usually denoted by $A \perp\!\!\!\perp B$. Equivalently,

- $A \perp\!\!\!\perp B \Leftrightarrow \mathbb{P}(A|B) = \mathbb{P}(A)$,
- $A \perp\!\!\!\perp B \Leftrightarrow \mathbb{P}(A|B) = \mathbb{P}(A|B^c)$

Example (class vs party)

	party	
class	Labour	Tory
working	0.42	0.28
bourgeoisie	0.06	0.24

	party	
class	Labour	Tory
working	0.60	0.40
bourgeoisie	0.20	0.80

Table: Joint probability (left) vs. conditional probability (right)

Modeling relationships between variables (2)

Conditional independence

Generalizing to more than two events requires strong assumptions (mutual independence). Better handle with

Definition (Conditional independence of events)

Two events A and B are conditionally independent if and only if

$$\mathbb{P}(A, B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C),$$

which is usually denoted by $A \perp\!\!\!\perp B|C$

Example (Does QI depends on weight?)

Consider the events $A =$ "having low QI", $B =$ "having low weight".

Modeling relationships between variables (2)

Conditional independence

Generalizing to more than two events requires strong assumptions (mutual independence). Better handle with

Definition (Conditional independence of events)

Two events A and B are conditionally independent if and only if

$$\mathbb{P}(A, B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C),$$

which is usually denoted by $A \perp\!\!\!\perp B|C$

Example (Does QI depends on weight?)

Consider the events $A =$ "having low QI", $B =$ "having low weight".

Modeling relationships between variables (2)

Conditional independence

Generalizing to more than two events requires strong assumptions (mutual independence). Better handle with

Definition (Conditional independence of events)

Two events A and B are conditionally independent if and only if

$$\mathbb{P}(A, B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C),$$

which is usually denoted by $A \perp\!\!\!\perp B|C$

Example (Does QI depends on weight?)

Consider the events $A =$ "having low QI", $B =$ "having low weight". Estimating[1] $\mathbb{P}(A, B)$, $\mathbb{P}(A)$ and $\mathbb{P}(B)$ in a sample would lead to

$$\mathbb{P}(A, B) \neq \mathbb{P}(A)\mathbb{P}(B)$$

[1] stupidly

Modeling relationships between variables (2)

Conditional independence

Generalizing to more than two events requires strong assumptions (mutual independence). Better handle with

Definition (Conditional independence of events)

Two events A and B are conditionally independent if and only if

$$\mathbb{P}(A, B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C),$$

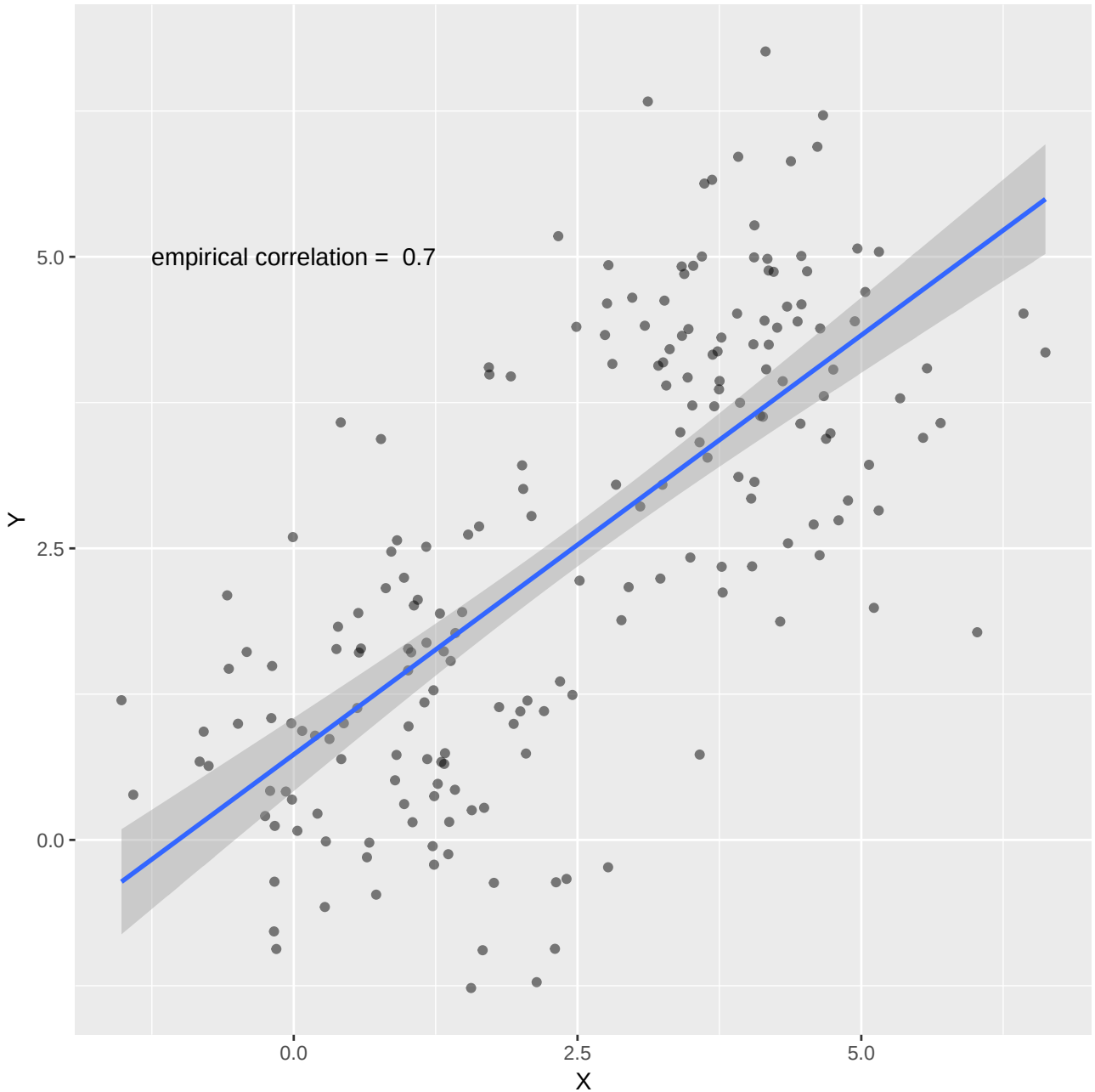
which is usually denoted by $A \perp\!\!\!\perp B|C$

Example (Does QI depends on weight?)

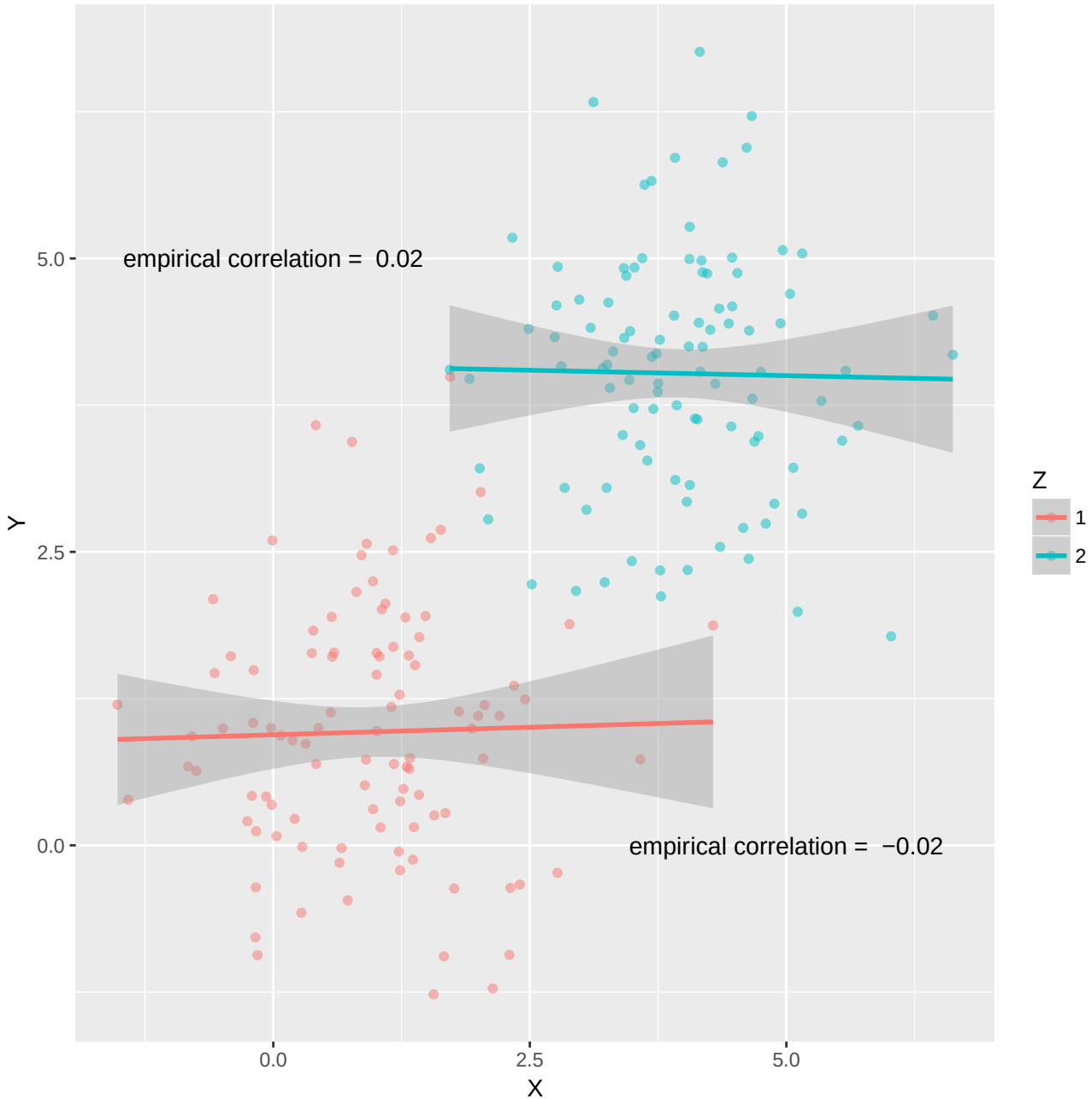
Consider the events $A =$ "having low QI", $B =$ "having low weight". But in fact, introducing $C =$ "having a given age",

$$\mathbb{P}(A, B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C)$$

Limits of correlation for network reconstruction



Limits of correlation for network reconstruction



Outline

sparse Gaussian Graphical Models

Correlation networks

Correlation (association network)

Similar expression profile $\rightsquigarrow$ high-correlation

1. Compute the correlation matrix (Pearson, Spearman, ...)
2. Predict an edge between two actors if their absolute correlation is above a given threshold

Questions

- How to set up the threshold?
- If we target actors with similar profiles, why not clustering?
- Information is drowned (all actors are correlated ...)

Graphical models

Definition

A graphical model gives a graphical (intuitive) representation of the dependence structure of a probability distribution, by linking

1. a random vector (or a set of random variables.) $X = \{X_1, \ldots, X_p\}$ with distribution $\mathbb{P}$,

2. a graph $\mathcal{G} = (\mathcal{P}, \mathcal{E})$ where
   - $\mathcal{P} = \{1, \ldots, p\}$ is the set of nodes associated to each variable,
   - $\mathcal{E}$ is a set of edges describing the dependence relationship of $X \sim \mathbb{P}$.

Conditional independence graph

It is the undirected graph $\mathcal{G} = \{\mathcal{P}, \mathcal{E}\}$ where

$$(i, j) \notin \mathcal{E} \Leftrightarrow X_i \perp\!\!\!\perp X_j | \mathcal{P} \backslash \{i, j\}.$$

Graphical models

Definition

A graphical model gives a graphical (intuitive) representation of the dependence structure of a probability distribution, by linking

1. a random vector (or a set of random variables.) $X = \{X_1, \ldots, X_p\}$ with distribution $\mathbb{P}$,

2. a graph $\mathcal{G} = (\mathcal{P}, \mathcal{E})$ where
   - $\mathcal{P} = \{1, \ldots, p\}$ is the set of nodes associated to each variable,
   - $\mathcal{E}$ is a set of edges describing the dependence relationship of $X \sim \mathbb{P}$.

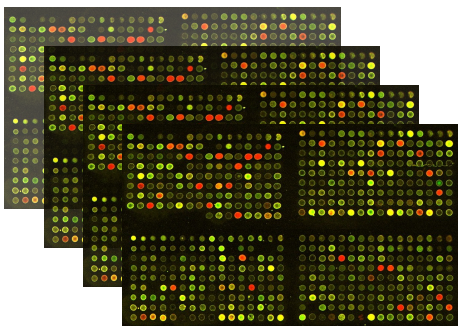
Conditional independence graph

It is the undirected graph $\mathcal{G} = \{\mathcal{P}, \mathcal{E}\}$ where

$$(i, j) \notin \mathcal{E} \Leftrightarrow X_i \perp\!\!\!\perp X_j | \mathcal{P} \backslash \{i, j\}.$$

The Gaussian case

The data



Inference

$$\mathbf{X} = \begin{pmatrix} x_1^1 & x_1^2 & x_1^3 & \dots & x_1^p \\ \vdots & & & & \\ x_n^1 & x_n^2 & x_1^2 & \dots & x_n^p \end{pmatrix}$$

Assuming $f_X(\mathbf{X})$ multivariate Gaussian

Greatly simplifies the inference:

- $\rightsquigarrow$ naturally links independence and conditional independence to the covariance and partial covariance,
- $\rightsquigarrow$ gives a straightforward interpretation to the graphical modeling previously considered.

Why Gaussianity helps?

Case of 2 variables or size-2 random vector

Let X, Y be two real random variables.

Definitions

$$\operatorname{cov}(X,Y) = \mathbb{E}\Big[\big(X - \mathbb{E}(X)\big)\big(Y - \mathbb{E}(Y)\big)\Big] = \mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y).$$

$$\rho_{XY} = \operatorname{cor}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sqrt{\mathbb{V}(X) \cdot \mathbb{V}(Y)}}.$$

Proposition

- $\operatorname{cov}(X,X) = \mathbb{V}(X) = \mathbb{E}[(X - \mathbb{E}X)(Y - \mathbb{E}Y)]$,
- $\operatorname{cov}(X+Y,Z) = \operatorname{cov}(X,Z) + \operatorname{cov}(X,Z)$,
- $\mathbb{V}(X+Y) = \mathbb{V}(X) + \mathbb{V}(Y) + \operatorname{cov}(X,Y)$.
- $X \perp\!\!\!\perp Y \Rightarrow \operatorname{cov}(X,Y) = 0$.
- $X \perp\!\!\!\perp Y \Leftrightarrow \operatorname{cov}(X,Y) = 0$ *when* X, Y *are Gaussian.*

Why Gaussianity helps?

Case of 2 variables or size-2 random vector

Let X, Y be two real random variables.

Definitions

$$\operatorname{cov}(X,Y) = \mathbb{E}\Big[\big(X - \mathbb{E}(X)\big)\big(Y - \mathbb{E}(Y)\big)\Big] = \mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y).$$

$$\rho_{XY} = \operatorname{cor}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sqrt{\mathbb{V}(X) \cdot \mathbb{V}(Y)}}.$$

Proposition

- $\operatorname{cov}(X,X) = \mathbb{V}(X) = \mathbb{E}[(X - \mathbb{E}X)(Y - \mathbb{E}Y)]$,
- $\operatorname{cov}(X+Y,Z) = \operatorname{cov}(X,Z) + \operatorname{cov}(X,Z)$,
- $\mathbb{V}(X+Y) = \mathbb{V}(X) + \mathbb{V}(Y) + \operatorname{cov}(X,Y)$.
- $X \perp\!\!\!\perp Y \Rightarrow \operatorname{cov}(X,Y) = 0$.
- $X \perp\!\!\!\perp Y \Leftrightarrow \operatorname{cov}(X,Y) = 0$ *when* X, Y *are Gaussian*.

The bivariate Gaussian distribution

The Covariance Matrix

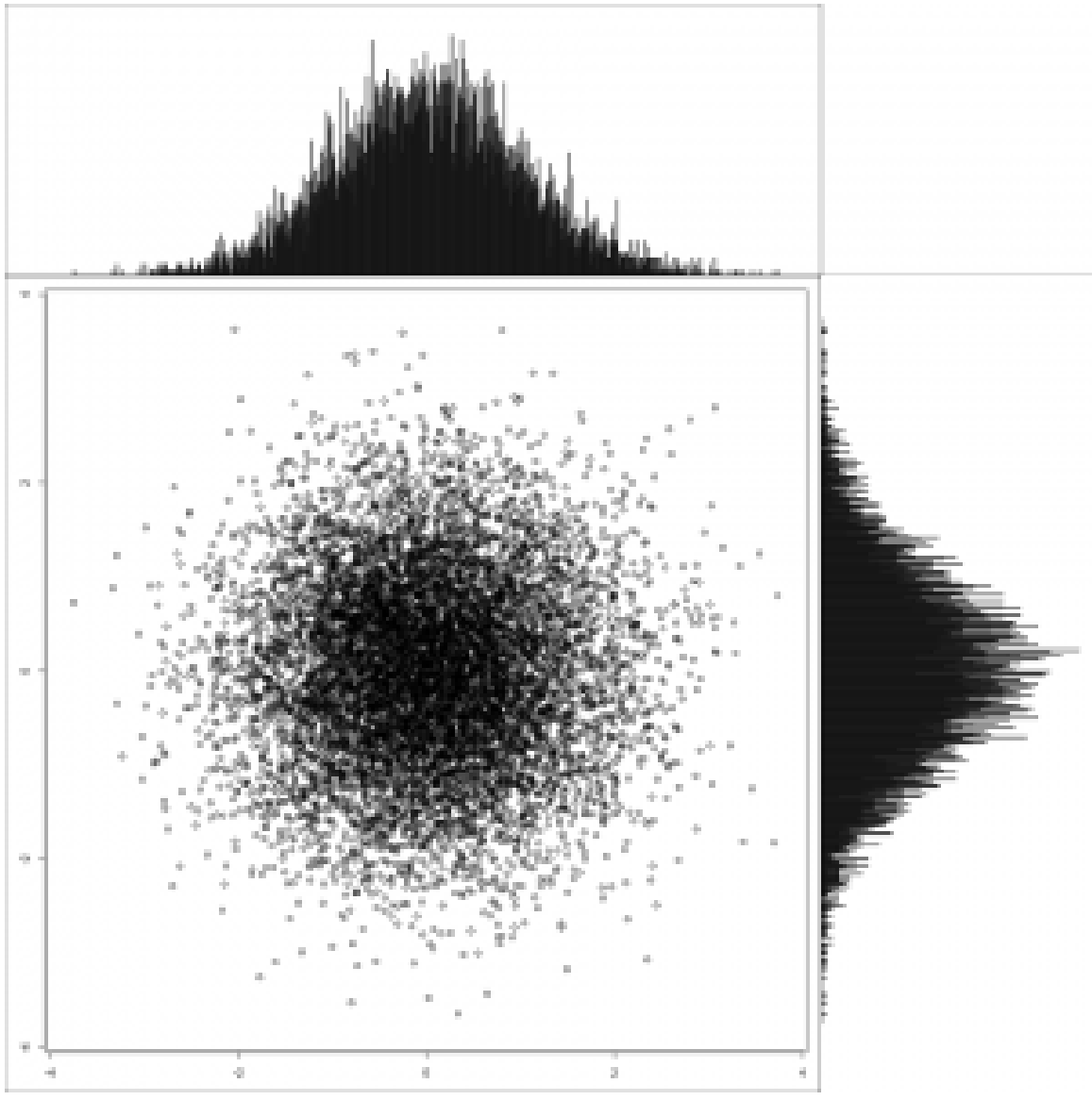
Let

$$X \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}),$$

with unit variance and $\rho_{XY} = 0$

$$\mathbf{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

The shape of the 2-D distribution evolves accordingly.



