Joint network inference with the consensual LASSO

Nathalie Villa-Vialaneix

http://www.nathalievilla.org

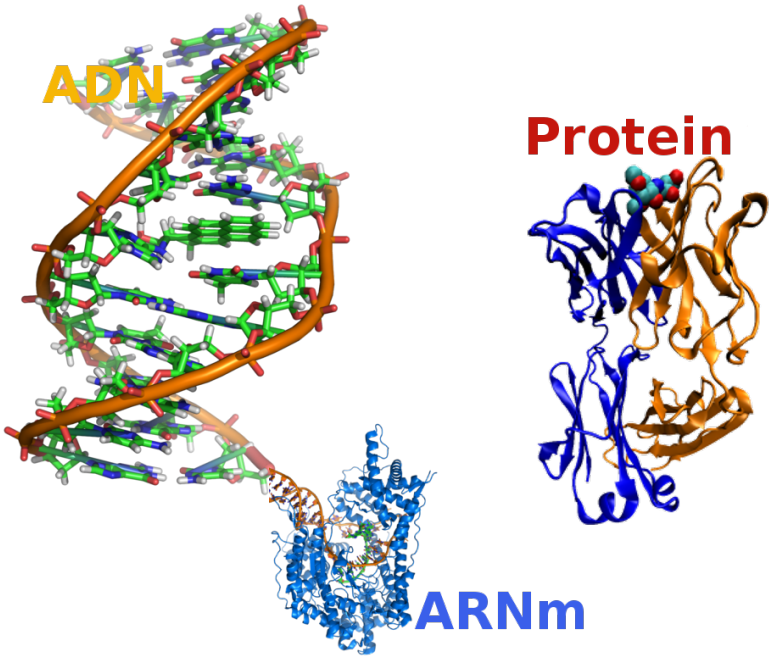
nathalie.villa@toulouse.inra.fr

2014 ENBIS-SFdS Spring Meeting
Paris, 9-11 April

Joint work with Matthieu Vignes, Nathalie Viguerie and Magali San Cristobal

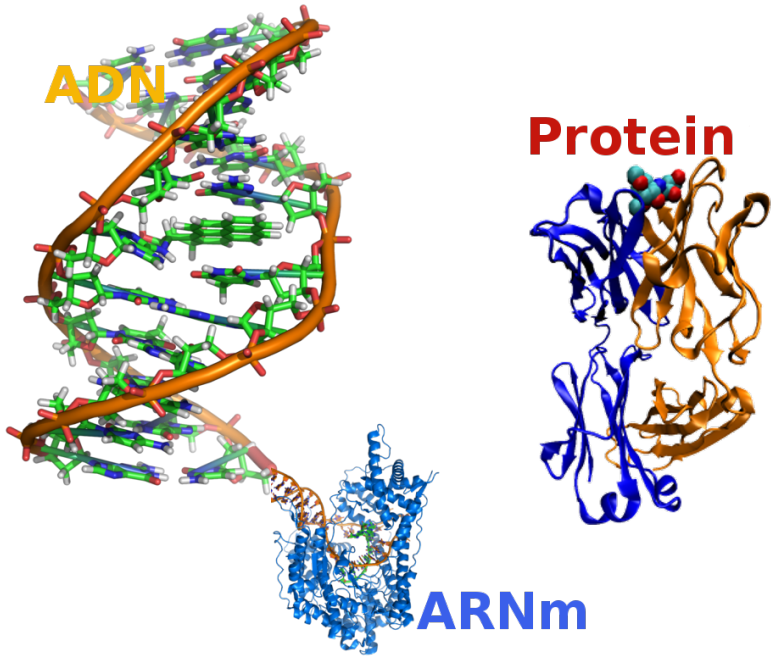
Outline

Transcriptomic data



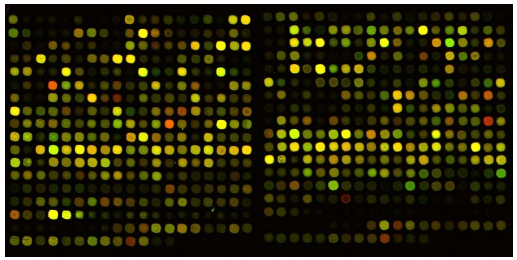
DNA transcripted into mRNA to produce proteins

Transcriptomic data

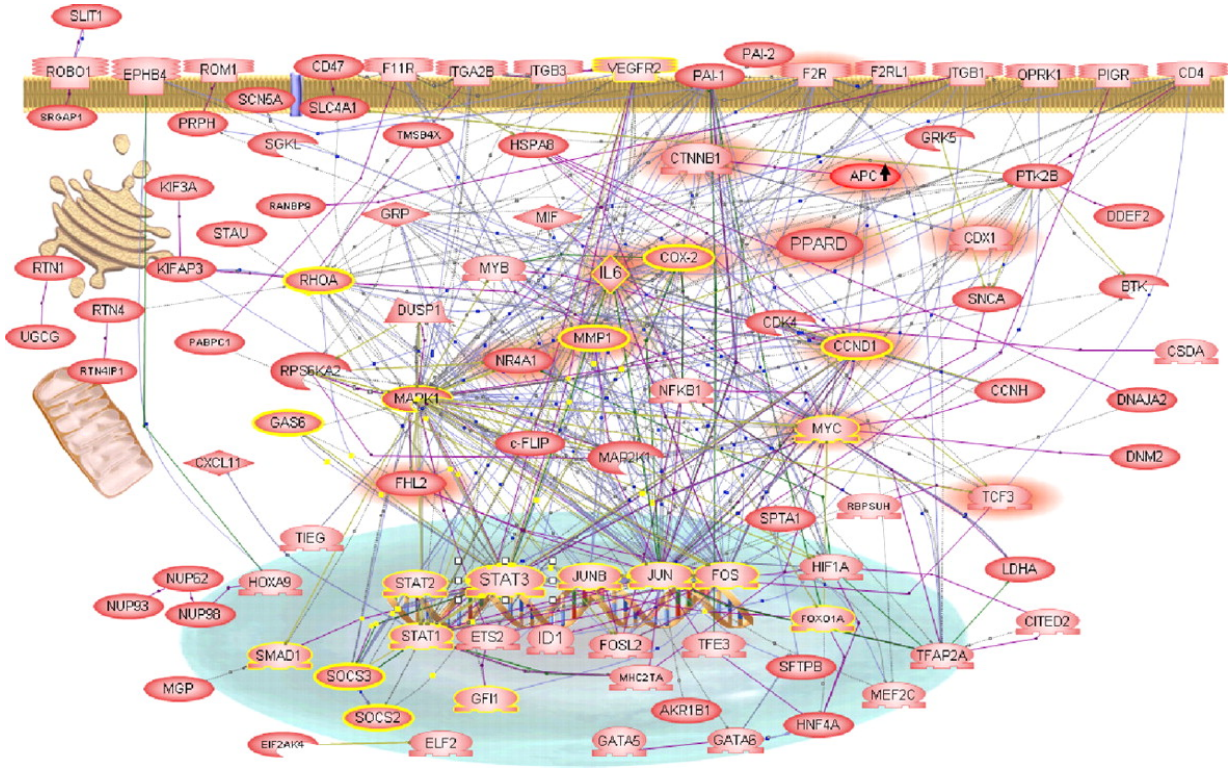


DNA transcripted into mRNA to produce proteins

transcriptomic data: measure of the quantity of mRNA corresponding to a given gene in given cells (blood, muscle...) of a living organism



Systems biology



Some genes' expressions **activate** or **repress** other genes' expressions

⇒ understanding the whole cascade helps to comprehend the global functioning of living organisms[1]

[1] Picture taken from: Abdollahi A *et al.*, *PNAS* 2007, **104**:12890-12895. © 2007 by National Academy of Sciences

Model framework

Data: large scale gene expression data

$$\underbrace{\begin{matrix} \text{individuals} \\ n \simeq 30/50 \end{matrix}}_{} \quad \underbrace{\left\{ X = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & x_i^j & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix} \right.}_{\text{variables (genes expression), } p \simeq 10^{3/4}}$$

What we want to obtain: a graph/network with

- nodes: genes;
- edges: strong links between gene expressions.

Advantages of inferring a network from large scale transcription data

1. **over raw data**: **focuses on the strongest direct relationships**: irrelevant or indirect relations are removed (more robust) and the data are easier to visualize and understand (**track transcription relations**).

Advantages of inferring a network from large scale transcription data

1. **over raw data**: **focuses on the strongest direct relationships**: irrelevant or indirect relations are removed (more robust) and the data are easier to visualize and understand (**track transcription relations**).
   Expression data are **analyzed all together** and not by pairs (**systems model**).