The bivariate Gaussian distribution

The Covariance Matrix

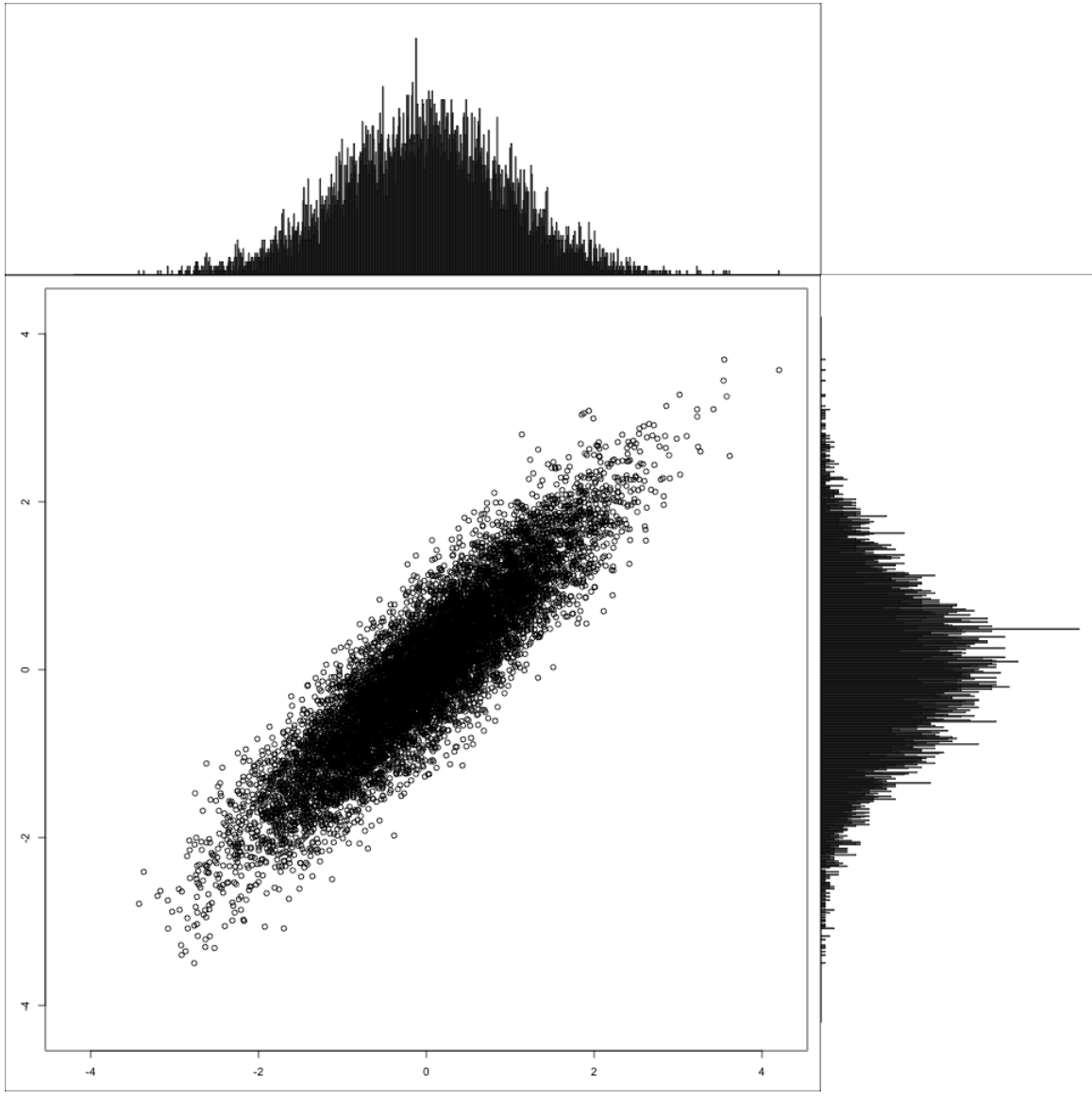
Let

$$X \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}),$$

with unit variance and $\rho_{XY} = 0.9$

$$\mathbf{\Sigma} = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}.$$

The shape of the 2-D distribution evolves accordingly.



Generalization: multivariate Gaussian vector

Now need partial covariance and partial correlation

Let X, Y, Z be real random variables.

Definitions

$$\mathrm{cov}(X, Y|Z) = \mathrm{cov}(X, Y) - \mathrm{cov}(X, Z)\mathrm{cov}(Y, Z)/\mathbb{V}(Z).$$

$$\rho_{XY|Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{1-\rho_{XZ}^2}\sqrt{1-\rho_{YZ}^2}}.$$

$\rightsquigarrow$ Give the interaction between X and Y once removed the effect of Z.

Proposition

When X, Y, Z are jointly Gaussian, then

$$\mathrm{cov}(X, Y|Z) = 0 \Leftrightarrow \mathrm{cor}(X, Y|Z) = 0 \Leftrightarrow X \perp\!\!\!\perp Y|Z.$$

Generalization: multivariate Gaussian vector

Now need partial covariance and partial correlation

Let X, Y, Z be real random variables.

Definitions

$$\mathrm{cov}(X, Y|Z) = \mathrm{cov}(X, Y) - \mathrm{cov}(X, Z)\mathrm{cov}(Y, Z)/\mathbb{V}(Z).$$

$$\rho_{XY|Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{1-\rho_{XZ}^2}\sqrt{1-\rho_{YZ}^2}}.$$

$\rightsquigarrow$ Give the interaction between X and Y once removed the effect of Z.

Proposition

When X, Y, Z are jointly Gaussian, then

$$\mathrm{cov}(X, Y|Z) = 0 \Leftrightarrow \mathrm{cor}(X, Y|Z) = 0 \Leftrightarrow X \perp\!\!\!\perp Y|Z.$$

Important properties of Gaussian vectors

Proposition (Gaussian vector and conditioning)

Consider a Gaussian vector with the following decomposition

$$Z = \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}), \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}, \quad \boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \boldsymbol{\Omega}_{11} & \boldsymbol{\Omega}_{12} \\ \boldsymbol{\Omega}_{21} & \boldsymbol{\Omega}_{22} \end{pmatrix}.$$

Then,

$$Z_2 | Z_1 = \mathbf{z} \sim \mathcal{N}\left(-\boldsymbol{\Omega}_{22}^{-1}\boldsymbol{\Omega}_{21}\mathbf{z}, \boldsymbol{\Omega}_{22}^{-1}\right)$$

and

$$\boldsymbol{\Omega}_{22}^{-1} = \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}.$$

Corollary

Partial correlations are related to the inverse of the covariance matrix:

$$\operatorname{cor}(Z_i, Z_j | Z_k, k \neq i, j) = -\frac{\Omega_{ij}}{\sqrt{\Omega_{ii}\Omega_{jj}}}$$

Gaussian Graphical Model: canonical settings

Biological experiments in comparable Gaussian conditions

Profiles of a set $\mathcal{P} = \{1, \ldots, p\}$ of genes is described by $X \in \mathbb{R}^p$ such as

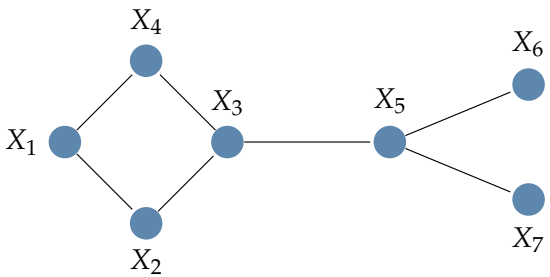
1. $X \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, with $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ the precision matrix.
2. a sample $(X^1, \ldots, X^n)$ of exp. stacked in an $n \times p$ data matrix $\mathbf{X}$.

Conditional independence structure

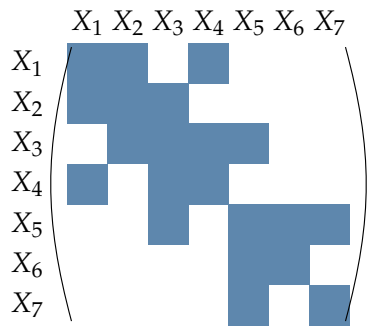
$$(i,j) \notin \mathcal{E} \Leftrightarrow X_i \perp\!\!\!\perp X_j | X_{\setminus\{i,j\}} \Leftrightarrow \Theta_{ij} = 0.$$

Graphical interpretation

$$\mathcal{G} = (\mathcal{P}, \mathcal{E})$$



$\boldsymbol{\Theta}$



$\rightsquigarrow$ "Covariance" selection

Gaussian Graphical Model and Linear Regression

Linear regression viewpoint

Gene expression X_i is linearly explained by the other genes':

$$X_i | X_{\setminus i} = -\sum_{j \neq i} \frac{\Theta_{ij}}{\Theta_{ii}} X_j + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \Omega_{ii}^{-1}), \quad \varepsilon_i \perp X$$

Conditional on its neighborhood, other profiles do not give additional insights

$$X_i | X_{\setminus i} = \sum_{j \in \text{neighbors}(i)} \beta_j X_j + \varepsilon_i \quad \text{with } \beta_j = -\frac{\Theta_{ij}}{\Theta_{ii}}.$$

$\rightsquigarrow$ "Neighborhood" selection

Gaussian Graphical Model and AR process (1)

Time course data

Time course- data experiment can be represented as a multivariate vector $X = (X_1, \ldots, X_p) \in \mathbb{R}^p$, generated through a first order vector autoregressive process $VAR(1)$:

$$X^t = \mathbf{\Theta} X^{t-1} + \mathbf{b} + \boldsymbol{\varepsilon}^t, \quad t \in [1, n]$$

where $\boldsymbol{\varepsilon}^t$ is a white noise to ensure the Markov property and $X^0 \sim \mathcal{N}(0, \mathbf{\Sigma}^0)$.

Consequence: a Gaussian Graphical Model

- Each $X^t | X^{t-1} \sim \mathcal{N}(\theta X^{t-1}, \mathbf{\Sigma})$,
- or, equivalently, $X_j^t | X^{t-1} \sim \mathcal{N}(\mathbf{\Theta}_j X^{t-1}, \mathbf{\Sigma})$

where $\mathbf{\Sigma}$ is known and $\mathbf{\Theta}_j$ is the jth row of $\mathbf{\Theta}$.

Gaussian Graphical Model and AR process (1)

Time course data

Time course- data experiment can be represented as a multivariate vector $X = (X_1, \ldots, X_p) \in \mathbb{R}^p$, generated through a first order vector autoregressive process $VAR(1)$:

$$X^t = \mathbf{\Theta} X^{t-1} + \mathbf{b} + \boldsymbol{\varepsilon}^t, \quad t \in [1, n]$$

where $\boldsymbol{\varepsilon}^t$ is a white noise to ensure the Markov property and $X^0 \sim \mathcal{N}(0, \mathbf{\Sigma}^0)$.

Consequence: a Gaussian Graphical Model

- Each $X^t | X^{t-1} \sim \mathcal{N}(\theta X^{t-1}, \mathbf{\Sigma})$,
- or, equivalently, $X_j^t | X^{t-1} \sim \mathcal{N}(\mathbf{\Theta}_j X^{t-1}, \mathbf{\Sigma})$

where $\mathbf{\Sigma}$ is known and $\mathbf{\Theta}_j$ is the jth row of $\mathbf{\Theta}$.

Gaussian Graphical Model and AR process (2)

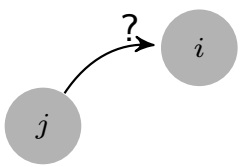
Interpretation as a GGM

The VAR(1) as a covariance selection model

$$\theta_{ij} = \frac{\text{cov}\left(X_i^t, X_j^{t-1} | X_{\mathcal{P}\setminus j}^{t-1}\right)}{\text{var}\left(X_j^{t-1} | X_{\mathcal{P}\setminus j}^{t-1}\right)},$$

Graphical Interpretation

$\rightsquigarrow$ The matrix $\mathbf{\Theta} = (\theta_{ij})_{i,j\in\mathcal{P}}$ encodes the network $\mathcal{G}$ we are looking for.



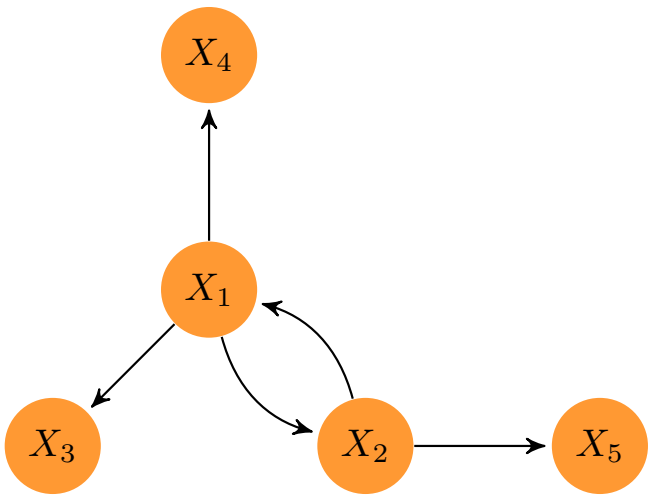
if and only if

conditional dependency between X_j^{t-1} and X_i^t
or
non-null partial correlation between X_j^{t-1} and X_i^t
$\Updownarrow$
$\theta_{ij} \neq 0$

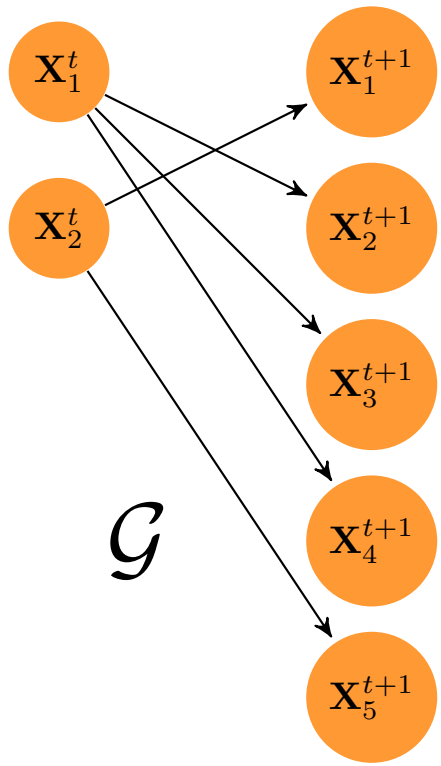
Gaussian Graphical Model and AR process (3)

Graphical interpretation

1. Follow-up of one single experiment/individual;
2. Close enough time-points to ensure
   - dependency between consecutive measurements;
   - homogeneity of the Markov process.



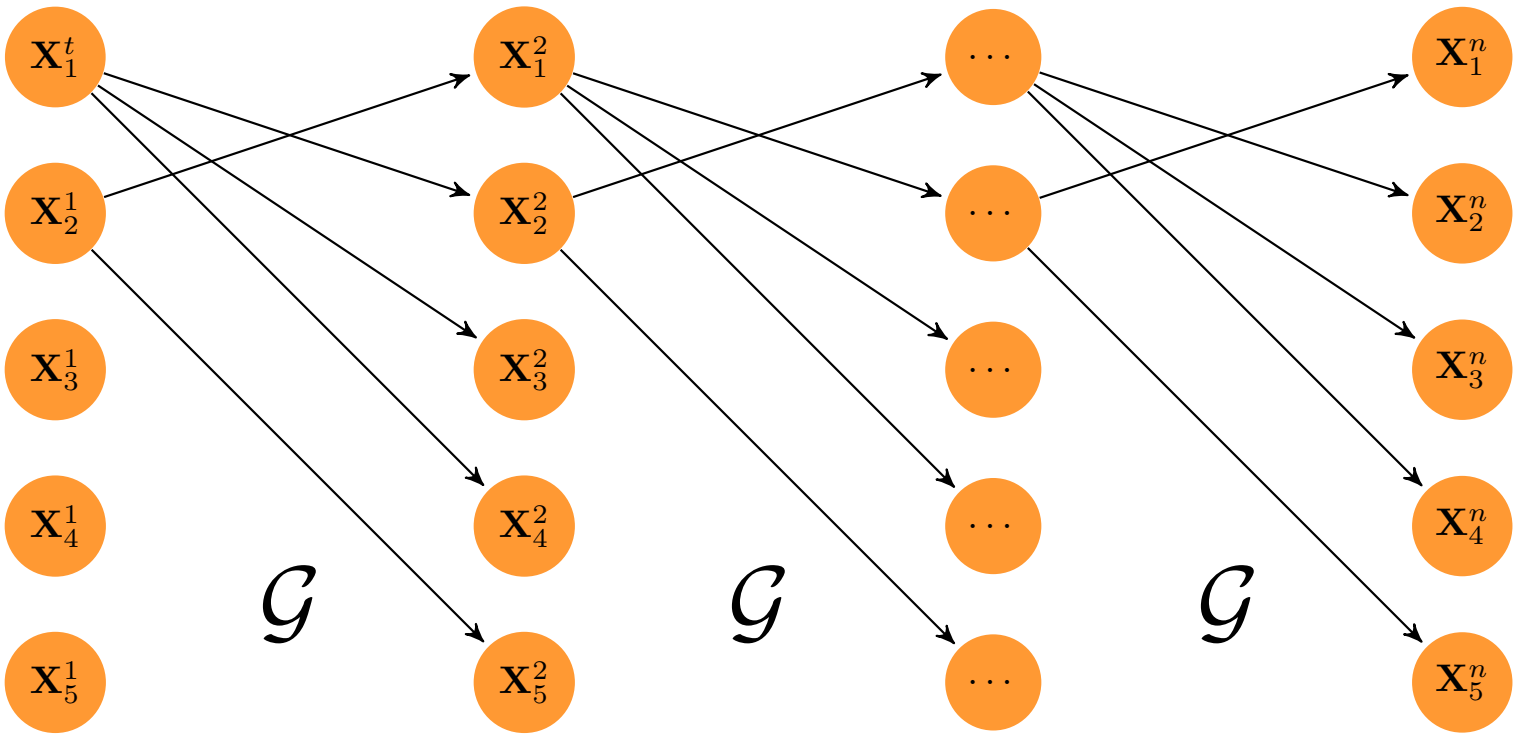
stands for



Gaussian Graphical Model and AR process (3)

Graphical interpretation

1. Follow-up of one single experiment/individual;
2. Close enough time-points to ensure
   - dependency between consecutive measurements;
   - homogeneity of the Markov process.



Outline

Some families of methods for network reconstruction

Test-based methods

- Tests the nullity of each entries
- Combinatorial problem when $p > 30$ ...

Sparsity-inducing regularization methods

- induce sparsity with the ℓ_1-norm penalization
- Use results from convex optimization
- Versatile and computationally efficient

Bayesian methods

- Compute the posterior probability of each edge
- Usually more computationally demanding
- For special graphs, computation gets easier

Outline

sparse Gaussian Graphical Models

Inference: maximum likelihood estimator

The natural approach for parametric statistics

Let X be a random vector with distribution defined by $f_X(x; \mathbf{\Theta})$, where $\mathbf{\Theta}$ are the model parameters.

Maximum likelihood estimator

$$\hat{\mathbf{\Theta}} = \arg\max_{\mathbf{\Theta}} \ell(\mathbf{\Theta}; \mathbf{X})$$

where ℓ is the log likelihood, a function of the parameters:

$$\ell(\mathbf{\Theta}; \mathbf{X}) = \log \prod_{i=1}^{n} f_X(\mathbf{x}_i; \mathbf{\Theta}),$$

where $\mathbf{x}_i$ is the ith row of $\mathbf{X}$.

Remarks

- This a convex optimization problem,
- We just need to detect non zero coefficients in $\mathbf{\Theta}$

The multivariate Gaussian log-likelihood

Let $\mathbf{S} = n^{-1}\mathbf{X}^\intercal\mathbf{X}$ be the empirical variance-covariance matrix: $\mathbf{S}$ is a sufficient statistic of $\mathbf{\Theta}$.

The log-likelihood

$$\ell(\mathbf{\Theta}; \mathbf{S}) = \frac{n}{2}\log\det(\mathbf{\Theta}) - \frac{n}{2}\mathrm{Trace}(\mathbf{S}\mathbf{\Theta}) + \frac{n}{2}\log(2\pi).$$

- $\rightsquigarrow$ The MLE $= \mathbf{S}^{-1}$ of $\mathbf{\Theta}$ is not defined for $n < p$ and never sparse.
- $\rightsquigarrow$ The need for regularization is huge.

Application to GGM: the "Graphical-Lasso"

A penalized likelihood approach

$$\hat{\mathbf{\Theta}}_\lambda = \arg\max_{\mathbf{\Theta} \in \mathbb{S}_+} \ell(\mathbf{\Theta}; \mathbf{X}) - \lambda \|\mathbf{\Theta}\|_{\ell_1}$$

where

- ℓ is the model log-likelihood,
- $\|\cdot\|_{\ell_1}$ is a penalty function tuned by $\lambda > 0$.
  1. *regularization* (needed when $n \ll p$),
  2. *selection* (sparsity induced by the ℓ_1-norm),
- solved in R-packages **glasso**, **quic**, **huge** ($\mathcal{O}(p^3)$)

Application to GGM: "Neighborhood selection"

A close cousin, thank to the relationship between Gaussian vector and linear regression

Remember that

$$X_i | X_{\setminus i} = \sum_{j \in \text{neighbors}(i)} \beta_j X_j + \varepsilon_i \quad \text{with } \beta_j = -\frac{\Theta_{ij}}{\Theta_{ii}}.$$

A penalized least-square approach

Let $\mathbf{X}_i$ be the ith column of the data matrix (i.e data associated to variable (gene) i), and $\mathbf{X}_{\setminus i}$ deprived of colmun i. We select the neighbors of variable i by solving

$$\widehat{\boldsymbol{\beta}}^{(i)} = \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^{p-1}} \frac{1}{n} \left\| \mathbf{X}_i - \mathbf{X}_{\setminus i} \boldsymbol{\beta} \right\|_2^2 + \lambda \left\| \boldsymbol{\beta} \right\|_1$$

- not symmetric, not positive-definite
+ p Lasso solved with Lars-like algorithms ($\mathcal{O}(npd)$ for d neighbors).

Outline

sparse Gaussian Graphical Models

Practical implications of theoretical results

Selection consistency (Ravikumar, Wainwright, 2009-2012)

Denote $d = \max_{j \in \mathcal{P}}(\text{degree}_\text{j})$. Consistency for an appropriate λ and

- $n \approx \mathcal{O}(d^2 \log(p))$ for the graphical Lasso and Clime.
- $n \approx \mathcal{O}(d \log(p))$ for neighborhood selection (sharp).

(Irrepresentability) conditions are not strictly comparable...

Ultra high-dimension phenomenon (Verzelen, 2011)

Minimax risk for sparse regression with d-sparse models: useless when

$$\frac{d \log(p/d)}{n} \geq 1/2, \qquad (\text{e.g.}, n = 50, p = 200, d \geq 8).$$

Good news! when n is small, we don't need to solve huge problems because they can't but fail.

Practical implications of theoretical results

Selection consistency (Ravikumar, Wainwright, 2009-2012)

Denote $d = \max_{j \in \mathcal{P}}(\text{degree}_\text{j})$. Consistency for an appropriate λ and

- $n \approx \mathcal{O}(d^2 \log(p))$ for the graphical Lasso and Clime.
- $n \approx \mathcal{O}(d \log(p))$ for neighborhood selection (sharp).

(Irrepresentability) conditions are not strictly comparable...

Ultra high-dimension phenomenon (Verzelen, 2011)

Minimax risk for sparse regression with d-sparse models: useless when

$$\frac{d \log(p/d)}{n} \geq 1/2, \qquad (\text{e.g.}, n = 50, p = 200, d \geq 8).$$

Good news! when n is small, we don't need to solve huge problems because they can't but fail.

Model selection

Cross-validation

Optimal in terms of prediction, not in terms of selection

Information based criteria

- GGMSelect (Girault *et al*, '12) selects among a family of candidates.
- Adapt IC to sparse high dimensional problems, e.g.

$$\text{EBIC}_\gamma(\widehat{\mathbf{\Theta}}_\lambda) = -2\text{loglik}(\widehat{\mathbf{\Theta}}_\lambda; \mathbf{X}) + |\mathcal{E}_\lambda|(\log(n) + 4\gamma \log(p)),$$

Resampling/subsampling

Keep edges frequently selected on an range of λ after sub-samplings

- Stability Selection (Meinshausen and Bühlman, 2010, Bach 2008)
- Stability approach to Regularization Selection (StaRS) (Liu, 2010).

Concluding remark about GGM

Sparse GGM

+ very solid statistical and computational framework
+ competitive to other inference methods (DREAM 5 benchmark, 2012)
− performances remain questionable on real data, as for other methods

$\rightsquigarrow$ Network inference is a very difficult problem
$\rightsquigarrow$ Some biological questions can be answered without network inference

Outline

Assess the standard GGMs approaches

Full analysis can be found at `http://julien.cremeriefamily.info/doc/teachings/exposome/td_exposome_correction.html`

```
suppressMessages(library(huge, quietly = TRUE))
```

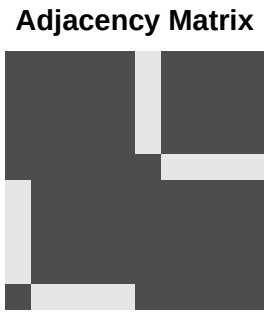
1. Simulated data
   - Test that an approach is working under some simple conditions
   - Especially usefull when the approach has no underlying model
   - Essential sanity check
2. Breast cancer data (pinpoint interesting genes/pathways)
   - Several hundred breast cancers (estrogen receptor + and -)
   - Several thousand genes
   - Goal: How can GGMs approaches help ?

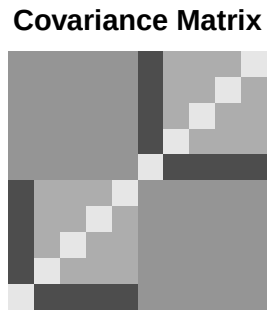
Simple simulations (network with hubs)

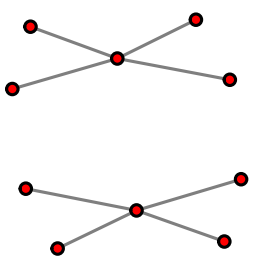
```
set.seed(11)
n <- 80; d <- 10;
rd.net  <- huge.generator(
  n, ## number of samples
  d, ## number of genes
  graph="hub", ## type of net
  g = 2, ## number of group)
  verbose=FALSE)
```

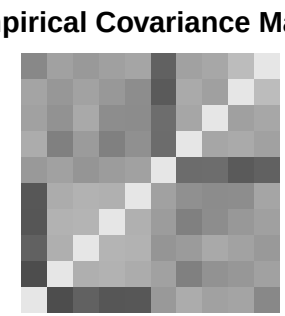
Simple simulations (network with hubs)

```
plot(rd.net)
```









Inference using GGMs and correlation

Inference

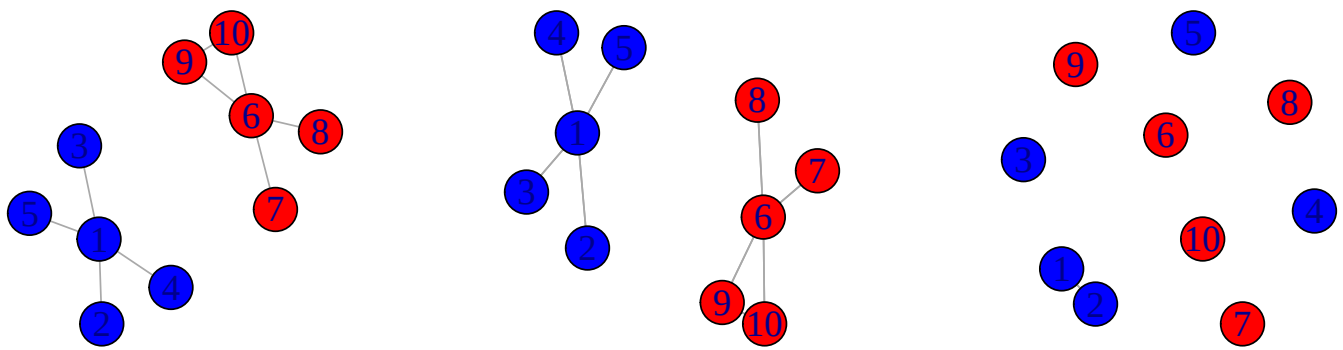
```
## glasso, mb and ct
glasso <- huge(rd.net$data, method="glasso",
               nlambda=50,  verbose=F)
mb <- huge(rd.net$data, method="mb",
           nlambda=50, verbose=F)
corthr <- huge(rd.net$data, method="ct",
              nlambda = 50, verbose=F)
```

Selection

```
## glasso, mb and ct
glasso.sel <- huge.select(glasso, "stars", verbose=F)
mb.sel <- huge.select(mb, "stars", verbose=F)
corthr.sel <- huge.select(corthr, "stars", verbose=F)
```

Inference using GGMs and correlation (results)

```
gr.glasso <- graph.adjacency(glasso.sel$refit)
V(gr.glasso)$label.cex <- 2
V(gr.glasso)$color <- rep(c("blue", "red"), each=5)
par(mfrow=c(1, 3))
plot(gr.glasso, vertex.size=30, edge.arrow.mode = "-")
plot(gr.mb, vertex.size=30, edge.arrow.mode = "-")
plot(gr.cor, vertex.size=30, edge.arrow.mode = "-")
```



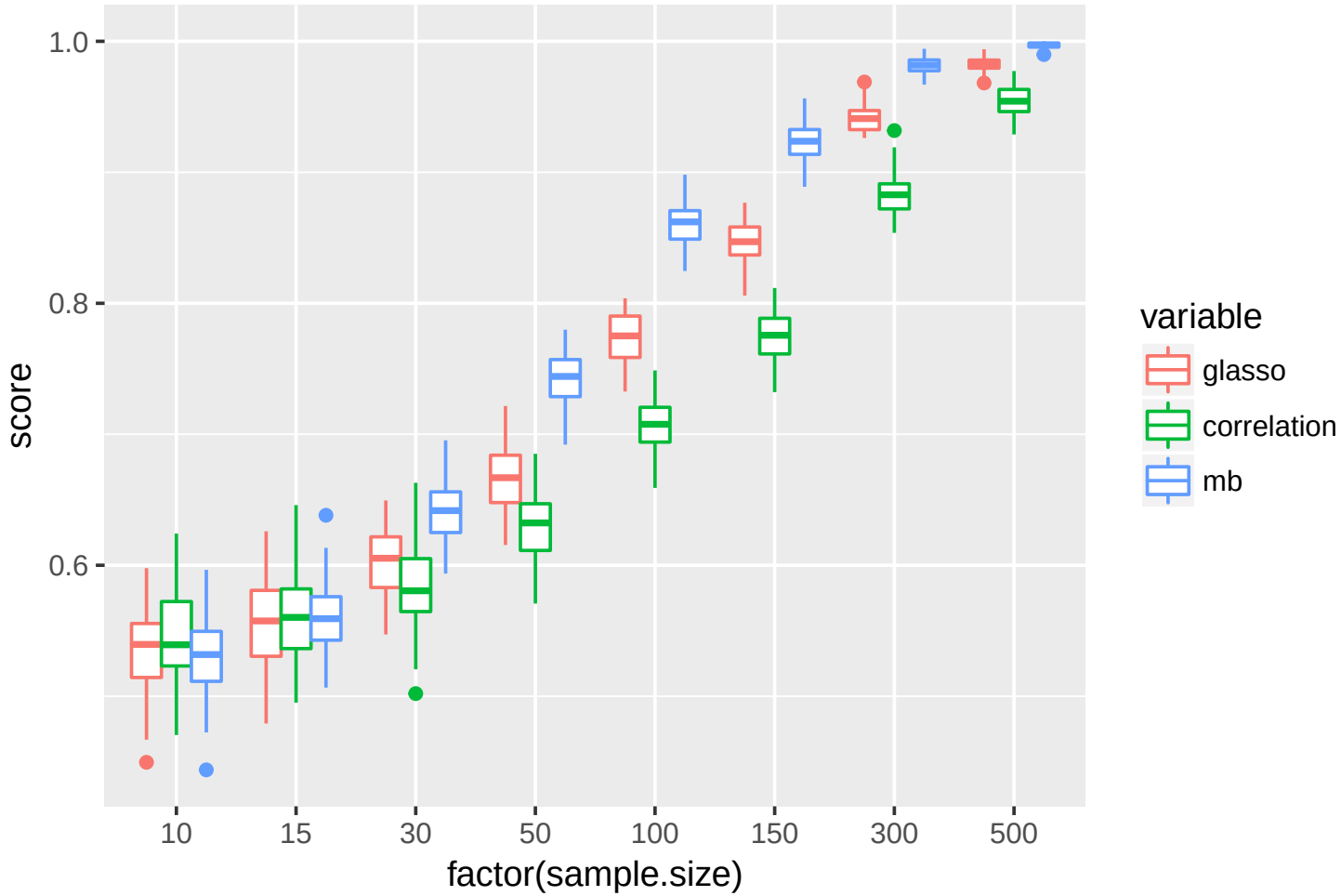
A bit of code to run a simulation

```
suppressMessages(require(reshape2))
one.simu <- function(i) {
  lbd.c <- seq(1, 0, -10^-2);
  d <- 25; seq.n <- c(10, 15, 30, 50, 100, 150, 300, 500)
  out <- data.frame(t(sapply(seq.n, function(n) {
   exp <- huge.generator(n, d, graph="cluster",
                         g=3, prob=1, verbose=F)
   gl <- huge(exp$data, method="glasso", nlambda=50, verbose=F)
   mb <- huge(exp$data, method="mb", nlambda=50, verbose=F)
   cthr <- huge(exp$data, method="ct", lambda=lbd.c, verbose=F)
   res.cthr <- perf.auc(perf.roc(cthr$path, exp$theta))
   res.gl <- perf.auc(perf.roc(gl$path, exp$theta))
   res.mb <- perf.auc(perf.roc(mb$path, exp$theta))
   return(setNames(c(res.gl,res.cthr,res.mb,n,i),
   c("glasso","correlation","mb", "sample size", "simu")))
  })))
return(melt(out, measure.vars = 1:3, value.name = "score"))}
```

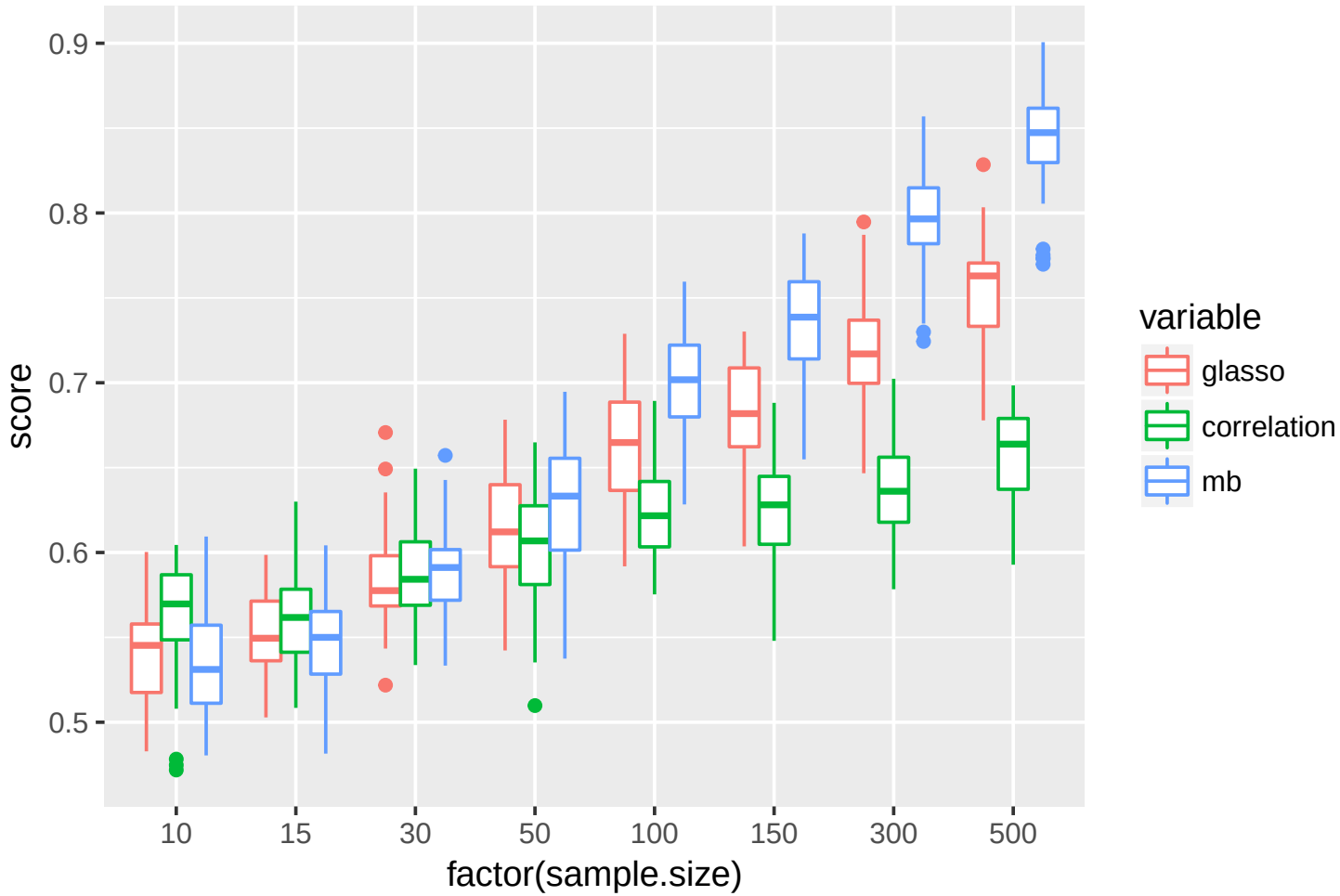
Run

```
suppressMessages(library(parallel))
res <- do.call(rbind, mclapply(1:40, one.simu, mc.cores=4))
```