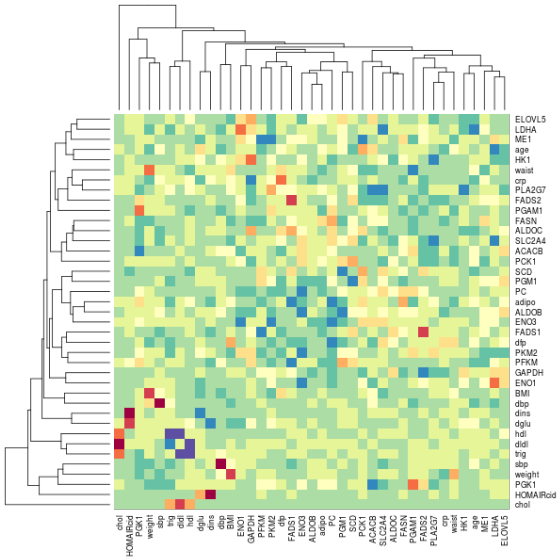
Advantages of inferring a network from large scale transcription data

1. **over raw data**: **focuses on the strongest direct relationships**: irrelevant or indirect relations are removed (more robust) and the data are easier to visualize and understand (**track transcription relations**).
   Expression data are **analyzed all together** and not by pairs (**systems model**).

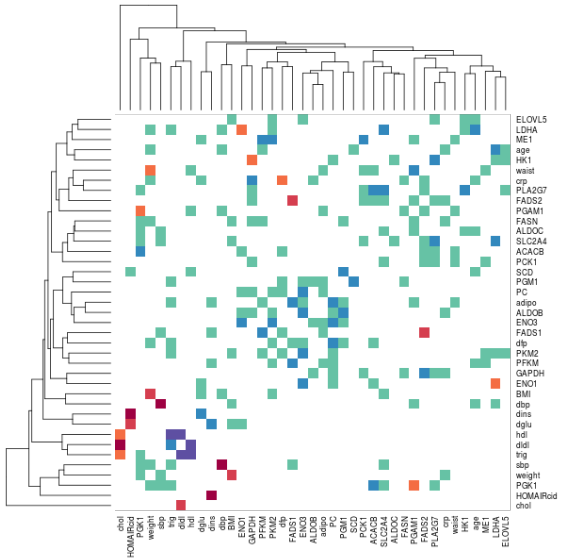
2. **over bibliographic network**: can handle **interactions with yet unknown** (not annotated) **genes** and deal with data collected in a particular condition.

Using *correlations*: relevance network [Butte and Kohane, 1999, Butte and Kohane, 2000]

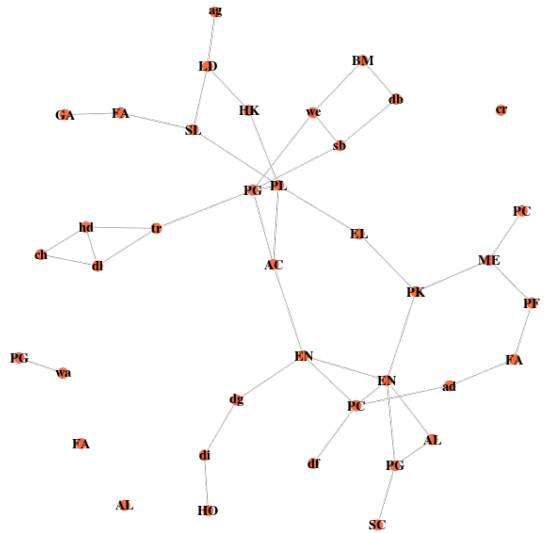
First (naive) approach: calculate correlations between expressions for all pairs of genes, threshold the smallest ones and build the network.



Correlations

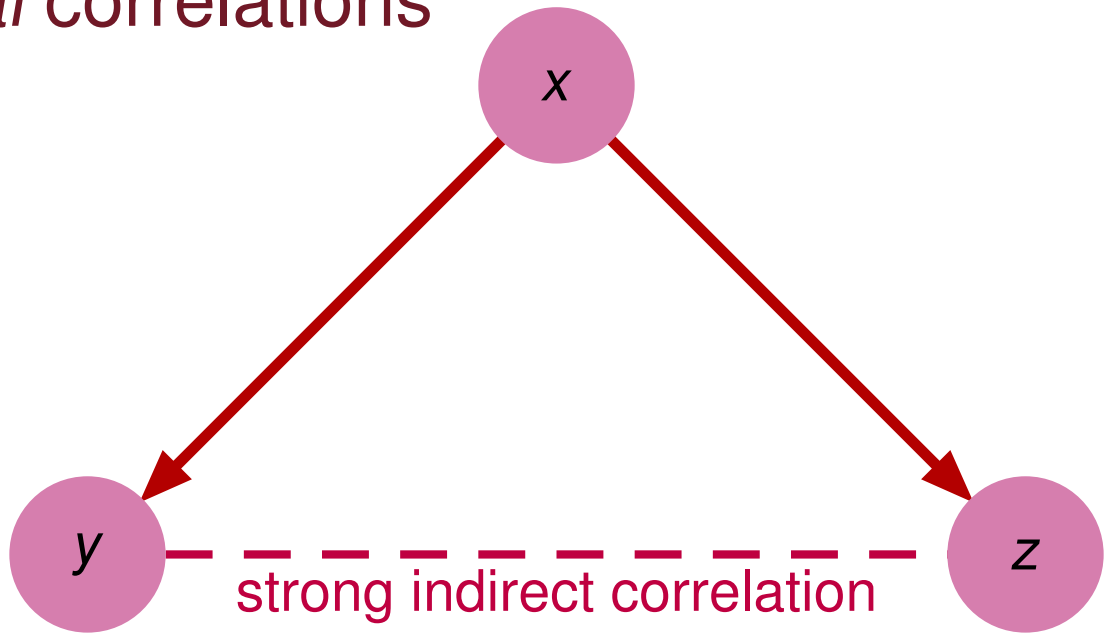


Thresholding

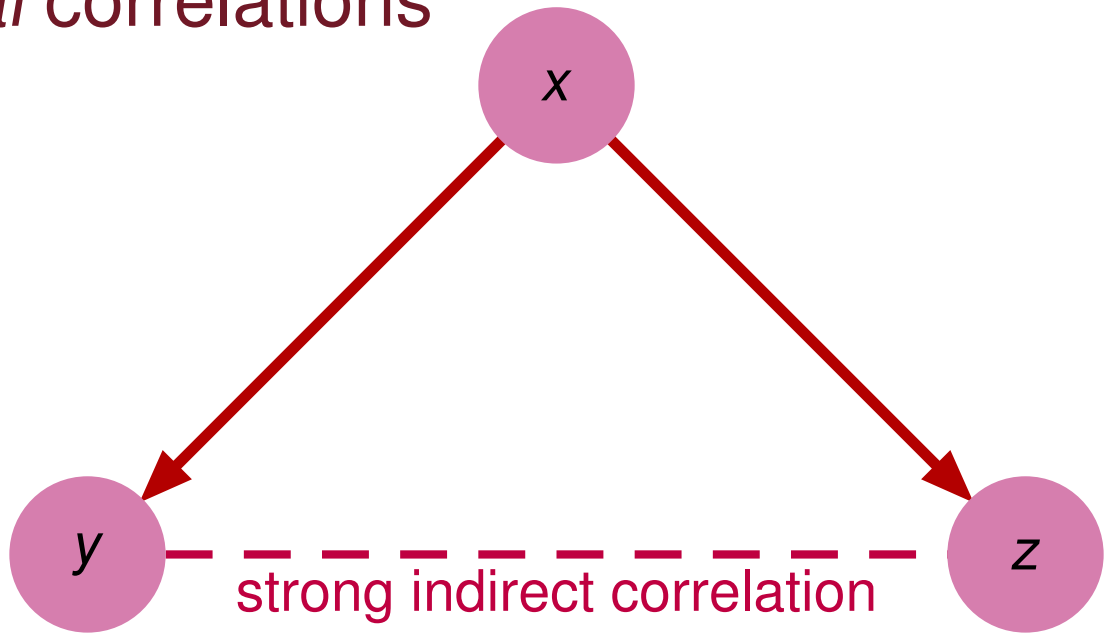


Graph

Using *partial* correlations

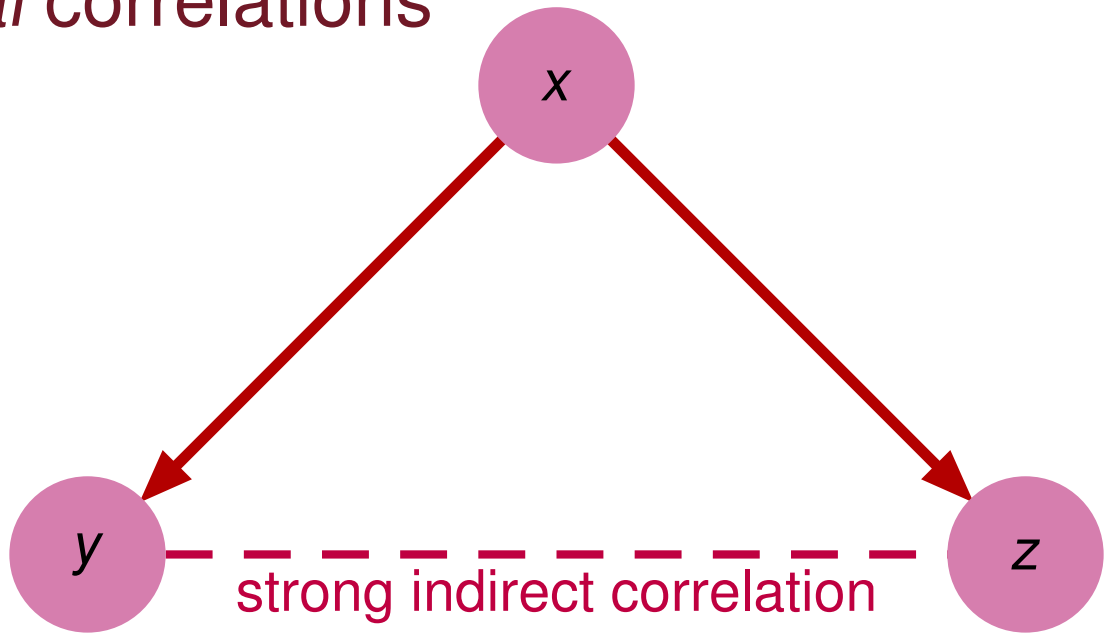


Using *partial* correlations




```
set.seed(2807); x <- rnorm(100)
y <- 2*x+1+rnorm(100,0,0.1); cor(x,y) [1] 0.998826
z <- 2*x+1+rnorm(100,0,0.1); cor(x,z) [1] 0.998751
cor(y,z) [1] 0.9971105
```

Using *partial* correlations




```
set.seed(2807); x <- rnorm(100)
y <- 2*x+1+rnorm(100,0,0.1); cor(x,y) [1] 0.998826
z <- 2*x+1+rnorm(100,0,0.1); cor(x,z) [1] 0.998751
cor(y,z) [1] 0.9971105
# Partial correlation
cor(lm(x~z)$residuals,lm(y~z)$residuals) [1] 0.7801174
cor(lm(x~y)$residuals,lm(z~y)$residuals) [1] 0.7639094
cor(lm(y~x)$residuals,lm(z~x)$residuals) [1] -0.1933699
```

Partial correlation in the Gaussian framework

$(X_i)_{i=1,\dots,n}$ are **i.i.d. Gaussian random variables** $\mathcal{N}(0,\Sigma)$ (gene expression); then

$$j \longleftrightarrow j' (\text{genes } j \text{ and } j' \text{ are linked}) \Leftrightarrow \mathbb{C}\text{or}\left(X^j, X^{j'} | (X^k)_{k\neq j,j'}\right) > 0$$

Partial correlation in the Gaussian framework

$(X_i)_{i=1,\dots,n}$ are **i.i.d. Gaussian random variables** $\mathcal{N}(0,\Sigma)$ (gene expression); then

$$j \longleftrightarrow j' (\text{genes } j \text{ and } j' \text{ are linked}) \Leftrightarrow \mathbb{C}\text{or}\left(X^j, X^{j'} | (X^k)_{k\neq j,j'}\right) > 0$$

If (**concentration matrix**) $S = \Sigma^{-1}$,

$$\mathbb{C}\mathbf{or}\left(X^j, X^{j'} | (X^k)_{k\neq j,j'}\right) = -\frac{S_{jj'}}{\sqrt{S_{jj} S_{j'j'}}}$$

$\Rightarrow$ Estimate Σ^{-1} to unravel the graph structure

Partial correlation in the Gaussian framework

$(X_i)_{i=1,\dots,n}$ are **i.i.d. Gaussian random variables** $\mathcal{N}(0,\Sigma)$ (gene expression); then

$$j \longleftrightarrow j' (\text{genes } j \text{ and } j' \text{ are linked}) \Leftrightarrow \mathbb{C}\text{or}\left(X^j, X^{j'} | (X^k)_{k\neq j,j'}\right) > 0$$

If (**concentration matrix**) $S = \Sigma^{-1}$,

$$\mathbb{C}\mathbf{or}\left(X^j, X^{j'} | (X^k)_{k\neq j,j'}\right) = -\frac{S_{jj'}}{\sqrt{S_{jj} S_{j'j'}}}$$

$\Rightarrow$ Estimate Σ^{-1} to unravel the graph structure

Problem: Σ: p-dimensional matrix and $n \ll p \Rightarrow (\widehat{\Sigma}^n)^{-1}$ is a **poor estimate** of S)!

Various approaches for inferring networks with GGM

Graphical Gaussian Model

- seminal work:
  [Schäfer and Strimmer, 2005a, Schäfer and Strimmer, 2005b]
  (with shrinkage and a proposal for a Bayesian test of significance)
  - estimate Σ^{-1} by $(\widehat{\Sigma}^n + \lambda \mathbb{I})^{-1}$
  - use a Bayesian test to test which coefficients are significantly non zero.

Various approaches for inferring networks with GGM

Graphical Gaussian Model

- seminal work:
  [Schäfer and Strimmer, 2005a, Schäfer and Strimmer, 2005b]
  (with shrinkage and a proposal for a Bayesian test of significance)
  - estimate Σ^{-1} by $(\widehat{\Sigma}^n + \lambda \mathbb{I})^{-1}$
  - use a Bayesian test to test which coefficients are significantly non zero.

$\forall\, j$, estimate the linear model:

$$X^j = \beta_j^T X^{-j} + \epsilon \qquad ; \qquad \arg\max_{(\beta_{jj'})_{j'}} (\log \mathrm{ML}_j)$$

because $\beta_{jj'} = -\frac{s_{jj'}}{s_{jj}}$.

Various approaches for inferring networks with GGM

Graphical Gaussian Model

- seminal work:
  [Schäfer and Strimmer, 2005a, Schäfer and Strimmer, 2005b]
  (with shrinkage and a proposal for a Bayesian test of significance)
  - estimate Σ^{-1} by $(\widehat{\Sigma}^n + \lambda \mathbb{I})^{-1}$
  - use a Bayesian test to test which coefficients are significantly non zero.

$\forall\, j$, estimate the linear model:

$$X^j = \beta_j^T X^{-j} + \epsilon \qquad ; \qquad \arg \min_{(\beta_{jj'})_{j'}} \sum_{i=1}^{n} \left(X_{ij} - \beta_j^T X_i^{-j} \right)^2$$

because $\beta_{jj'} = -\frac{S_{jj'}}{S_{jj}}$.