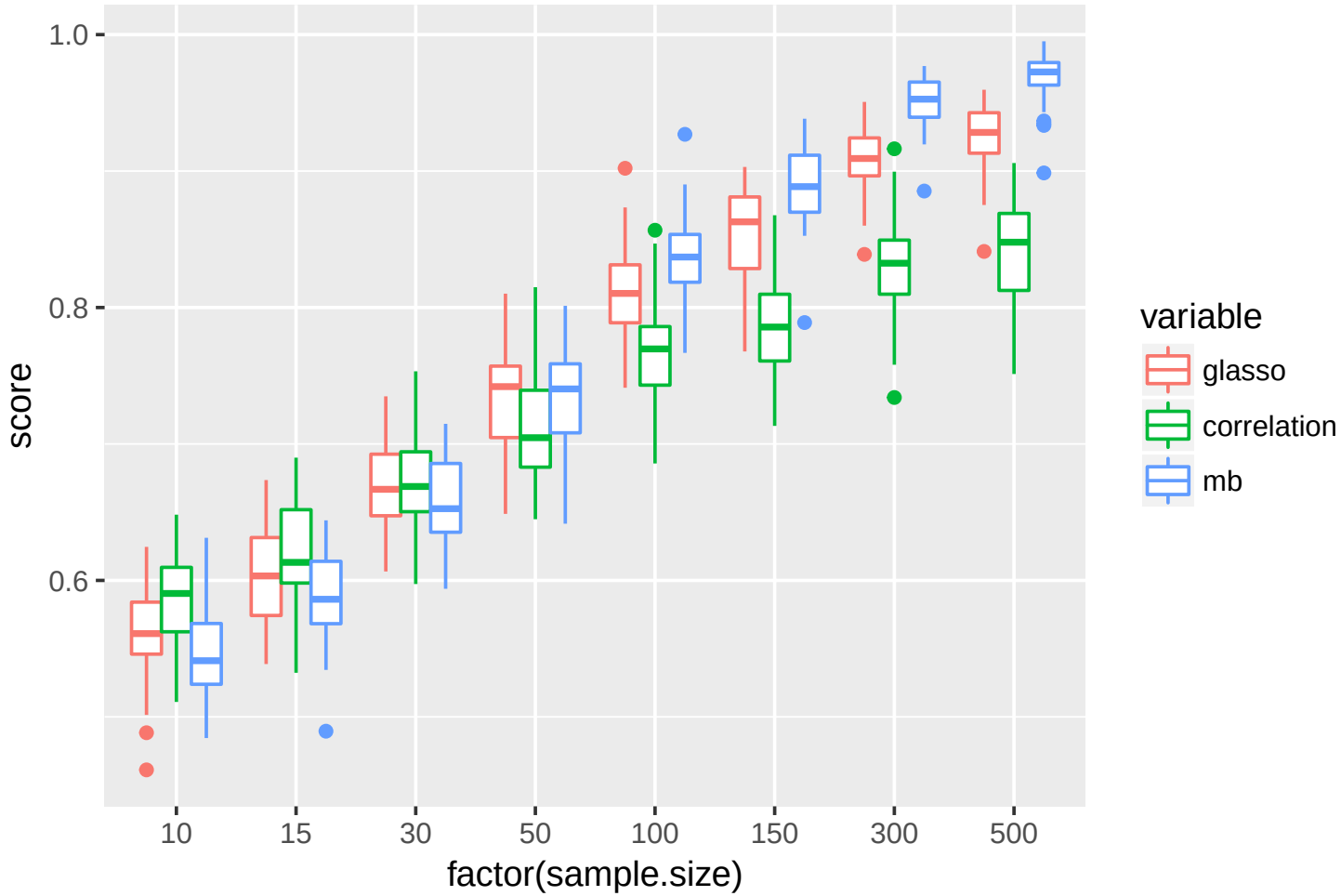
Simulation results (cluster - clique)



Simulation results (cluster, connection probability of 0.5)



Simulation results (random, connection probability of 0.3)



Breast cancer: transcriptomics for ER+ and ER- tumors

We look at a large public datasets from Guedj et al. 2011 with two main subgroups

- Estrogen receptor positive
- Estrogen receptor negative

```
load ("huge/breast_cancer_guedj11.RData")
load ("huge/gen_name.RData")
gene.name <- unlist(gene.name)
data.raw <- expr
table(class.ER)

## class.ER
## ERm ERp
## 162 375
```

Filtering Unknown genes

```
toDiscard <- which(gene.name == "Not.Known")
gene.name <- gene.name[-toDiscard]
data.raw <- data.raw[-toDiscard, ]
```

We get

```
dim(data.raw)

## [1] 41248   537
```

Differential analysis

Do we detect some gene expression differences ?

```
load ("huge/breast_cancer_guedj11.RData")
suppressMessages(library(limma))
design <- cbind(Moy=1, Erp=(class.ER == "ERp")+0)
fit <- lmFit(data.raw, design=design)
fit <- eBayes(fit)
res <- topTable(fit, coef="Erp", number=10^5,
                genelist=fit$genes, adjust.method="BH",
              sort.by="none", resort.by=NULL,
              p.value=1, lfc=0, confint=FALSE)
```

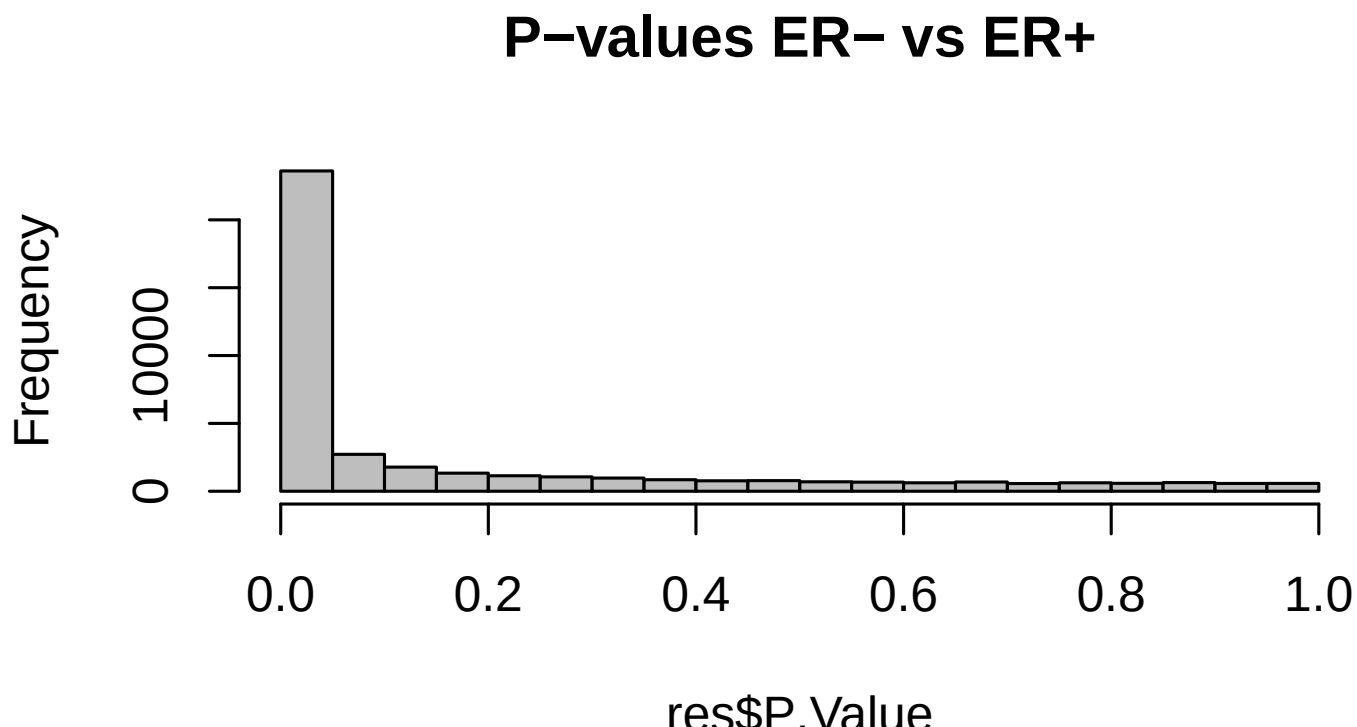
Many genes are differentially expressed

- The histogram of p-values looks good
- This is a well known fact (ER+ and ER- are very different)

```
sum(res$adj.P.Val < 10^-5)

## [1] 5907

hist(res$P.Value, breaks=30, col="grey",
     main="P-values ER- vs ER+")
```



What to do with this list of genes?

ESR1 has the most significant p-values

```
gene.name[order(res$adj.P.Val)[1]]

## 205225_at
##    "ESR1"
```

Network analysis

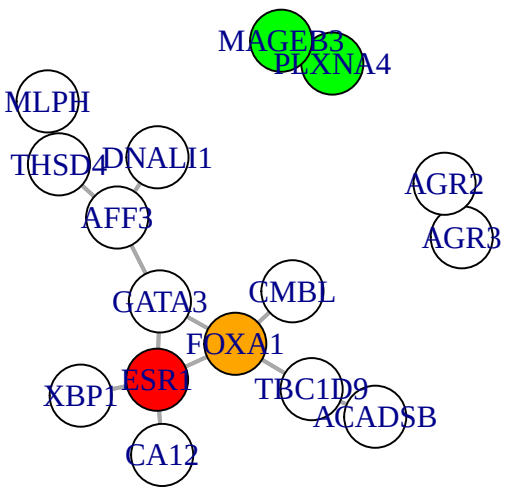
- Could we find partners of ESR1 that are specific to ER+?
- We cannot infer a network on 41000 genes (Verzelen 2011)
  - ⇝ Most differentially expressed genes
  - ⇝ Most varying genes
  - ⇝ Look at a specific pathway ...

Selecting some probes

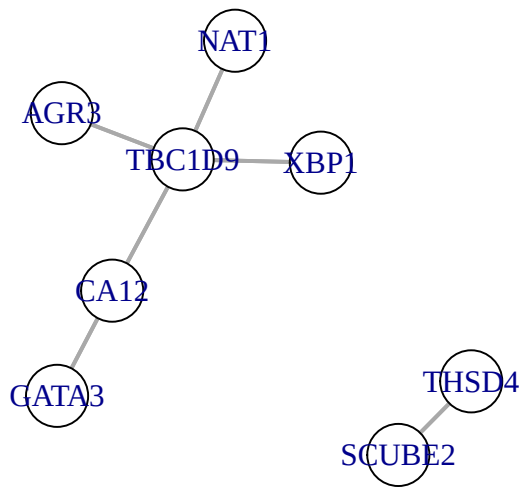
Take the 20 most differentially expressed plus some random

```
## Error in graph_from_adjacency_matrix(net_Mspec_):  could not find function
"graph_from_adjacency_matrix"
## Error in graph_from_adjacency_matrix(net_Pspec_):  could not find function
"graph_from_adjacency_matrix"
```

ER+ specific



ER− specific



FOXA1, ESR1, GATA3 a well known interaction

1. FOXA1 is a key determinant of estrogen receptor function and endocrine response. Antoni Hurtado et al. 2011 (Nat. Genet.):
   $\rightsquigarrow$ "FOXA1 is a key determinant that can influence differential interactions between ER and chromatin"
2. GATA3 acts upstream of FOXA1 in mediating ESR1 binding by shaping enhancer accessibility. Theodorou et al. 2013 (Genome Res.)
3. Estrogen receptor regulation of carbonic anhydrase XII through a distal enhancer in breast cancer. Barnett DH et al 2008 (Cancer Res.)
   $\rightsquigarrow$ "we show that CA12 is robustly regulated by estrogen via ER alpha in breast cancer cells"

Part II

Extensions

4. Accounting for latent organisation of the network
5. Accouting for sample heterogeneity
6. Model for count data

Extensions motivated by biological data

Strengthen the inference by

- accounting for biological features
  1. structure of the network (organization of biological mechanisms)
  2. sample heterogeneity (structure of the population)
  3. horizontal integration (use multiple data and platforms)
  4. Deal with covariates
- accounting for data features
  1. What if some important actor is missing?
  2. Extend to non strictly normal distribution
  3. Deal with a large number of actors

$\rightsquigarrow$ How? Essentially by crafting the regularization according to our prior knowledge

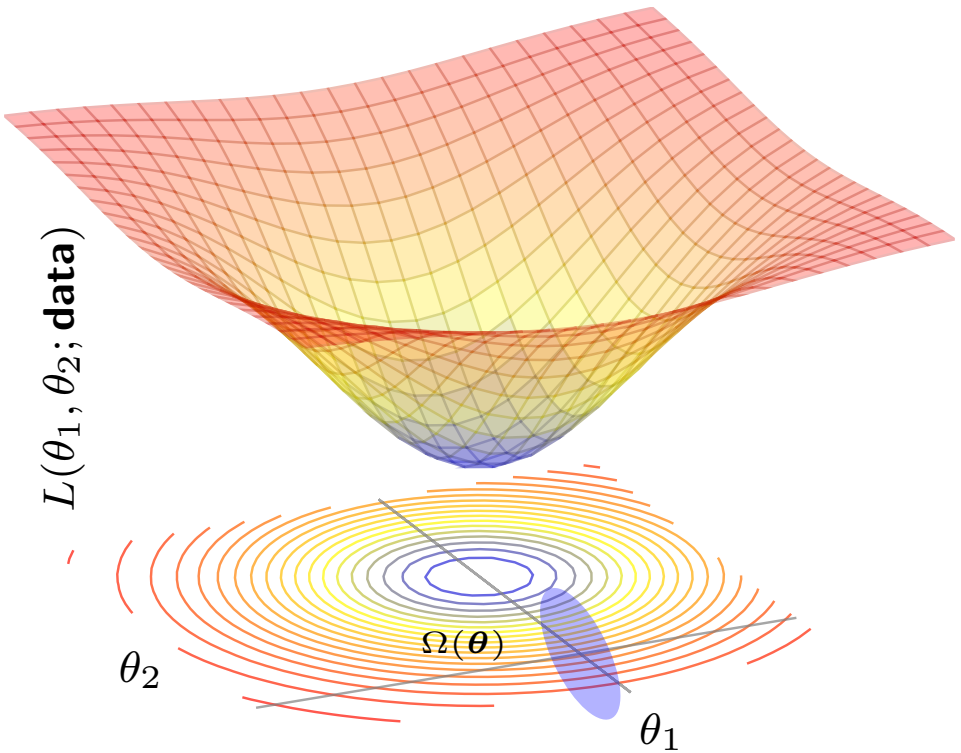
General strategy

Revisit "traditional" statistical methods under the light of optimization

1. statistical problem $\leftrightarrow$ optimization problem

$$\underset{\boldsymbol{\theta}}{\text{minimize}}\, L(\boldsymbol{\theta}; \textbf{data}) \quad \text{s.t.} \quad \Omega(\boldsymbol{\theta}) \leq c.$$

2. modification of the original problem/regularization



modify Ω and/or L to

- control the computational cost
- control the model complexity
- account for prior knowledge

looking for

$\rightsquigarrow$ $\uparrow$ performance and interpretability

$\rightsquigarrow$ trade-off between speed and accuracy

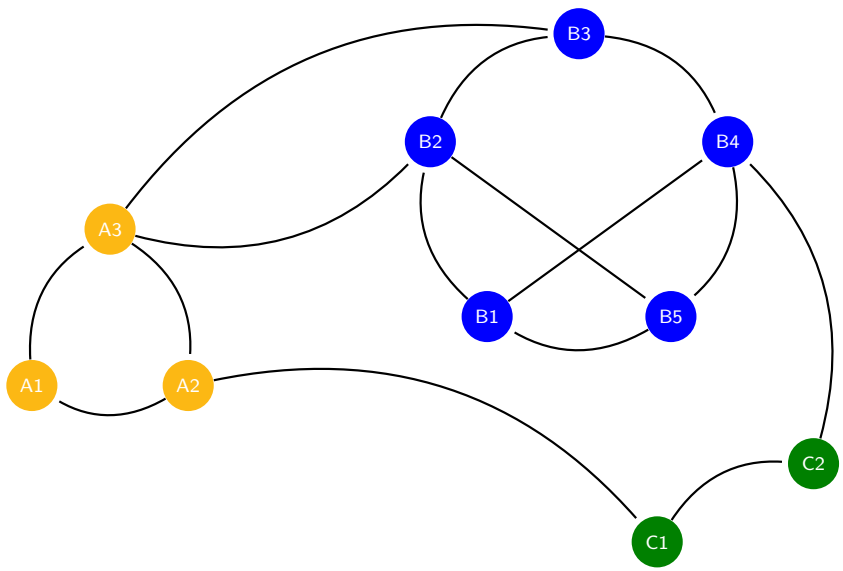
Outline

Handling with the data structure and scarcity

By introducing some prior

Priors should be biologically grounded

1. no too many genes effectively interact: sparsity,
2. networks are organized: latent clustering.



Structured regularization

SIMoNe: Statistical Inference for MOdular NEtworks

$$\arg\max_{\mathbf{\Theta},\mathbf{Z}} \ell(\mathbf{\Theta};\mathbf{Y}) - \lambda \, \|\mathbf{P}_{\mathbf{Z}} \star \mathbf{\Theta}\|_{\ell_1},$$

where $\mathbf{P}_{\mathbf{Z}}$ is a matrix of weights depending on a underlying latent structure $\mathbf{Z}$ (depicted through a stochastic block model).

$\rightsquigarrow$ Cluster-driven inference via an EM-like strategy.


Ambroise, Chiquet, Matias. Inferring sparse GGM with latent structure, EJS, 2009.


Marlin, Schmidt, Murphy: similar Bayesian work UCI 2010.


Wong et al., close update: *Adaptive Graphical Lasso*, 2014.


Chiquet et al., SIMoNe R-package *(needs updates. . .)*, Note Bioinformatics, 2009.

How to come up with a latent clustering?

Biological expertise

- Build $\mathbf{Z}$ from prior biological information
  - transcription factors vs. regulatees,
  - number of potential binding sites,
  - KEGG pathways, ...
- Build the weight matrix from $\mathbf{Z}$.

Inference: Erdös-Rényi **Mix**ture for **Net**works (Daudin et al., 2008; Latouche et al., 2011)

- Equivalent to the Stochastic Bloc Model (SBM);
- Spread the nodes into $\mathcal{Q}$ classes;
- Connexion probabilities depend upon node classes:

$$\mathbb{P}(i \leftrightarrow j | i \in \text{class } q, j \in \text{class } \ell) = \pi_{q\ell}.$$

- Build $P_{\mathbf{Z}} \propto 1 - \pi_{q\ell}$.

Illustration on breast Cancer

Prediction of the outcome of preoperative chemotherapy

Hess *et al.* Journal. of Clinical Oncology, 2006.

Data set

133 patients classified as

1. pathologic complete response,
2. residual disease,

according to a signature of 26 genes (small network).

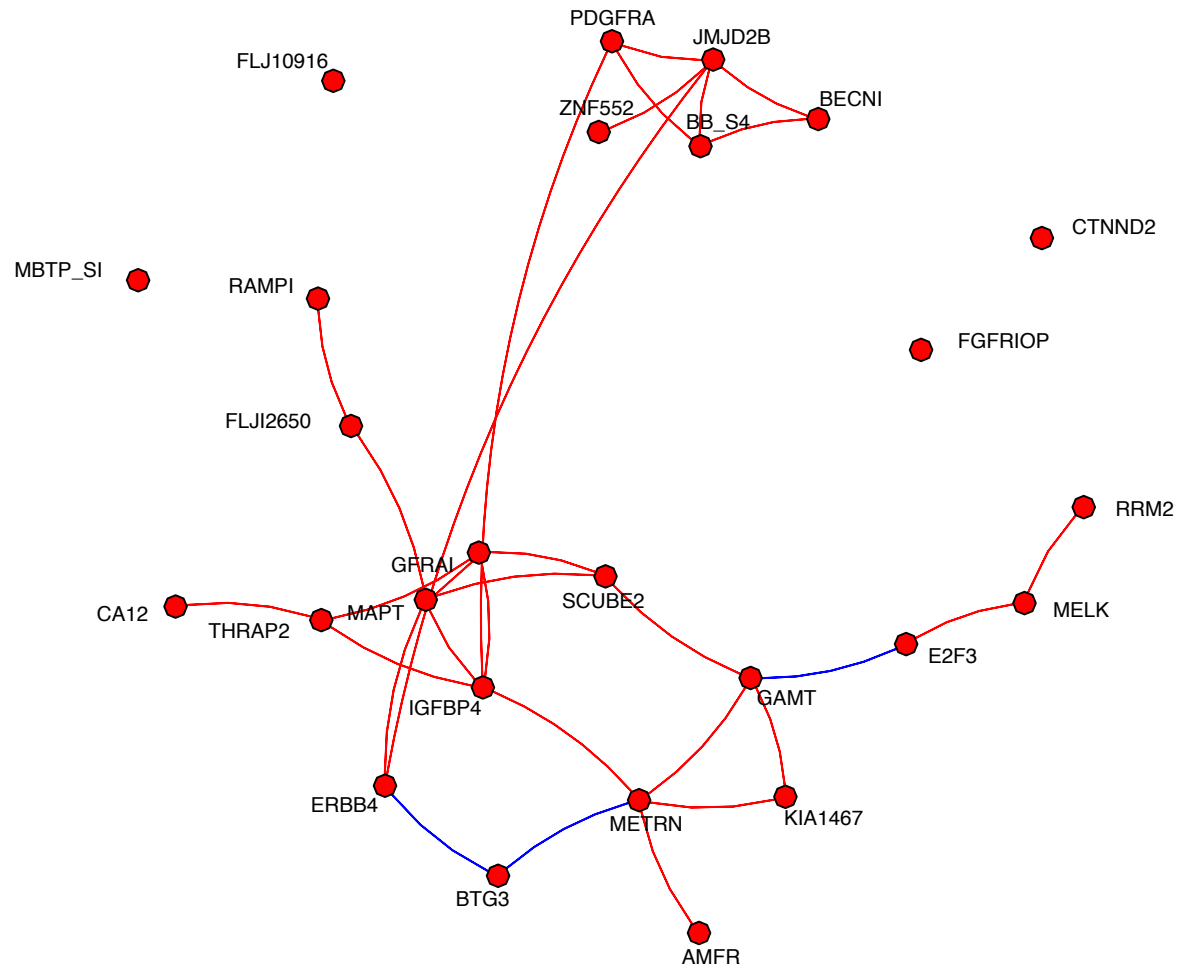


Figure: Pooling the data, Neighborhood Selection

Illustration on breast Cancer

Prediction of the outcome of preoperative chemotherapy

Hess *et al.*
Journal. of Clinical Oncology, 2006.