Various approaches for inferring networks with GGM

Graphical Gaussian Model

- seminal work:
  [Schäfer and Strimmer, 2005a, Schäfer and Strimmer, 2005b]
  (with shrinkage and a proposal for a Bayesian test of significance)
  - estimate Σ^{-1} by $(\widehat{\Sigma}^n + \lambda \mathbb{I})^{-1}$
  - use a Bayesian test to test which coefficients are significantly non zero.
- sparse approaches:
  [Meinshausen and Bühlmann, 2006, Friedman et al., 2008]:
  $\forall\, j$, estimate the linear model:

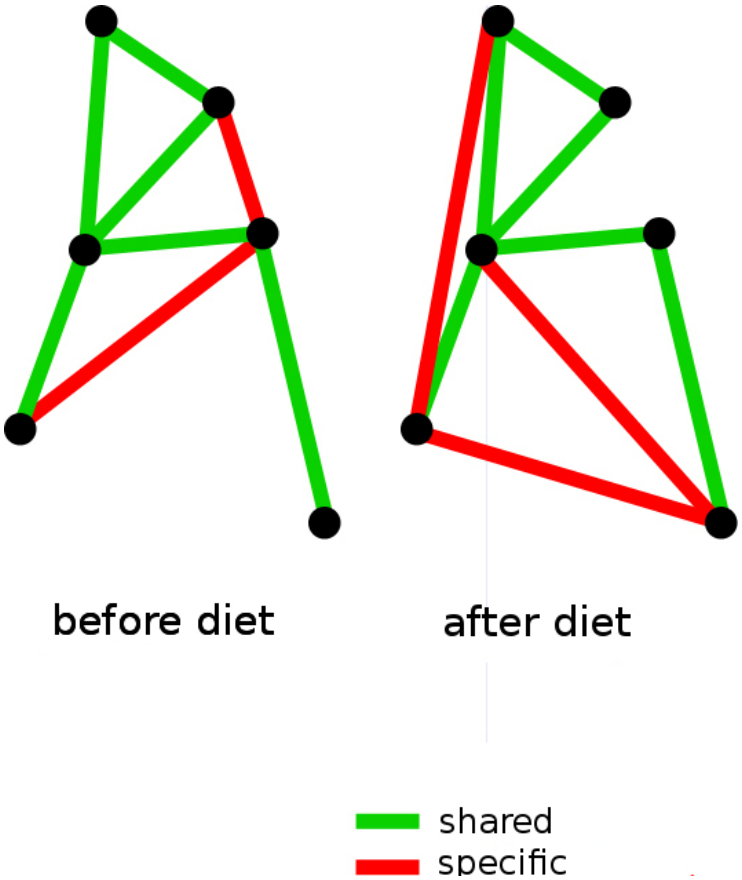
$$X^j = \beta_j^T X^{-j} + \epsilon \quad ; \quad \arg\min_{(\beta_{jj'})_{j'}} \sum_{i=1}^{n} \left(X_{ij} - \beta_j^T X_i^{-j} \right)^2 + \lambda \|\beta_j\|_{L^1}$$

with $\|\beta_j\|_{L^1} = \sum_{j'} |\beta_{jj'}|$
L^1 penalty yields to $\beta_{jj'} = 0$ for most j' (**variable selection**)

Outline

Motivation for multiple networks inference

Pan-European project Diogenes[2] (with Nathalie Viguerie, INSERM): gene expressions (lipid tissues) from 204 obese women **before** and **after** a low-calorie diet (LCD).



- **Assumption**: A common functioning exists regardless the condition;
- Which genes are linked **independently from/depending on** the condition?

[2] http://www.diogenes-eu.org/; see also **[Viguerie et al., 2012]**

Naive approach: independent estimations

Notations: p genes measured in k samples, each corresponding to a specific condition: $(X_j^c)_{j=1,\dots,p} \sim \mathcal{N}(0, \Sigma^c)$, for $c = 1, \dots, k$.
For $c = 1, \dots, k$, n_c independent observations $(X_{ij}^c)_{i=1,\dots,n_c}$ and $\sum_c n_c = n$.

Independent inference

Estimation $\forall\, c = 1, \dots, k$ and $\forall\, j = 1, \dots, p$,

$$X_j^c = \mathbf{X}_{\setminus j}^c \beta_j^c + \epsilon_j^c$$

are estimated (independently) by maximizing pseudo-likelihood:

$$\mathcal{L}(S|\mathbf{X}) = \sum_{c=1}^{k} \sum_{j=1}^{p} \sum_{i=1}^{n_c} \log \mathbb{P}\left(X_{ij}^c | \mathbf{X}_{i,\setminus j}^c,\, S_j^c\right)$$

Related papers

Problem: previous estimation does not use the fact that the different networks should be somehow alike!

Previous proposals

- **[Chiquet et al., 2011]** replace Σ^c by $\widetilde{\Sigma}^c = \frac{1}{2}\Sigma^c + \frac{1}{2}\overline{\Sigma}$ and add a sparse penalty;

Related papers

Problem: previous estimation does not use the fact that the different networks should be somehow alike!

Previous proposals

- **[Chiquet et al., 2011]** replace Σ^c by $\widetilde{\Sigma}^c = \frac{1}{2}\Sigma^c + \frac{1}{2}\overline{\Sigma}$ and add a sparse penalty;
- **[Chiquet et al., 2011]** LASSO and Group-LASSO type penalties to force identical or sign-coherent edges between conditions:

$$\sum_{jj'} \sqrt{\sum_c (S^c_{jj'})^2} \qquad \text{or} \quad \sum_{jj'} \left[\sqrt{\sum_c (S^c_{jj'})^2_+} + \sqrt{\sum_c (S^c_{jj'})^2_-} \right]$$

$\Rightarrow S^c_{jj'} = 0 \; \forall \, c$ for most entries OR $S^c_{jj'}$ can only be of a given sign (positive or negative) whatever c

Related papers

Problem: previous estimation does not use the fact that the different networks should be somehow alike!

Previous proposals

- **[Chiquet et al., 2011]** replace Σ^c by $\widetilde{\Sigma}^c = \frac{1}{2}\Sigma^c + \frac{1}{2}\overline{\Sigma}$ and add a sparse penalty;
- **[Chiquet et al., 2011]** LASSO and Group-LASSO type penalties to force identical or sign-coherent edges between conditions
- **[Danaher et al., 2013]** add the penalty $\sum_{c \neq c'} \|S^c - S^{c'}\|_{L^1} \Rightarrow$ very strong consistency between conditions (sparse penalty over the inferred networks identical values for concentration matrix entries);

Related papers

Problem: previous estimation does not use the fact that the different networks should be somehow alike!

Previous proposals

- **[Chiquet et al., 2011]** replace Σ^c by $\widetilde{\Sigma}^c = \frac{1}{2}\Sigma^c + \frac{1}{2}\overline{\Sigma}$ and add a sparse penalty;
- **[Chiquet et al., 2011]** LASSO and Group-LASSO type penalties to force identical or sign-coherent edges between conditions
- **[Danaher et al., 2013]** add the penalty $\sum_{c \neq c'} \|S^c - S^{c'}\|_{L^1} \Rightarrow$ very strong consistency between conditions (sparse penalty over the inferred networks identical values for concentration matrix entries);
- **[Mohan et al., 2012]** add a group-LASSO like penalty $\sum_{c \neq c'} \sum_j \|S_j^c - S_j^{c'}\|_{L^2}$ that focuses on differences due to a few number of **nodes** only.

Consensus LASSO

Proposal

Infer multiple networks by forcing them toward a consensual network: i.e., explicitly **keeping the differences** between conditions under control but **with a L^2 penalty** (allow for more differences than Group-LASSO type penalties).

Original optimization:

$$\max_{(\beta_{jk}^c)_{k\neq j, c=1,\ldots,C}} \sum_c \left(\log \mathrm{ML}_j^c - \lambda \sum_{k\neq j} |\beta_{jk}^c| \right).$$

Consensus LASSO

Proposal

Infer multiple networks by forcing them toward a consensual network: i.e., explicitly **keeping the differences** between conditions under control but **with a L^2 penalty** (allow for more differences than Group-LASSO type penalties).

Original optimization:

$$\max_{(\beta_{jk}^c)_{k\neq j, c=1,\ldots,C}} \sum_c \left(\log \mathrm{ML}_j^c - \lambda \sum_{k\neq j} |\beta_{jk}^c| \right).$$

[Ambroise et al., 2009, Chiquet et al., 2011]: is equivalent to minimize p problems having dimension $k(p-1)$:

$$\frac{1}{2}\beta_j^T \widehat{\Sigma}_{\backslash j \backslash j} \beta_j + \beta_j^T \widehat{\Sigma}_{j \backslash j} + \lambda \|\beta_j\|_{L^1}, \qquad \beta_j = (\beta_j^1, \ldots, \beta_j^k)$$

with $\widehat{\Sigma}_{\backslash j \backslash j}$: block diagonal matrix $\mathbb{D}\mathrm{iag}\left(\widehat{\Sigma}^1_{\backslash j \backslash j}, \ldots, \widehat{\Sigma}^k_{\backslash j \backslash j}\right)$ and similarly for $\widehat{\Sigma}_{j \backslash j}$.

Consensus LASSO

Proposal

Infer multiple networks by forcing them toward a consensual network: i.e., explicitly **keeping the differences** between conditions under control but **with a L^2 penalty** (allow for more differences than Group-LASSO type penalties).

Add a constraint to force inference toward a "consensus" β^{cons}

$$\frac{1}{2}\beta_j^T \widehat{\Sigma}_{\backslash j \backslash j} \beta_j + \beta_j^T \widehat{\Sigma}_{j \backslash j} + \lambda \|\beta_j\|_{L^1} + \mu \sum_c w_c \|\beta_j^c - \beta_j^{\text{cons}}\|_{L^2}^2$$

with:

- w_c: real number used to weight the conditions ($w_c = 1$ or $w_c = \frac{1}{\sqrt{n_c}}$);
- μ regularization parameter;
- β_j^{cons} whatever you want...?

Choice of a consensus: set one...

Typical case:

- a prior network is known (e.g., from **bibliography**);
- with no prior information, use a fixed prior corresponding to (e.g.) **global inference**

$\Rightarrow$ given (and fixed) β^{cons}

Choice of a consensus: set one...

Typical case:

- a prior network is known (e.g., from **bibliography**);
- with no prior information, use a fixed prior corresponding to (e.g.) **global inference**

$\Rightarrow$ given (and fixed) β^{cons}

Proposition

Using a fixed β_j^{cons}, the optimization problem is equivalent to minimizing the p following standard quadratic problem in $\mathbb{R}^{k(p-1)}$ with L_1-penalty:

$$\frac{1}{2}\beta_j^T B^1(\mu)\beta_j + \beta_j^T B^2(\mu) + \lambda||\beta_j||_{L^1},$$

where

- $B^1(\mu) = \widehat{\Sigma}_{\backslash j \backslash j} + 2\mu\mathbb{I}_{k(p-1)}$, with $\mathbb{I}_{k(p-1)}$ the $k(p-1)$-identity matrix
- $B^2(\mu) = \widehat{\Sigma}_{j \backslash j} - 2\mu\mathbb{I}_{k(p-1)}\beta^{\text{cons}}$ with $\beta^{\text{cons}} = \left((\beta_j^{\text{cons}})^T, \ldots, (\beta_j^{\text{cons}})^T\right)^T$

Choice of a consensus: adapt one during training...

Derive the consensus from the condition-specific estimates:

$$\beta_j^{\text{cons}} = \sum_c \frac{n_c}{n} \beta_j^c$$

Choice of a consensus: adapt one during training...

Derive the consensus from the condition-specific estimates:

$$\beta_j^{\text{cons}} = \sum_c \frac{n_c}{n} \beta_j^c$$

Proposition

Using $\beta_j^{\text{cons}} = \sum_{c=1}^{k} \frac{n_c}{n} \beta_j^c$, the optimization problem is equivalent to minimizing the following standard quadratic problem with L_1-penalty:

$$\frac{1}{2} \beta_j^T S_j(\mu) \beta_j + \beta_j^T \widehat{\Sigma}_{j\backslash j} + \lambda ||\beta_j||_{L^1}$$

where $S_j(\mu) = \widehat{\Sigma}_{\backslash j \backslash j} + 2\mu A^T(\mu) A(\mu)$ where $A(\mu)$ is a $[k(p-1) \times k(p-1)]$-matrix that does not depend on j.

Computational aspects: optimization

Common framework

Objective function can be decomposed into:

convex part $C(\beta_j) = \frac{1}{2}\beta_j^T Q_j^1(\mu) + \beta_j^T Q_j^2(\mu)$

L^1-norm penalty $\mathcal{P}(\beta_j) = \|\beta_j\|_{L^1}$

Computational aspects: optimization

Common framework

Objective function can be decomposed into:

convex part $C(\beta_j) = \frac{1}{2}\beta_j^T Q_j^1(\mu) + \beta_j^T Q_j^2(\mu)$

L^1-norm penalty $\mathcal{P}(\beta_j) = \|\beta_j\|_{L^1}$

optimization by "active set" [Osborne et al., 2000, Chiquet et al., 2011]

1: **repeat**(λ given)

2: Given $\mathcal{A}$ and $\beta_{jj'}$ st: $\beta_{jj'} \neq 0$, $\forall\, j' \in \mathcal{A}$, solve (over h) the **smooth minimization problem** restricted to $\mathcal{A}$

$$C(\beta_j + h) + \lambda\mathcal{P}(\beta_j + h) \qquad \Rightarrow \qquad \beta_j \leftarrow \beta_j + h$$

4: **until**

Computational aspects: optimization

Common framework

Objective function can be decomposed into:

convex part $C(\beta_j) = \frac{1}{2}\beta_j^T Q_j^1(\mu) + \beta_j^T Q_j^2(\mu)$

L^1-norm penalty $\mathcal{P}(\beta_j) = \|\beta_j\|_{L^1}$

optimization by "active set" [Osborne et al., 2000, Chiquet et al., 2011]

1: **repeat**(λ given)

2: Given $\mathcal{A}$ and $\beta_{jj'}$ st: $\beta_{jj'} \neq 0$, $\forall\, j' \in \mathcal{A}$, solve (over h) the **smooth minimization problem** restricted to $\mathcal{A}$

$$C(\beta_j + h) + \lambda\mathcal{P}(\beta_j + h) \qquad \Rightarrow \qquad \beta_j \leftarrow \beta_j + h$$

3: Update $\mathcal{A}$ by adding most violating variables, i.e., variables st:

$$\text{abs}\left|\partial C(\beta_j) + \lambda\partial\mathcal{P}(\beta_j)\right| > 0$$

with $[\partial\mathcal{P}(\beta_j)]_{j'} \in [-1, 1]$ if $j' \notin \mathcal{A}$

4: **until**

Computational aspects: optimization

Common framework

Objective function can be decomposed into:

convex part $C(\beta_j) = \frac{1}{2}\beta_j^T Q_j^1(\mu) + \beta_j^T Q_j^2(\mu)$

L^1-norm penalty $\mathcal{P}(\beta_j) = \|\beta_j\|_{L^1}$

optimization by "active set" [Osborne et al., 2000, Chiquet et al., 2011]

1: **repeat**(λ given)

2: Given $\mathcal{A}$ and $\beta_{jj'}$ st: $\beta_{jj'} \neq 0$, $\forall\, j' \in \mathcal{A}$, solve (over h) the **smooth minimization problem** restricted to $\mathcal{A}$

$$C(\beta_j + h) + \lambda\mathcal{P}(\beta_j + h) \qquad \Rightarrow \qquad \beta_j \leftarrow \beta_j + h$$

3: Update $\mathcal{A}$ by adding most violating variables, i.e., variables st:

$$\text{abs}\left|\partial C(\beta_j) + \lambda\partial\mathcal{P}(\beta_j)\right| > 0$$

with $[\partial\mathcal{P}(\beta_j)]_{j'} \in [-1, 1]$ if $j' \notin \mathcal{A}$

4: **until all conditions satisfied** i.e. $\text{abs}\left|\partial C(\beta_j) + \lambda\partial\mathcal{P}(\beta_j)\right| > 0$

Computational aspects: optimization

Common framework

Objective function can be decomposed into:

convex part $C(\beta_j) = \frac{1}{2}\beta_j^T Q_j^1(\mu) + \beta_j^T Q_j^2(\mu)$

L^1-norm penalty $\mathcal{P}(\beta_j) = \|\beta_j\|_{L^1}$

optimization by "active set" [Osborne et al., 2000, Chiquet et al., 2011]

1: **repeat**(λ given)

2: Given $\mathcal{A}$ and $\beta_{jj'}$ st: $\beta_{jj'} \neq 0$, $\forall\, j' \in \mathcal{A}$, solve (over h) the **smooth minimization problem** restricted to $\mathcal{A}$

$$C(\beta_j + h) + \lambda\mathcal{P}(\beta_j + h) \quad \Rightarrow \quad \beta_j \leftarrow \beta_j + h$$

3: Update $\mathcal{A}$ by adding most violating variables, i.e., variables st:

$$\text{abs}\left|\partial C(\beta_j) + \lambda\partial\mathcal{P}(\beta_j)\right| > 0$$

with $[\partial\mathcal{P}(\beta_j)]_{j'} \in [-1, 1]$ if $j' \notin \mathcal{A}$

4: **until all conditions satisfied** i.e. $\text{abs}\left|\partial C(\beta_j) + \lambda\partial\mathcal{P}(\beta_j)\right| > 0$