Data set

133 patients classified as

1. pathologic complete response,
2. residual disease,

according to a signature of 26 genes (small network).

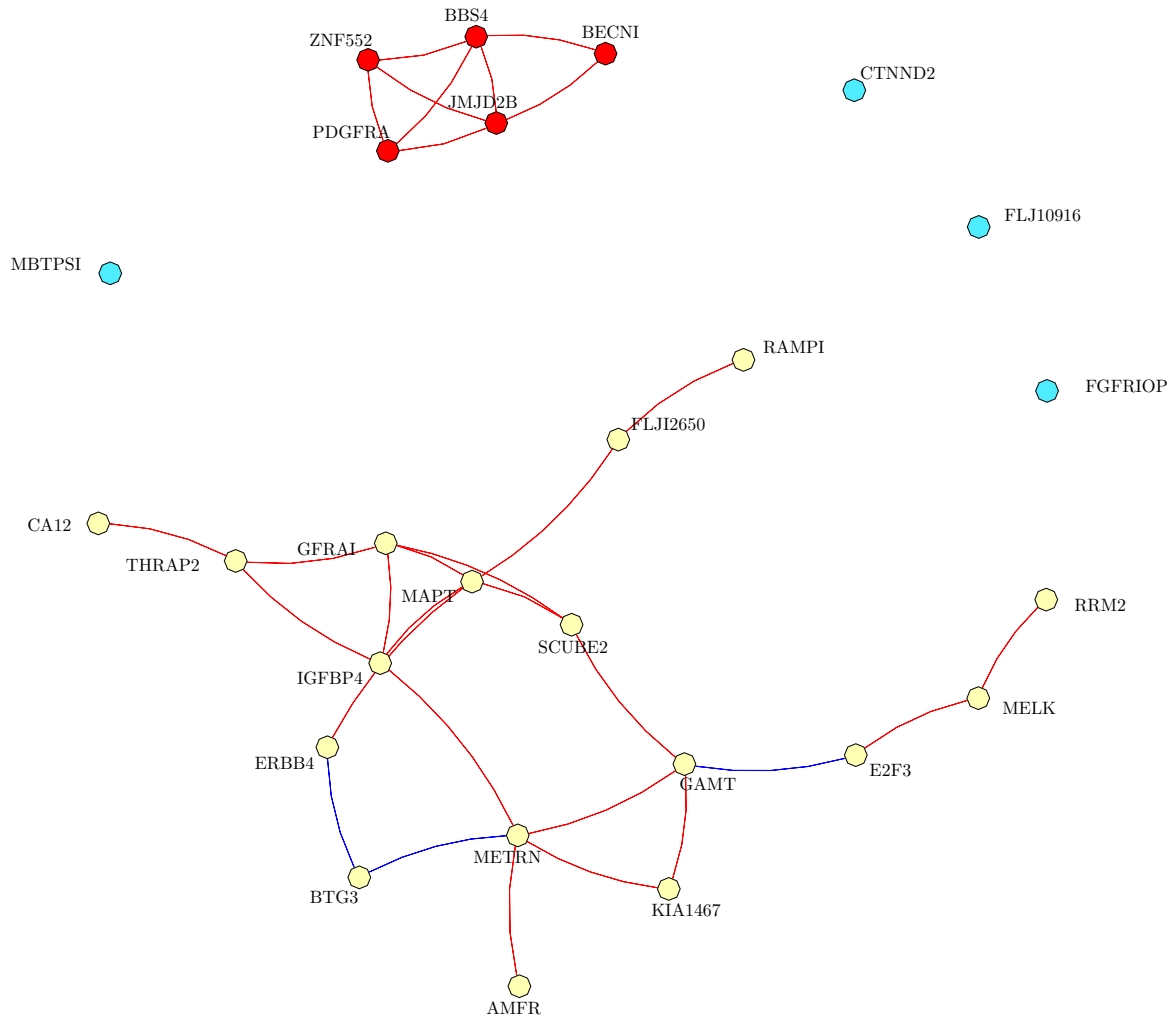


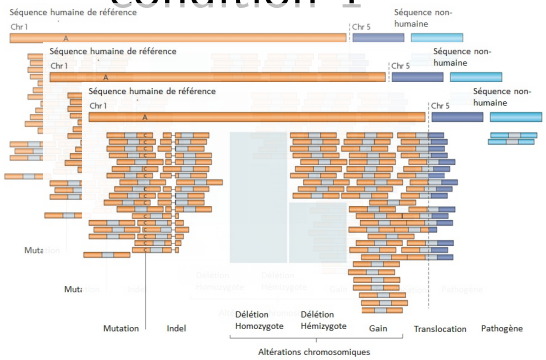
Figure: Pooling the data, SIMoNE with clustering

Outline

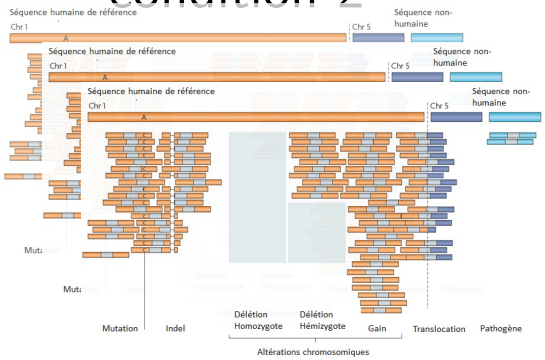
Handling scarcity and heterogeneity of data

Merge several experimental conditions

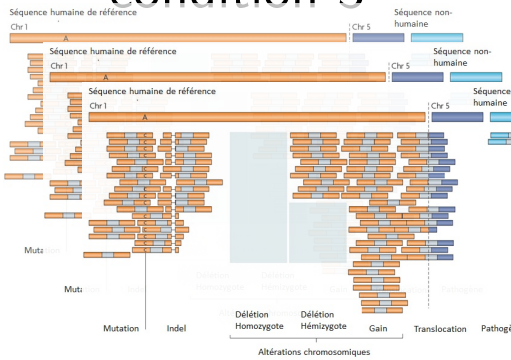
condition 1



condition 2



condition 3



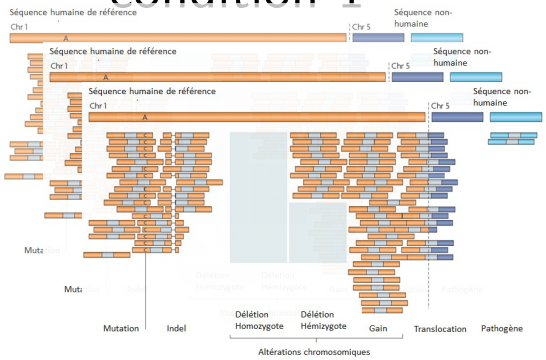
Multiple inference of GGM

$$\arg\max_{\mathbf{\Theta}^{(c)}, c=1\ldots,C} \sum_{c=1}^{C} \ell(\mathbf{\Theta}^{(c)}; \mathbf{S}^{(c)}) - \lambda \, \mathrm{pen}_{\ell_1}(\mathbf{\Theta}^{(c)}).$$

Handling scarcity and heterogeneity of data

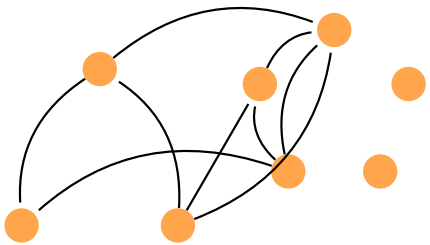
Inferring each graph independently does not help

condition 1

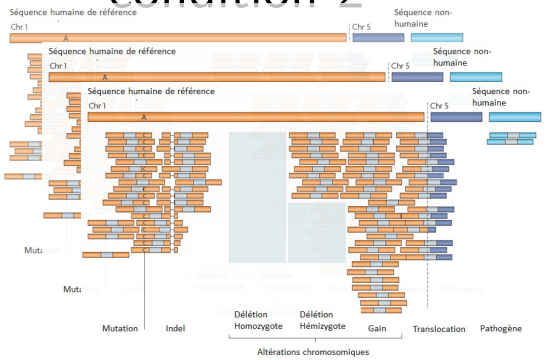


$(Y_1^{(1)}, \ldots, Y_{n_1}^{(1)})$

inference

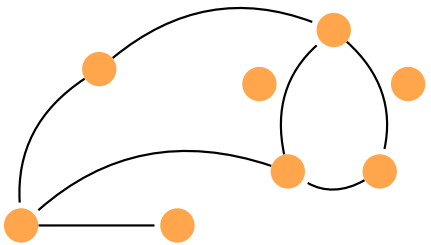


condition 2

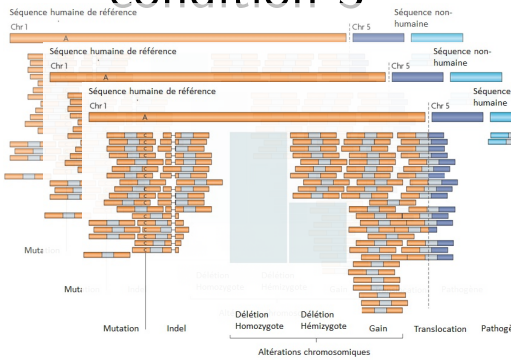


$(Y_1^{(2)}, \ldots, Y_{n_2}^{(2)})$

inference

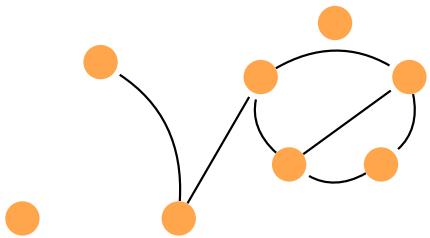


condition 3



$(Y_1^{(3)}, \ldots, Y_{n_3}^{(3)})$

inference



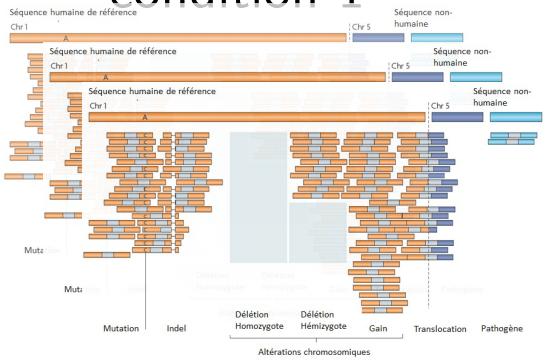
Multiple inference of GGM

$$\sum_{c}$$

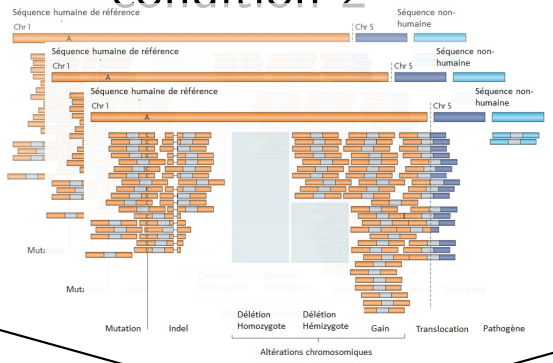
Handling scarcity and heterogeneity of data

By pooling all the available data (like we just have with Hess' data set)

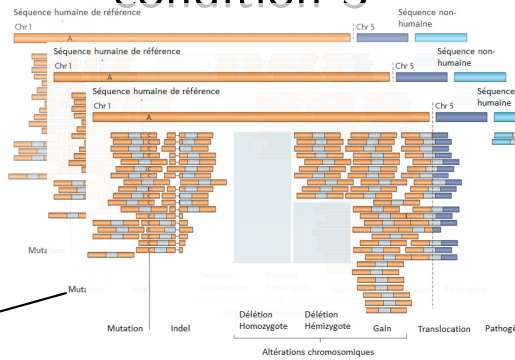
condition 1



condition 2

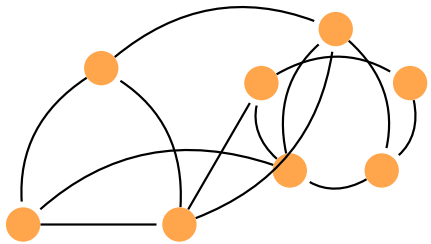


condition 3



$$(Y_1, \ldots, Y_n),\ n = n_1 + n_2 + n_3.$$

inference

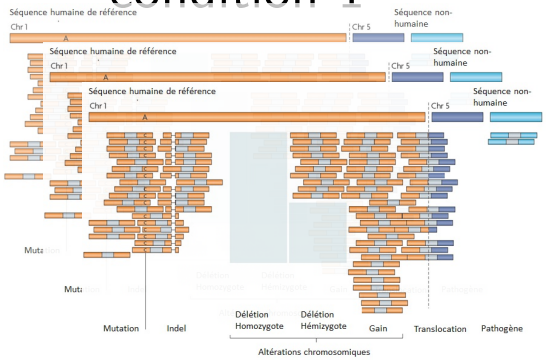


Multiple inference of GGM

Handling scarcity and heterogeneity of data

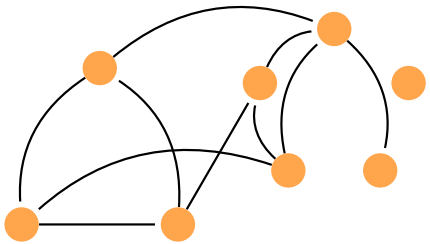
By breaking the separability

condition 1

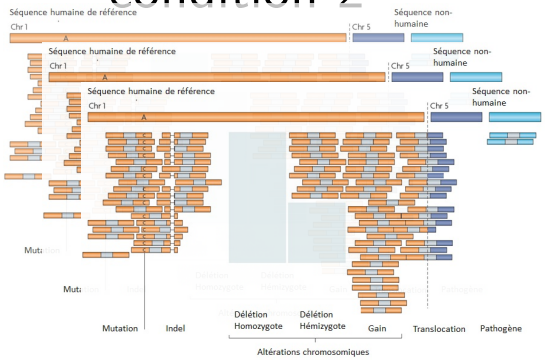


$(Y_1^{(1)}, \ldots, Y_{n_1}^{(1)})$

inference

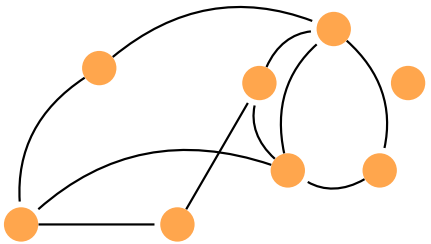


condition 2

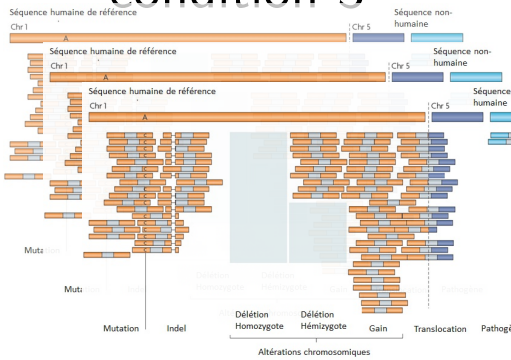


$(Y_1^{(2)}, \ldots, Y_{n_2}^{(2)})$

inference

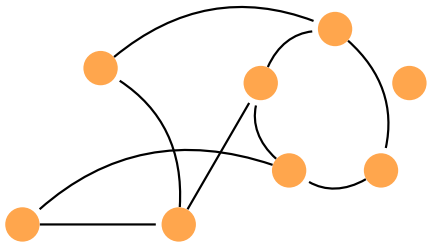


condition 3



$(Y_1^{(3)}, \ldots, Y_{n_3}^{(3)})$

inference

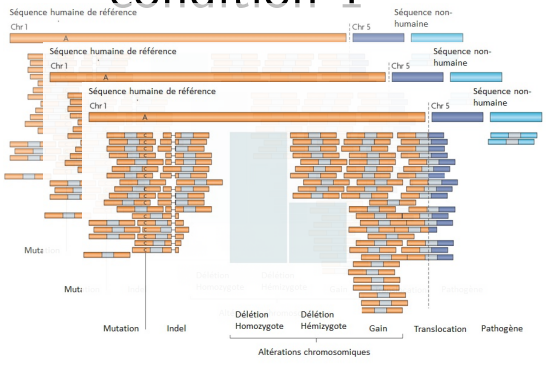


Multiple inference of GGM

Handling scarcity and heterogeneity of data

By breaking the separability

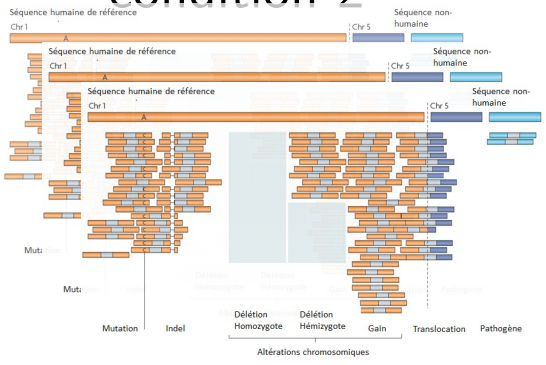
condition 1



$(Y_1^{(1)}, \ldots, Y_{n_1}^{(1)})$

inference

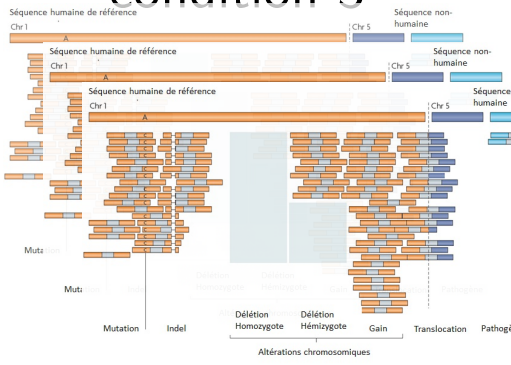
condition 2



$(Y_1^{(2)}, \ldots, Y_{n_2}^{(2)})$

inference

condition 3



$(Y_1^{(3)}, \ldots, Y_{n_3}^{(3)})$

inference

Multiple inference of GGM

$$\underset{\mathbf{\Theta}^{(c)}, c=1\ldots,C}{\arg\max} \sum_{c=1}^{C} \ell(\mathbf{\Theta}^{(c)}; \mathbf{S}^{(c)}) - \lambda \, \mathrm{pen}_{\ell_1}(\mathbf{\Theta}^{(c)}).$$

A multitask approach

Chiquet, Grandvalet, Ambroise, Statistics and Computing 2010/11

Break the separability

Joint the optimization problem by either modifying

$$\underset{\mathbf{\Theta}^{(c)}, c=1\ldots,C}{\arg\max} \sum_{c=1}^{C} \tilde{\ell}(\mathbf{\Theta}^{(c)}; \tilde{\mathbf{S}}^{(c)}) - \lambda \ \mathrm{pen}_{\ell_1}(\mathbf{\Theta}^{(c)}).$$

1. the fitting term
2. the regularization term

A multitask approach

Chiquet, Grandvalet, Ambroise, Statistics and Computing 2010/11

Break the separability

Joint the optimization problem by either modifying

$$\underset{\mathbf{\Theta}^{(c)}, c=1\ldots,C}{\arg\max} \sum_{c=1}^{C} \tilde{\ell}(\mathbf{\Theta}^{(c)}; \tilde{\mathbf{S}}^{(c)}) - \lambda \, \mathrm{pen}_{\ell_1}(\mathbf{\Theta}^{(c)}).$$

1. the fitting term
2. the regularization term

Intertwined-Lasso

- $\overline{\mathbf{S}} = \frac{1}{n} \sum_{t=1}^{T} n_t \mathbf{S}^{(t)}$ is the "pooled-tasks" covariance matrix.
- $\widetilde{\mathbf{S}}^{(t)} = \alpha \mathbf{S}^{(t)} + (1-\alpha)\overline{\mathbf{S}}$ is a mixture between specific and pooled covariance matrices.