Repeat:

large λ

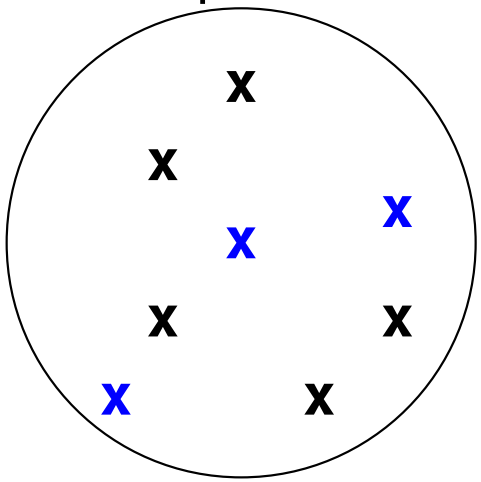
↓

small λ

using previous β as prior

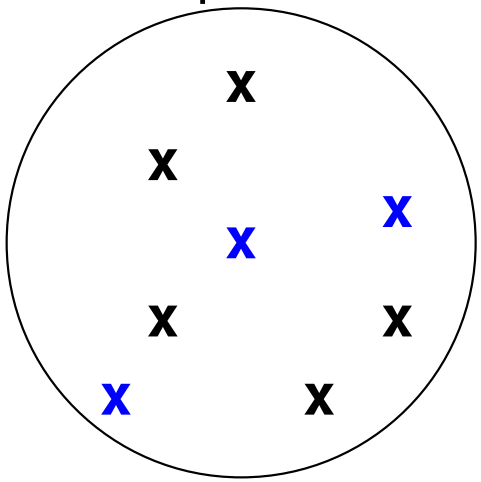
Bootstrap estimation $\simeq$ BOLASSO [Bach, 2008]

subsample n observations with replacement



Bootstrap estimation $\simeq$ BOLASSO [Bach, 2008]

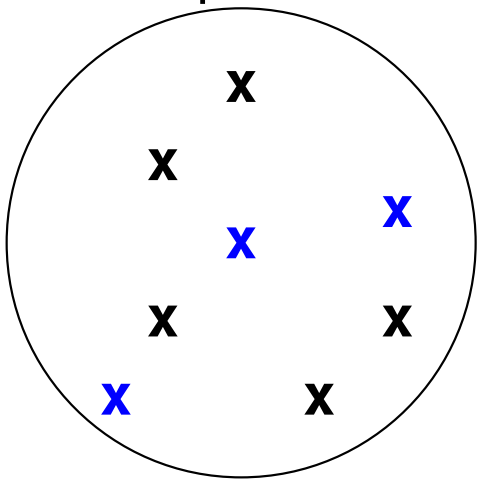
subsample n observations with replacement



$\xrightarrow{\text{cLasso estimation}}$ for varying λ, $(\beta_{jj'}^{\lambda,b})_{jj'}$

Bootstrap estimation $\simeq$ BOLASSO [Bach, 2008]

subsample n observations with replacement



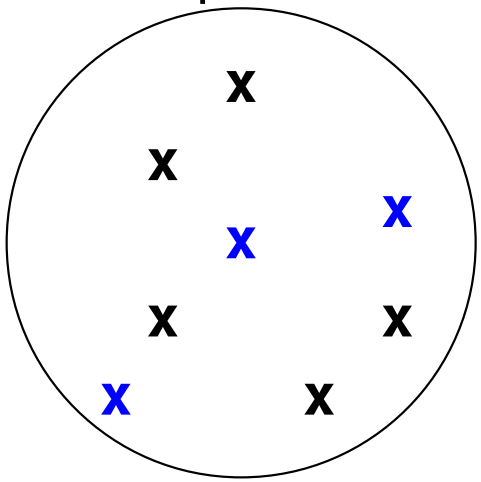
$\xrightarrow{\text{cLasso estimation}}$ for varying λ, $(\beta_{jj'}^{\lambda,b})_{jj'}$

$\downarrow$ threshold

keep the first T_1 largest coefficients

Bootstrap estimation $\simeq$ BOLASSO [Bach, 2008]

$B\times$ subsample n observations with replacement



$\xrightarrow{\text{cLasso estimation}}$ for varying λ, $(\beta_{jj'}^{\lambda,b})_{jj'}$

$\downarrow$ threshold

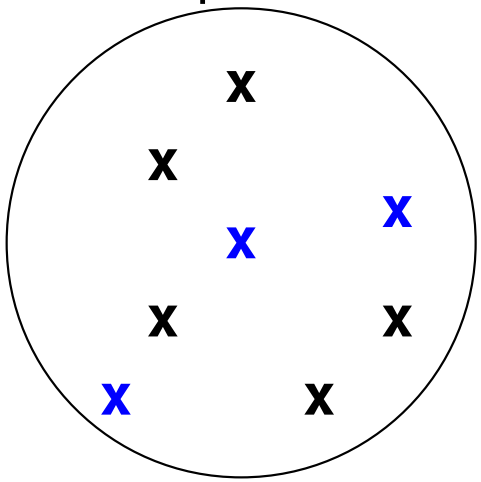
keep the first T_1 largest coefficients

Frequency table

$(1,2)$	$(1,3)$	...	(j,j')	...
130	25	...	120	...

Bootstrap estimation $\simeq$ BOLASSO [Bach, 2008]

$B\times$ subsample n observations with replacement



$\xrightarrow{\text{cLasso estimation}}$ for varying λ, $(\beta_{jj'}^{\lambda,b})_{jj'}$

$\downarrow$ threshold

keep the first T_1 largest coefficients

Frequency table

(1,2)	(1,3)	...	(j,j')	...
130	25	...	120	...

$\longrightarrow$ Keep the T_2 most frequent pairs

Outline

Simulated data

Expression data with known co-expression network

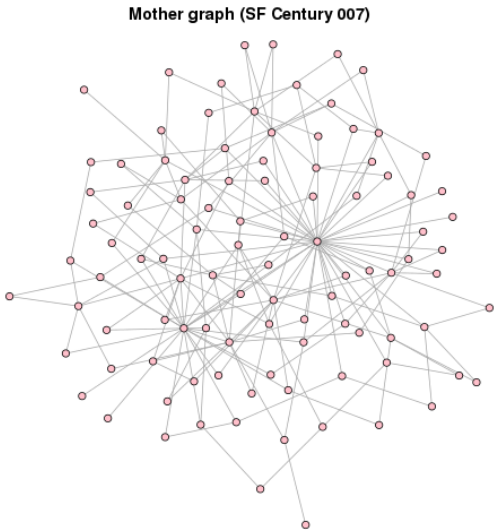
- original network (scale free) taken from `http://www.comp-sys-bio.org/AGN/data.html` (100 nodes, $\sim$ 200 edges, loops removed);
- rewire a ratio r of the edges to generate k "children" networks (sharing approximately $100(1 - 2r)$% of their edges);

Simulated data

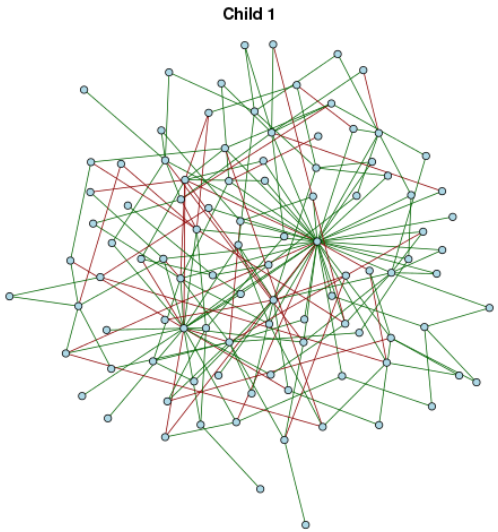
Expression data with known co-expression network

- original network (scale free) taken from http://www.comp-sys-bio.org/AGN/data.html (100 nodes, $\sim$ 200 edges, loops removed);
- rewire a ratio r of the edges to generate k "children" networks (sharing approximately $100(1-2r)\%$ of their edges);
- generate "expression data" with a random Gaussian process from each chid:
  - use the Laplacian of the graph to generate a putative concentration matrix;
  - use edge colors in the original network to set the edge sign;
  - correct the obtained matrix to make it positive;
  - invert to obtain a covariance matrix...;
  - ... which is used in a random Gaussian process to generate expression data (with noise).