A multitask approach

Chiquet, Grandvalet, Ambroise, Statistics and Computing 2010/11

Break the separability

Joint the optimization problem by either modifying

$$\underset{\mathbf{\Theta}^{(c)}, c=1\ldots,C}{\arg\max} \sum_{c=1}^{C} \tilde{\ell}(\mathbf{\Theta}^{(c)}; \tilde{\mathbf{S}}^{(c)}) - \lambda \, \mathrm{pen}_{\ell_1}(\mathbf{\Theta}^{(c)}).$$

1. the fitting term
2. the regularization term

Sparsity with grouping effect

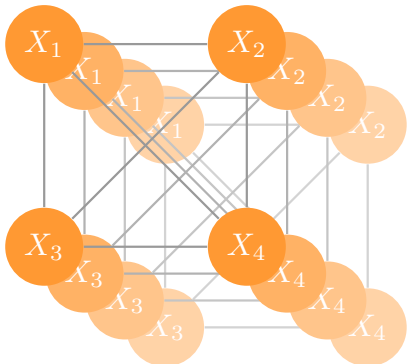
- Group-Lasso (Yuan and Lin 2006, Grandvalet and Canu, 1998),
- Cooperative-Lasso (Chiquet et al, AoAS, 2012),

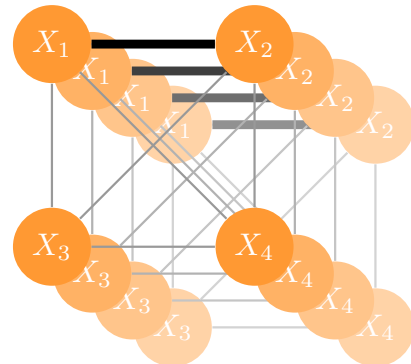
Grouping effects induced

Potential groups

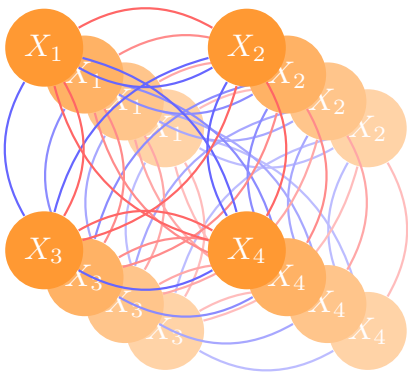
Group(s) induced by edges $(1, 2)$

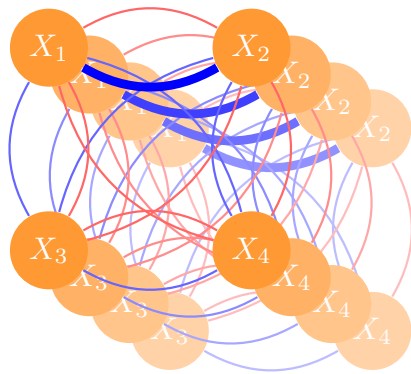
Group-Lasso

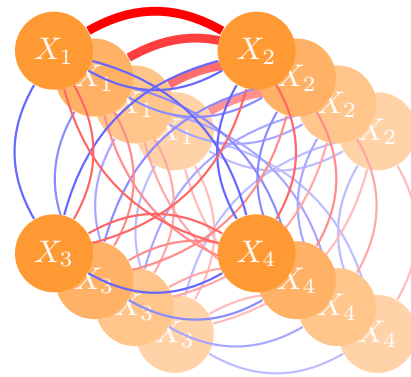




Cooperative-Lasso







Other grouping effects induced

Recent works

- Use Fused-Lasso, sparse group-Lasso
- Adapted several time to the Graphical Lasso framework
  - See, e.g. D. Witten's team works.
  - The multitask/neighborhood selection's approach remains competitive.
- Mohan et al., 2014
  - Networks differences are only due to perturbations at the node level.
  - For instance, a hub is encouraged to be shared across tasks.

Revisiting the Hess *et al.* data set

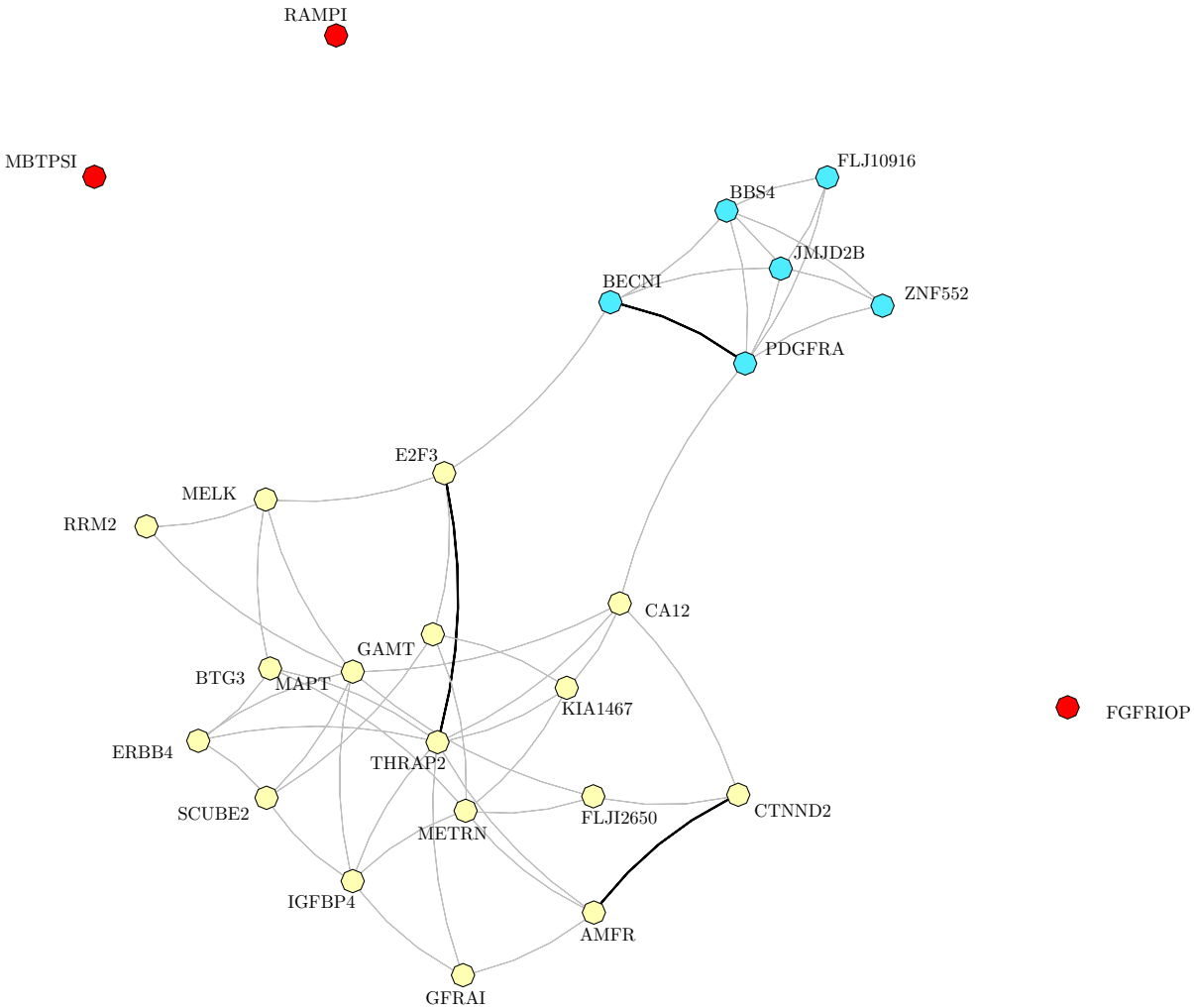


Figure: Cooperative-Lasso applied on the two sets of patients (PCR/noPCR). Bold edges are different in the finally selection graph.

Outline

Motivations: oak powdery mildew pathobiome

Metabarcoding data from [JFS16]

- $n = 116$ leaves, $p = 114$ species (66 bacteria, 47 fungies + *E. alphitoides*)

```
counts[1:3, c(1:4, 48:51)]
```

```
##       f_1 f_2 f_3 f_4 E_alphitoides b_1045 b_109 b_1093
## A1.02  72   5 131   0             0      0     0      0
## A1.03 516  14 362   0             0      0     0      0
## A1.04 305  24 238   0             0      0     0      0
```

- $d = 8$ covariates (tree susceptibility, distance to trunk, orientation, . . .)

```
covariates[1:3, ]
```

```
##              tree distTOtrunk distTOground pmInfection orientation
## A1.02 intermediate         202        155.5           1          SW
## A1.03 intermediate         175        144.5           0          SW
## A1.04 intermediate         168        141.5           0          SW
```

- Sampling effort in each sample (bacteria $\neq$ fungi)

```
offsets[1:3, c(1:4, 48:51)]
```

```
##       f_1  f_2  f_3  f_4 E_alphitoides b_1045 b_109 b_1093
## [1,] 2488 2488 2488 2488          2488   8315  8315   8315
## [2,] 2054 2054 2054 2054          2054    662   662    662
## [3,] 2122 2122 2122 2122          2122    480   480    480
```

Problematic & Basic formalism

Data tables: $\mathbf{Y} = (Y_{ij}), n \times p$; $\mathbf{X} = (X_{ik}), n \times d$; $\mathbf{O} = (O_{ij}), n \times p$ where

- $Y_{ij} =$ abundance (read counts) of species (genes) j in sample i
- $X_{ik} =$ value of covariate k in sample i
- $O_{ij} =$ offset (sampling effort) for species j in sample i

Need for multivariate analysis to

- understand between-species/genes interactions
  $\rightsquigarrow$ 'network' inference (variable/covariance selection)
- correct for technical and confounding effects
  $\rightsquigarrow$ account for covariables and sampling effort

$\rightsquigarrow$ need a generic framework to model dependences between count variables

Models for multivariate count data

If we were in a Gaussian world, the general linear model would be appropriate

For each sample $i = 1, \ldots, n$, it explains

- the abundances of the p species ($\mathbf{Y}_i$)
- by the values of the d covariates $\mathbf{X}_i$ and the p offsets $\mathbf{O}_i$

$$\mathbf{Y}_i = \underbrace{\mathbf{X}_i\mathbf{B}}_{\text{account for covariates}} + \underbrace{\mathbf{O}_i}_{\text{account for sampling effort}} + \boldsymbol{\varepsilon}_i,\ \boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}_p, \underbrace{\boldsymbol{\Sigma}}_{\text{dependence between species}})$$

+ null covariance $\Leftrightarrow$ independence $\rightsquigarrow$ uncorrelated species do not interact

But we are not, and there is no generic model for multivariate counts

- Data transformation ($\log$, $\sqrt{}$) : quick and dirty
- Non-Gaussian multivariate distributions: do not scale to data dimension yet
- Latent variable models: interaction occur in a latent (unobserved) layer

Models for multivariate count data

If we were in a Gaussian world, the general linear model would be appropriate

For each sample $i = 1, \ldots, n$, it explains

- the abundances of the p species ($\mathbf{Y}_i$)
- by the values of the d covariates $\mathbf{X}_i$ and the p offsets $\mathbf{O}_i$

$$\mathbf{Y}_i = \underbrace{\mathbf{X}_i\mathbf{B}}_{\text{account for covariates}} + \underbrace{\mathbf{O}_i}_{\text{account for sampling effort}} + \boldsymbol{\varepsilon}_i,\ \boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}_p, \underbrace{\boldsymbol{\Sigma}}_{\text{dependence between species}})$$

+ ~~null covariance $\Leftrightarrow$ independence $\rightsquigarrow$ uncorrelated species do not interact~~

But we are not, and there is no generic model for multivariate counts

- Data transformation ($\log$, $\sqrt{}$) : quick and dirty
- Non-Gaussian multivariate distributions: do not scale to data dimension yet
- Latent variable models: interaction occur in a latent (unobserved) layer

Poisson-log normal (PLN) distribution

A latent Gaussian model

Originally proposed by Atchisson [AiH89]

$$\mathbf{Z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$$

$$\mathbf{Y}_i \,|\, \mathbf{Z}_i \sim \mathcal{P}(\exp\{\mathbf{O}_i + \mathbf{X}_i^\intercal \mathbf{B} + \mathbf{Z}_i\})$$

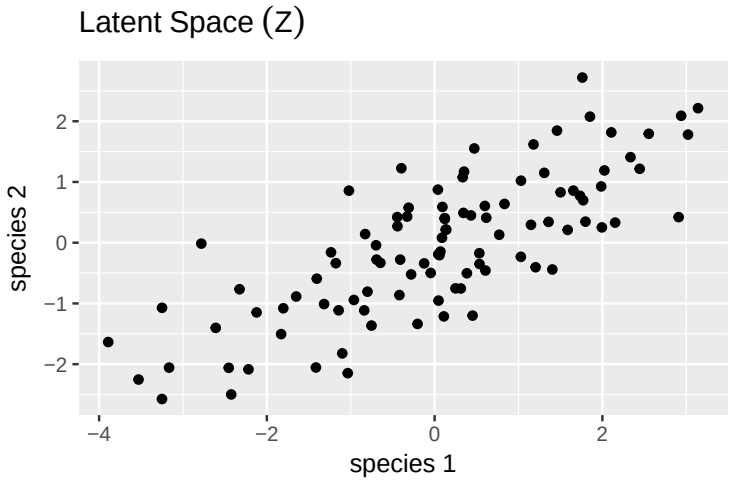
Interpretation

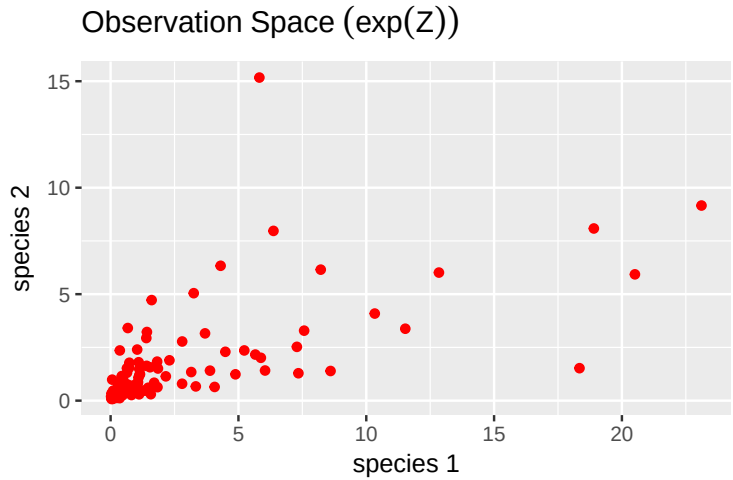
- Dependency structure encoded in the latent space (i.e. in $\mathbf{\Sigma}$)
- Additional effects are fixed
- Conditional Poisson distribution = noise model

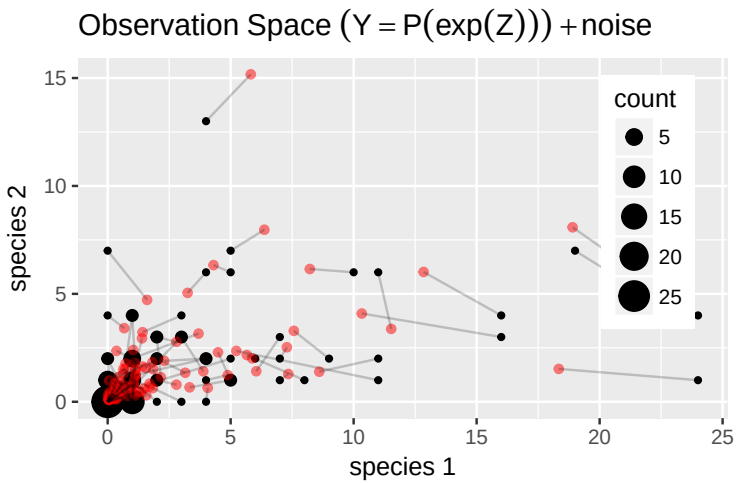
Properties

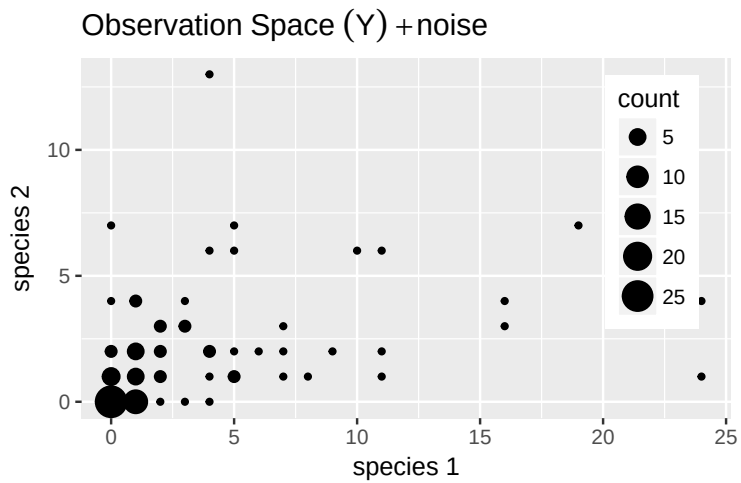
+ over-dispersion
+ covariance with arbitrary signs
- maximum likelihood via EM algorithm is limited to a couple of variables

Geometrical view









Our contributions

Algorithm/Numerical

A variational approach coupled with convex optimization techniques suited to higher dimensional data sets.
PLNmodels R/C++-package: `https://github.com/jchiquet/PLNmodels`

Extensions for multivariate analysis

Idea: put some additional constraint on the residual variance.

- Network Inference
  $\rightsquigarrow$ select direct interaction in $\boldsymbol{\Sigma}^{-1}$ via sparsity constraints
- *Principal component analysis*
  constraint the rank of $\boldsymbol{\Sigma}$ (most important effect in the variance)

Challenge: a variant of the variational algorithm is required for each model

PLN-network: unravel important interactions

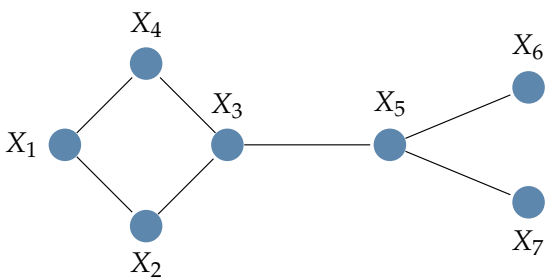
Variable selection of direct effects.

$$\mathbf{Z}_i \text{ iid} \sim \mathcal{N}_p(\mathbf{0}_p, \boldsymbol{\Sigma}), \qquad \|\boldsymbol{\Sigma}^{-1}\|_1 \leq c$$
$$\mathbf{Y}_i \,|\, \mathbf{Z}_i \sim \mathcal{P}(\exp\{\mathbf{O}_i + \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\})$$

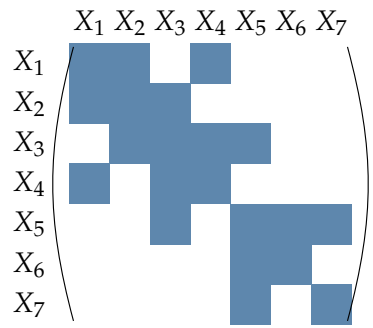
Interpretation: conditional independence structure.

$$(i,j) \notin \mathcal{E} \Leftrightarrow Z_i \perp\!\!\!\perp Z_j | Z_{\setminus\{i,j\}} \Leftrightarrow \boldsymbol{\Sigma}^{-1}_{ij} = 0.$$

$\mathcal{G} = (\mathcal{P}, \mathcal{E})$



$\boldsymbol{\Sigma}^{-1}$



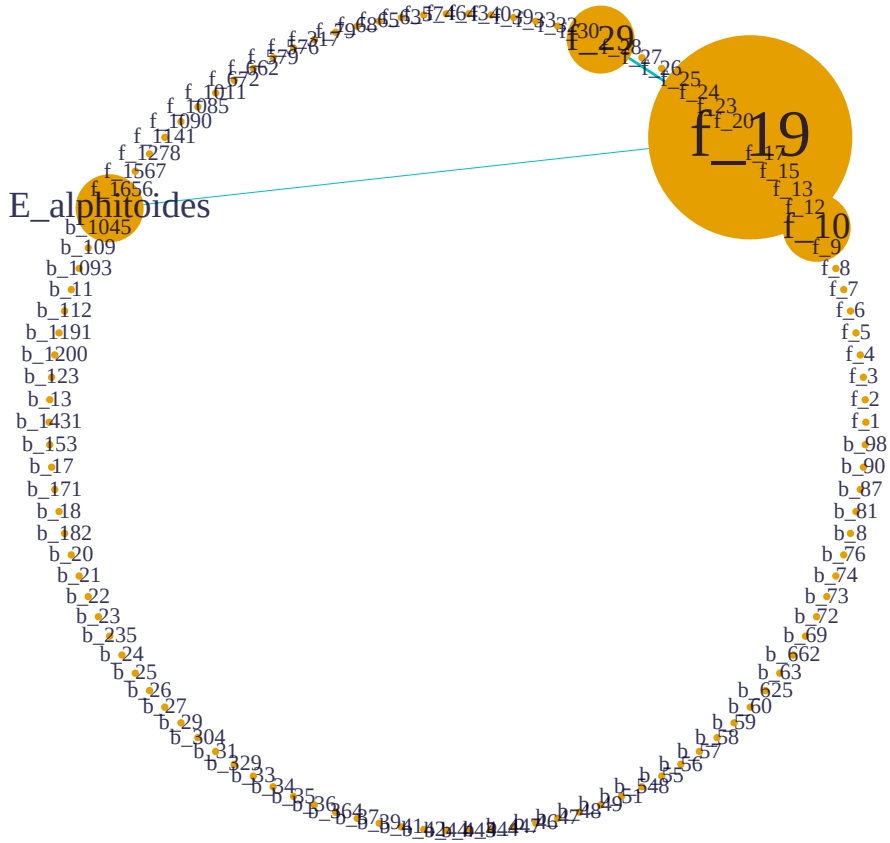
PLN-network: find a sparse reconstruction of the latent inverse covariance
Iterate over variational estimator and Graphical-Lasso [BDE08,YL08,FHT07] in the latent layer

Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```

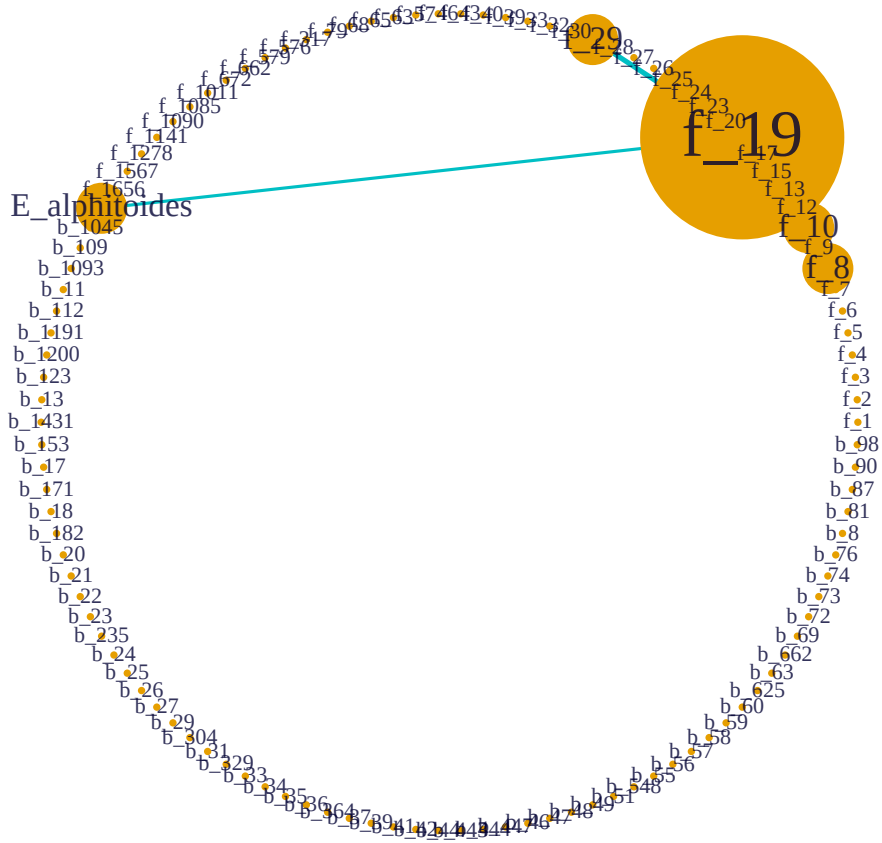
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



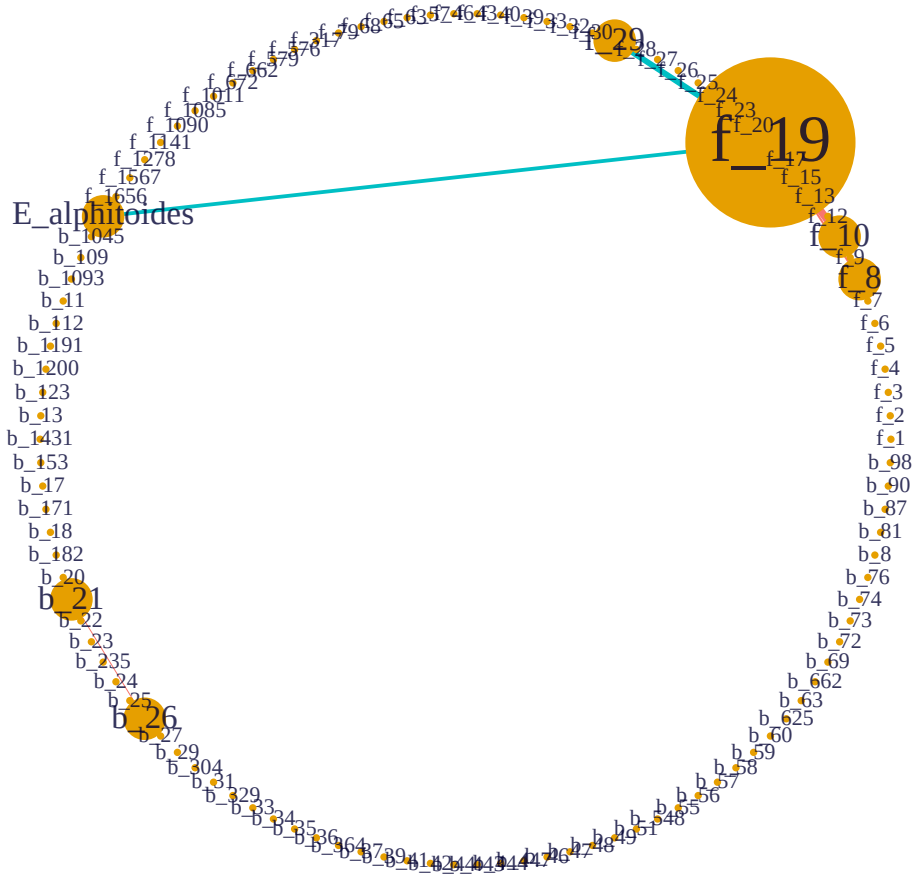
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



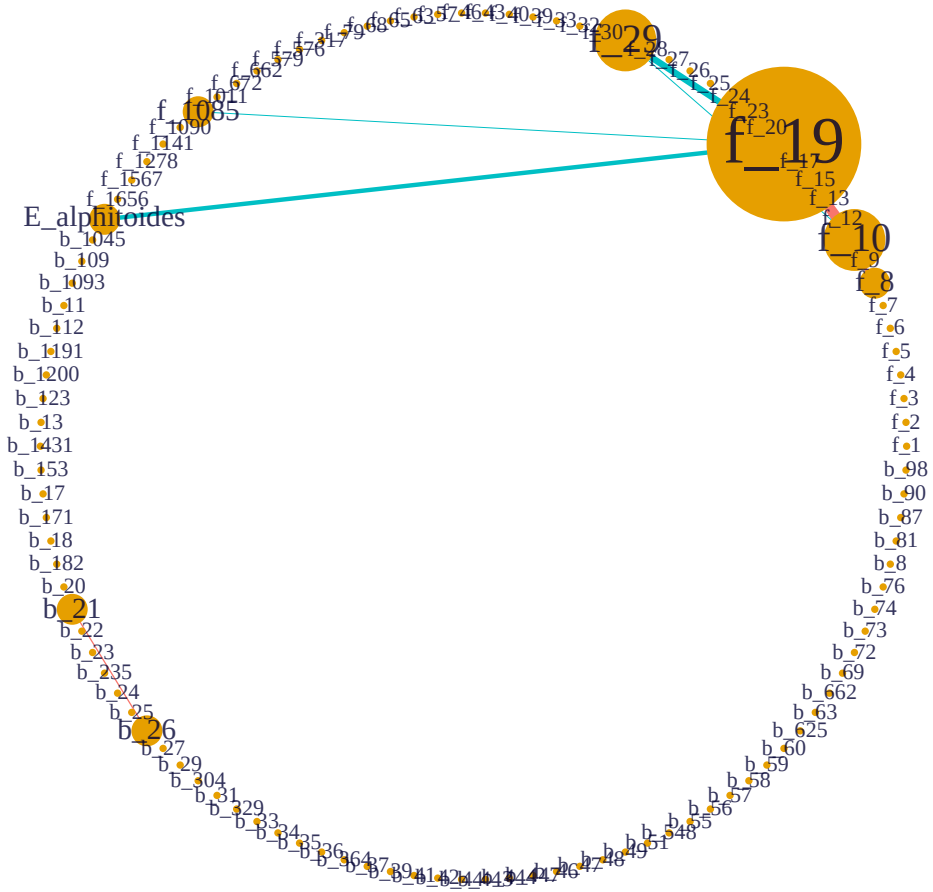
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



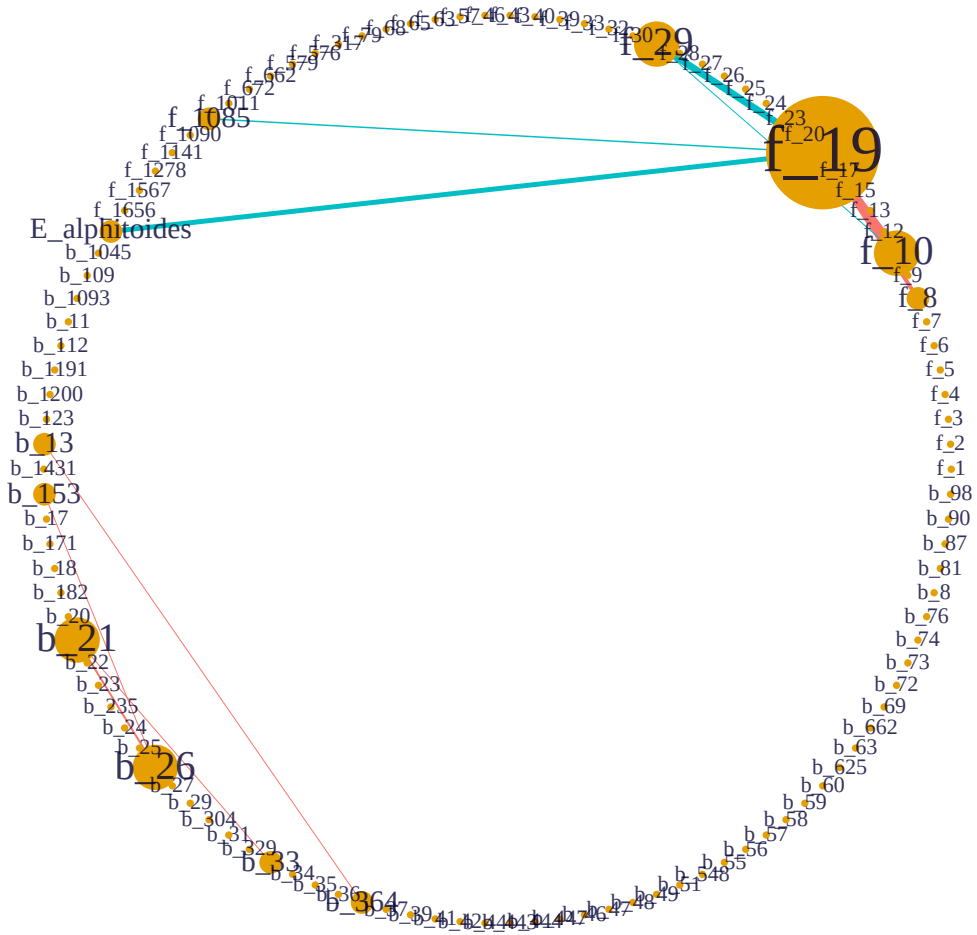
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



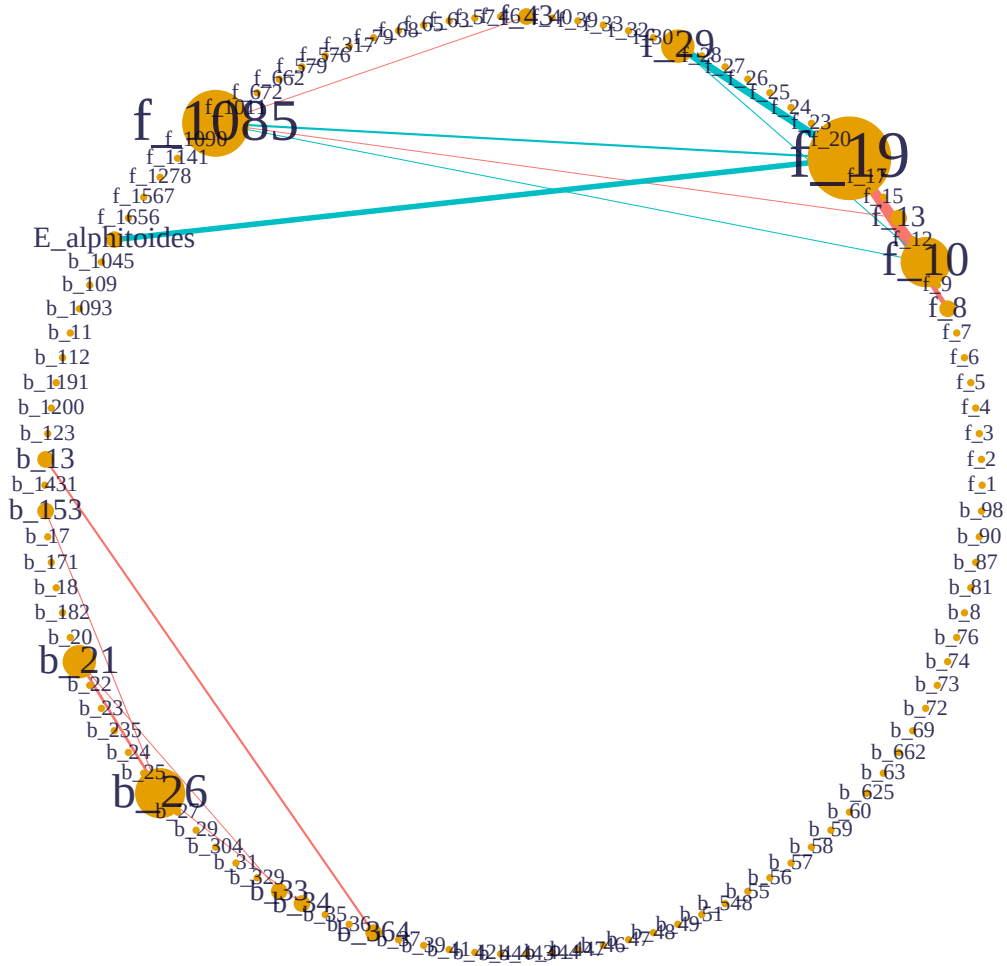
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



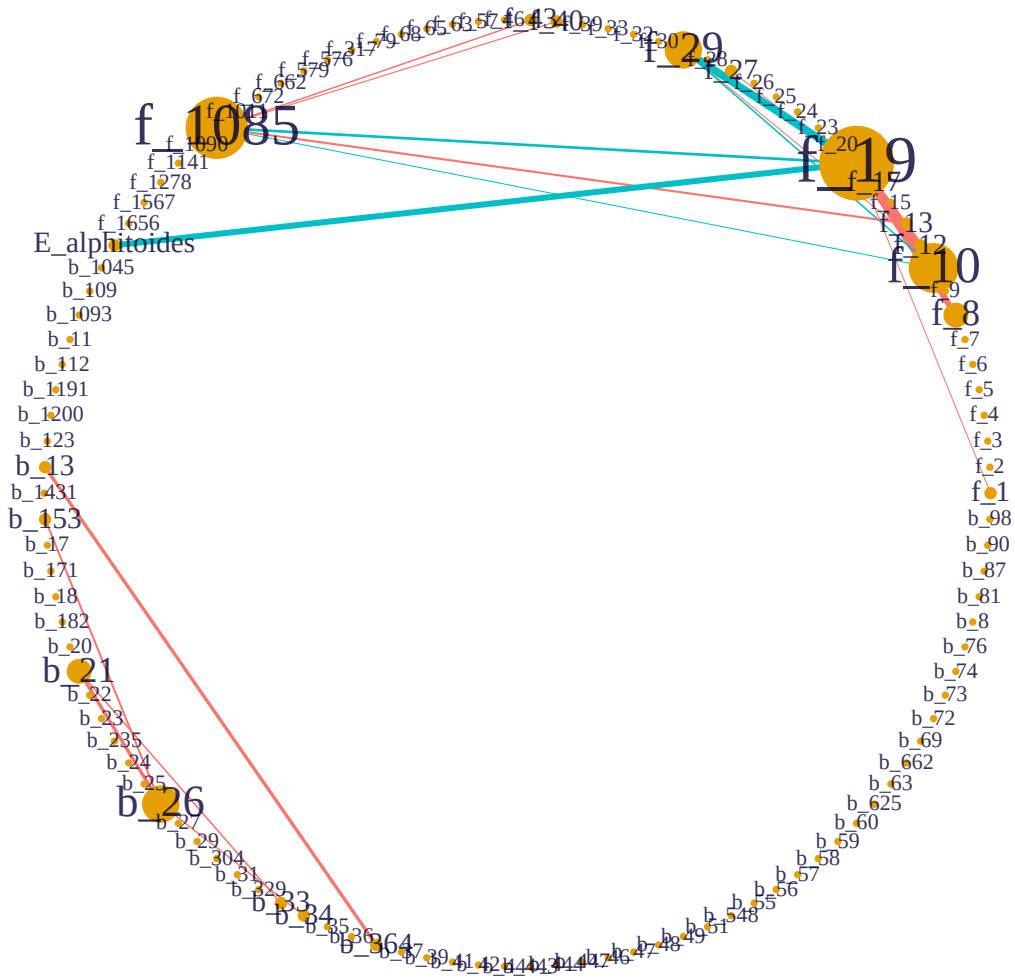
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