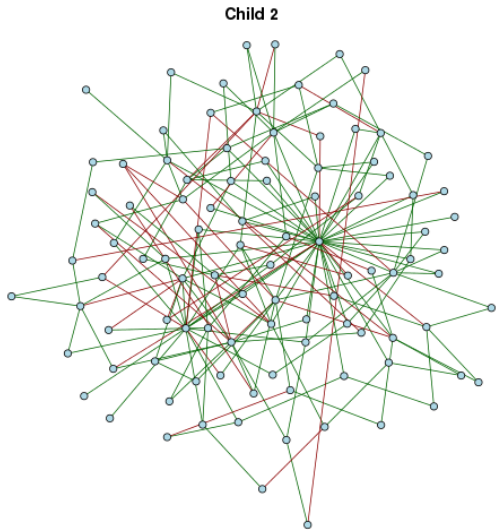
An example with $k = 2$, $r = 5\%$



mother network[3]



first child



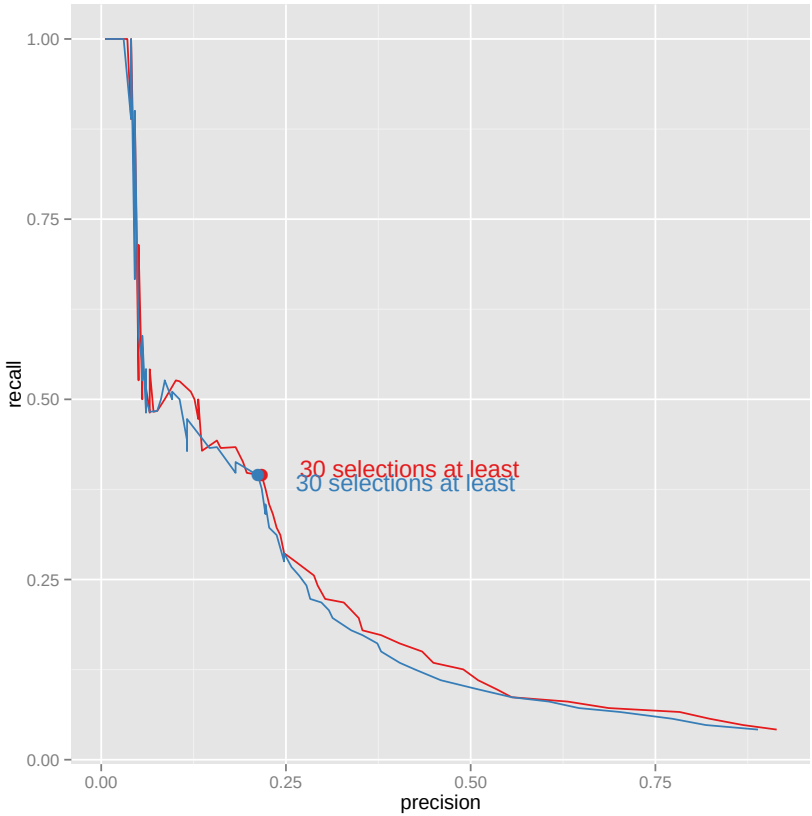
second child

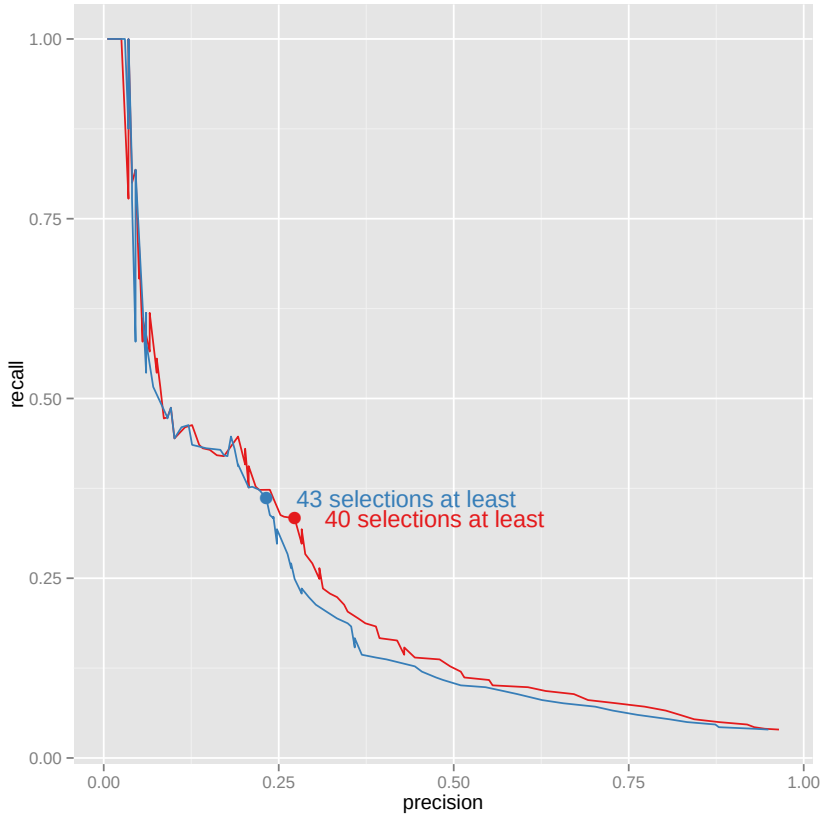
[3] actually the **parent network**. My co-author wisely noted that the mistake was unforgivable for a feminist...

Choice for T_2

Data: $r = 0.05$, $k = 2$ and $n_1 = n_2 = 20$

100 bootstrap samples, $\mu = 1$, $T_1 = 250$ **or** 500





Dots correspond to best $F = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$

$\Rightarrow$ Best F corresponds to selecting a number of edges approximately equal to the number of edges in the original network.

Choice for T_1 and μ

	μ 0.1/1	T_1 {250, 300, 500}	% of improvement of bootstrapping
network sizes		**rewired edges: 5%**	
20-20	1	500	30.69
20-30	0.1	500	11.87
30-30	1	300	20.15
50-50	1	300	14.36
20-20-20-20-20	1	500	86.04
30-30-30-30	0.1	500	42.67
network sizes		**rewired edges: 20%**	
20-20	0.1	300	-17.86
20-30	0.1	300	-18.35
30-30	1	500	-7.97
50-50	0.1	300	-7.83
20-20-20-20-20	0.1	500	10.27
30-30-30-30	1	500	13.48

Comparison with other approaches

Method compared (direct and bootstrap approaches)

- independant Graphical LASSO estimation **gLasso**
- methods implementated in the R package **simone** and described in **[Chiquet et al., 2011]**: intertwinned LASSO **iLasso**, cooperative LASSO **coopLasso** and group LASSO **groupLasso**
- fused graphical LASSO as described in **[Danaher et al., 2013]** as implemented in the R package **fgLasso**

Comparison with other approaches

Method compared (direct and bootstrap approaches)

- independant Graphical LASSO estimation **gLasso**
- methods implementated in the R package **simone** and described in **[Chiquet et al., 2011]**: intertwinned LASSO **iLasso**, cooperative LASSO **coopLasso** and group LASSO **groupLasso**
- fused graphical LASSO as described in **[Danaher et al., 2013]** as implemented in the R package **fgLasso**
- consensus Lasso with
  - fixed prior (the mother network) **cLasso(p)**
  - fixed prior (average over the conditions of independant estimations) **cLasso(2)**
  - adaptative estimation of the prior **cLasso(m)**

Comparison with other approaches

Method compared (direct and bootstrap approaches)

- independant Graphical LASSO estimation **gLasso**
- methods implementated in the R package **simone** and described in **[Chiquet et al., 2011]**: intertwinned LASSO **iLasso**, cooperative LASSO **coopLasso** and group LASSO **groupLasso**
- fused graphical LASSO as described in **[Danaher et al., 2013]** as implemented in the R package **fgLasso**
- consensus Lasso with
  - fixed prior (the mother network) **cLasso(p)**
  - fixed prior (average over the conditions of independant estimations) **cLasso(2)**
  - adaptative estimation of the prior **cLasso(m)**

Parameters set to: $T_1 = 500$, $B = 100$, $\mu = 1$

Selected results (best F)

rewired edges: 5% - **conditions**: 2 - **sample size**: 2×30

direct version

Method	**gLasso**	**iLasso**	**groupLasso**	**coopLasso**
	0.28	0.35	0.32	0.35
Method	**fgLasso**	**cLasso(m)**	**cLasso(p)**	**cLasso(2)**
	0.32	0.31	0.86	0.30

bootstrap version

Method	**gLasso**	**iLasso**	**groupLasso**	**coopLasso**
	0.31	0.34	0.36	0.34
Method	**fgLasso**	**cLasso(m)**	**cLasso(p)**	**cLasso(2)**
	0.36	0.37	0.86	0.35