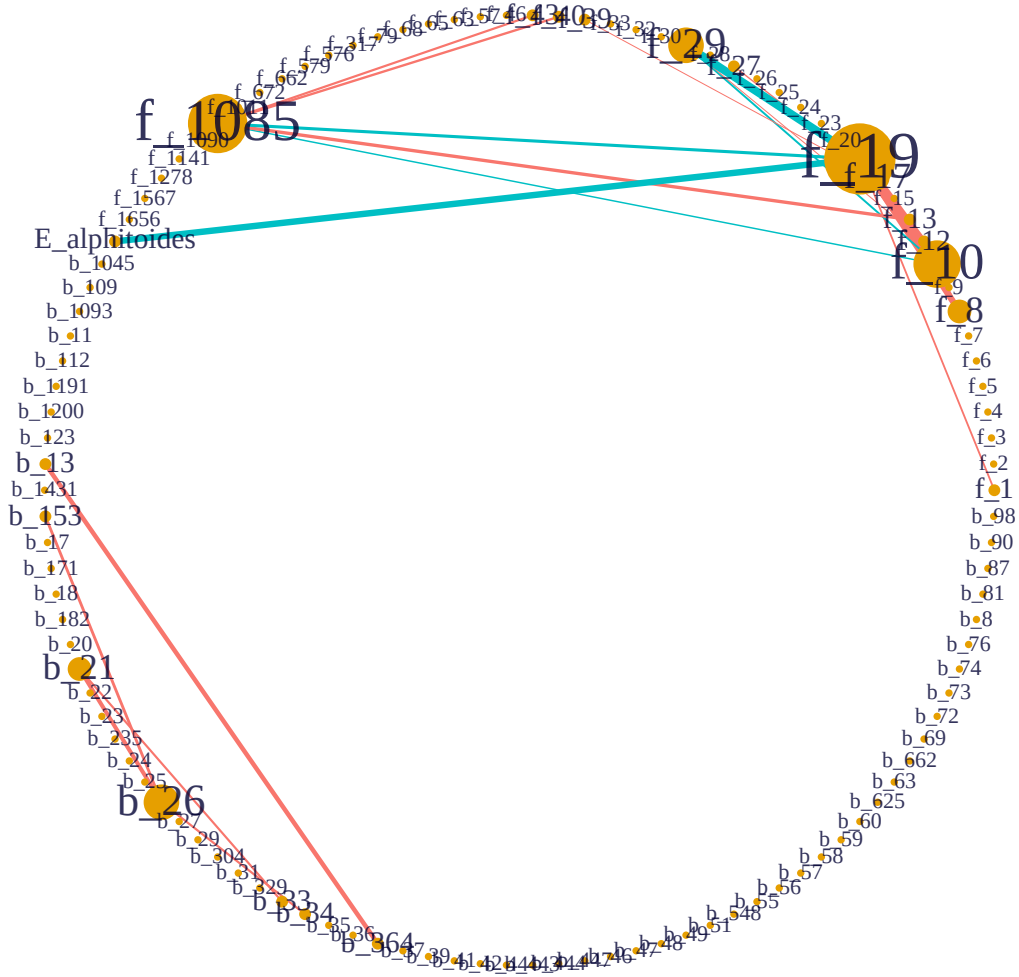
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



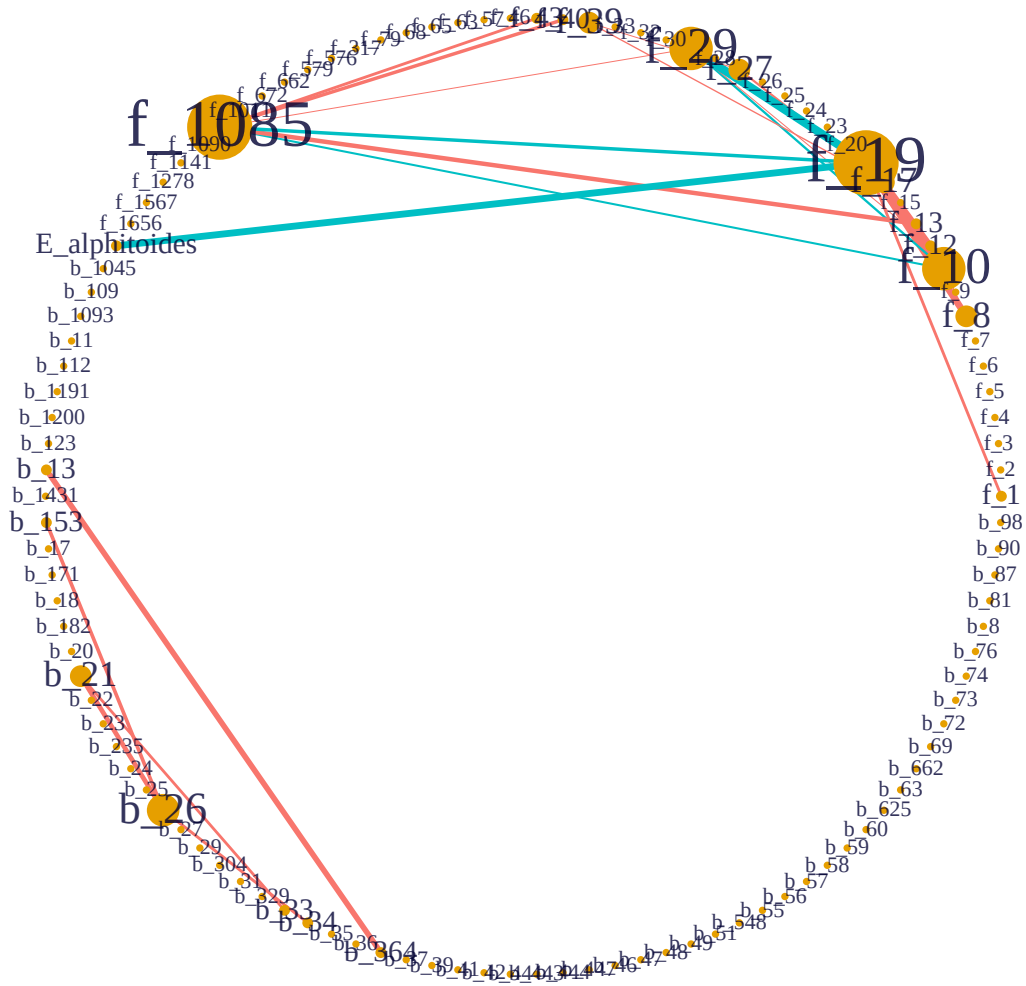
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



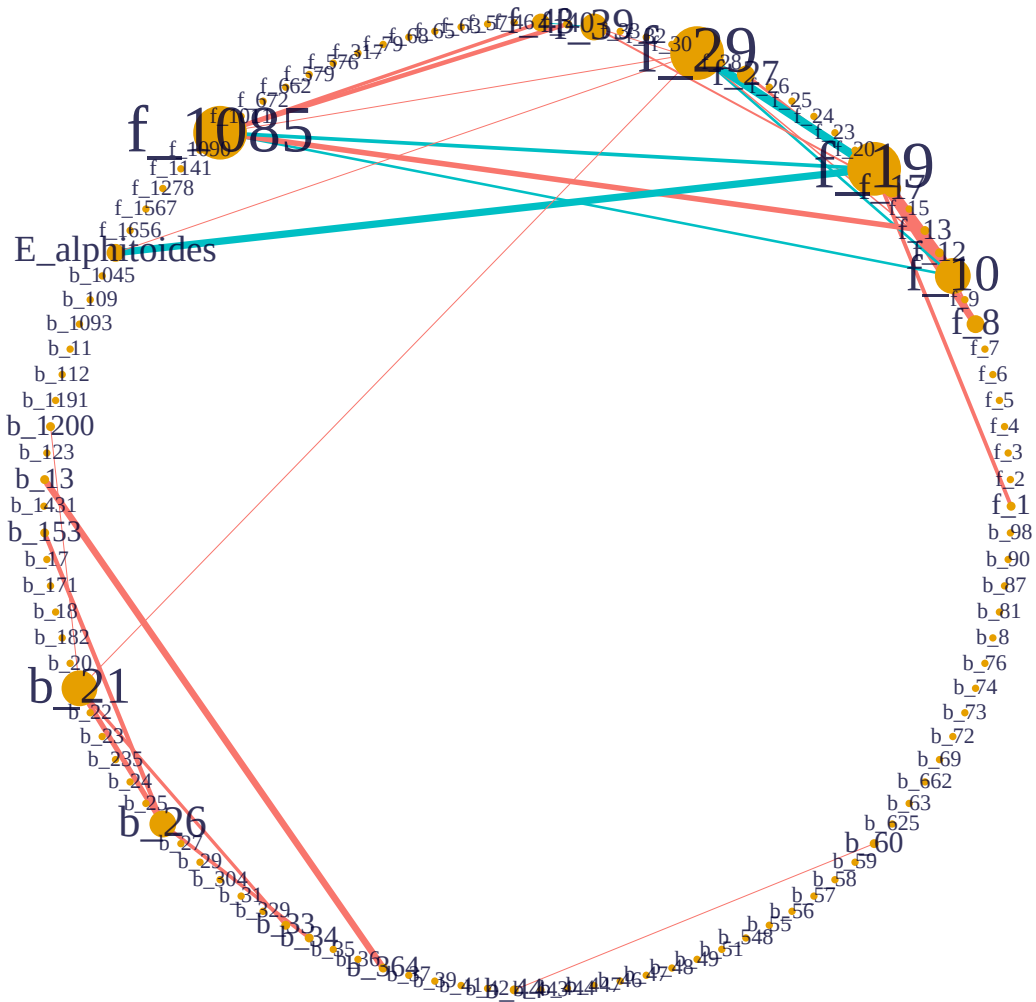
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



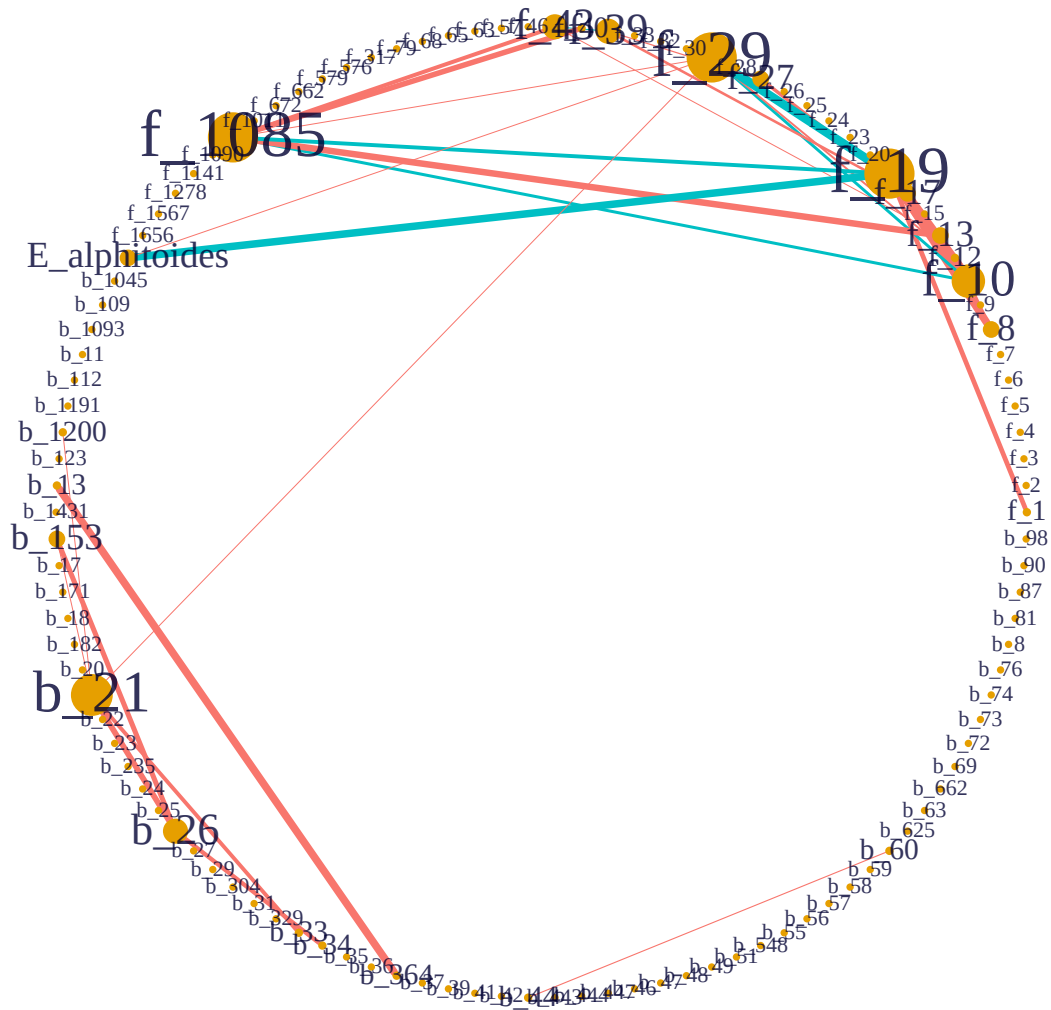
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



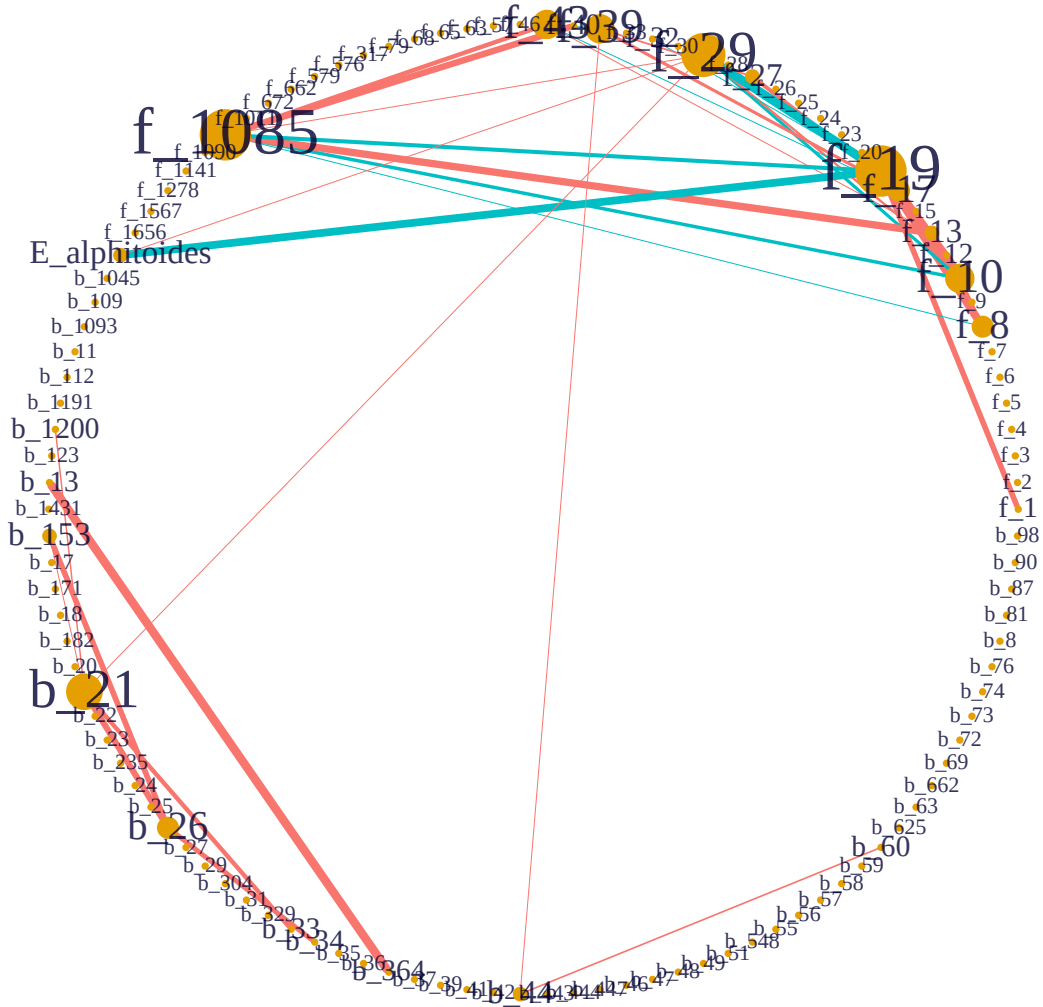
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



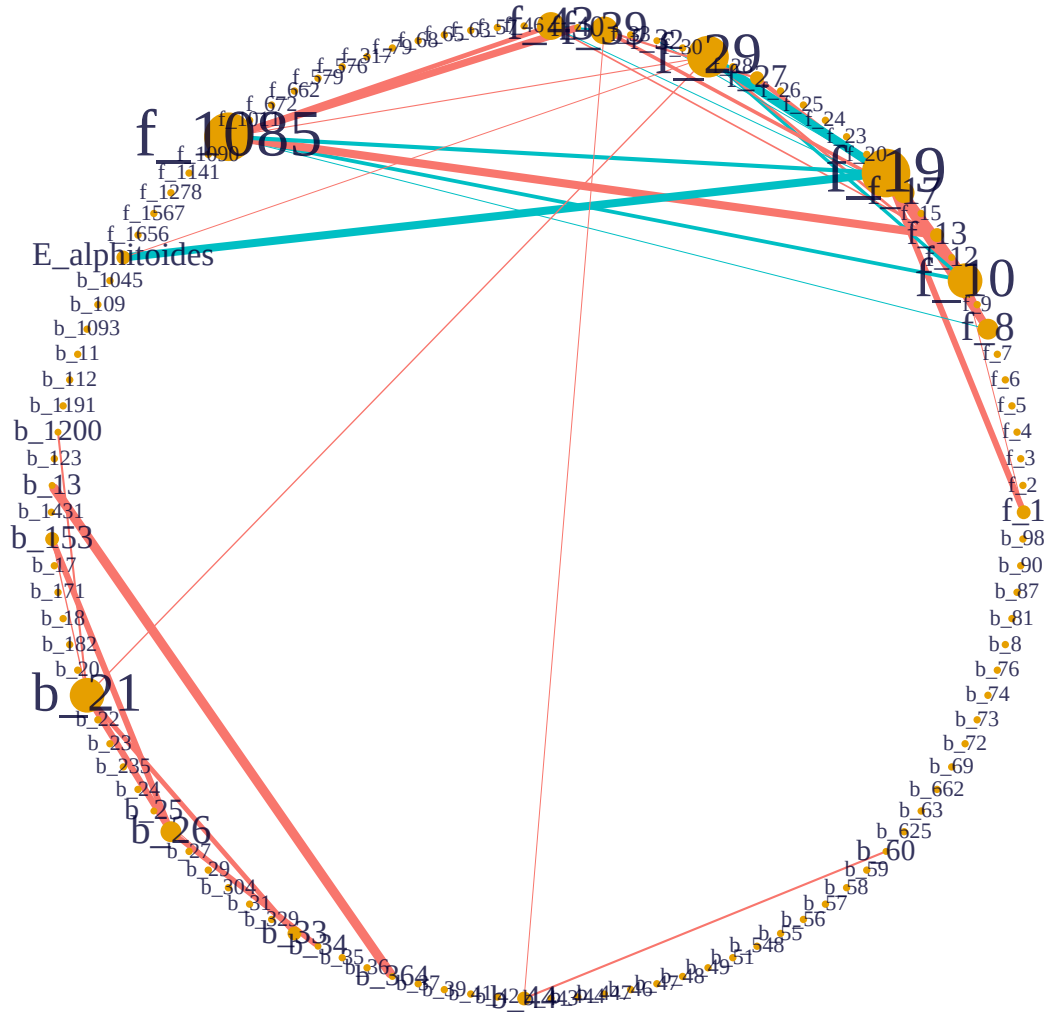
Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```



Networks of partial correlations for oak mildew pathobiome

```
# Models with offset and covariates (tree + orientation)
formula <- counts ~ 1 + covariates$tree + covariates$orientation + offset(log(offsets))
models_PLN <- PLNnetwork(formula, penalties = 10^seq(log10(2), log10(0.6), len = 30))
```