Conclusions

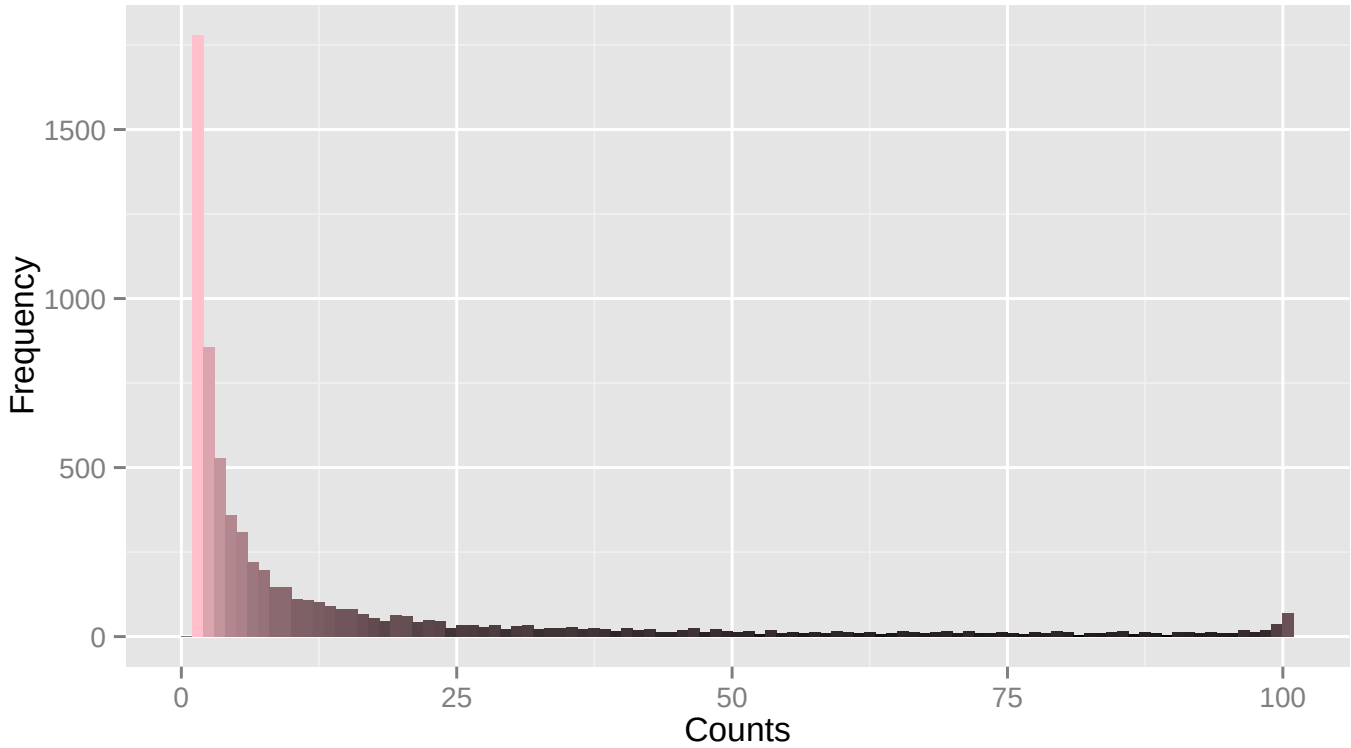
- bootstraping improves results (except for **iLasso** and for large r)
- joint inference improves results
- using a good prior is (as expected) very efficient
- adaptive approch for **cLasso** is better than naive approach

Real data

204 obese women ; expression of 221 genes before and after a LCD

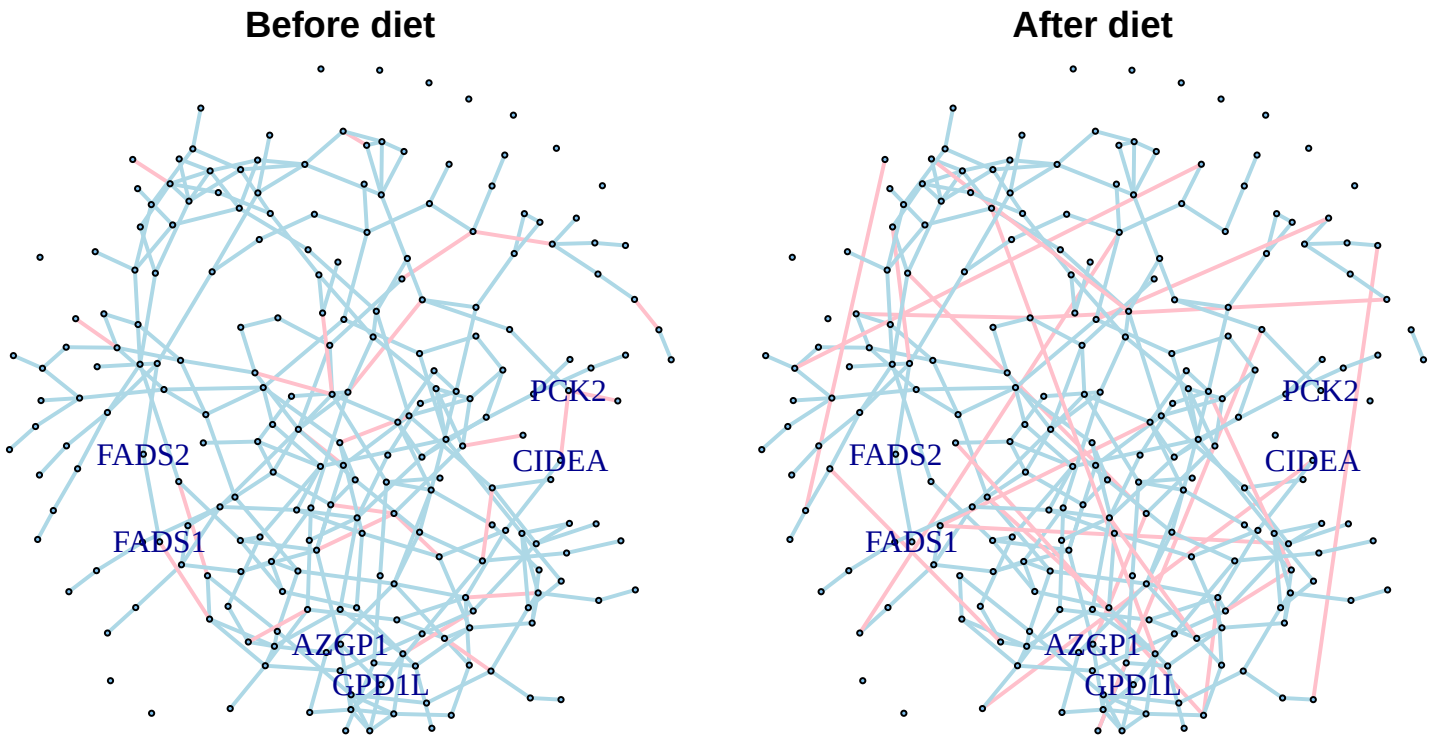
$\mu = 1$; $T_1 = 1000$ (target density: 4%)

Distribution of the number of times an edge is selected over 100 bootstrap samples



(70% of the pairs of nodes are never selected) $\Rightarrow T_2 = 80$

Networks



densities about 1.3% - some interactions (both shared and specific) make sense to the biologist

Thank you for your attention...

Programs available in the R package **therese** (on R-Forge)[4]. Joint work with

Magali SanCristobal
(GenPhySe, INRA Toulouse)

Matthieu Vignes
(MIAT, INRA Toulouse)

Nathalie Viguerie
(I2MC, INSERM Toulouse)

[4] https://r-forge.r-project.org/projects/therese-pkg

Questions?

References

Ambroise, C., Chiquet, J., and Matias, C. (2009).

Inferring sparse Gaussian graphical models with latent structure.
Electronic Journal of Statistics, 3:205–238.

Bach, F. (2008).

Bolasso: model consistent lasso estimation through the bootstrap.
In *Proceedings of the Twenty-fifth International Conference on Machine Learning (ICML)*.

Butte, A. and Kohane, I. (1999).

Unsupervised knowledge discovery in medical databases using relevance networks.
In *Proceedings of the AMIA Symposium*, pages 711–715.

Butte, A. and Kohane, I. (2000).

Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements.
In *Proceedings of the Pacific Symposium on Biocomputing*, pages 418–429.

Chiquet, J., Grandvalet, Y., and Ambroise, C. (2011).

Inferring multiple graphical structures.
Statistics and Computing, 21(4):537–553.

Danaher, P., Wang, P., and Witten, D. (2013).

The joint graphical lasso for inverse covariance estimation accross multiple classes.
Journal of the Royal Statistical Society Series B.
Forthcoming.

Friedman, J., Hastie, T., and Tibshirani, R. (2008).

Sparse inverse covariance estimation with the graphical lasso.
Biostatistics, 9(3):432–441.

Meinshausen, N. and Bühlmann, P. (2006).

High dimensional graphs and variable selection with the lasso.
Annals of Statistic, 34(3):1436–1462.

Mohan, K., Chung, J., Han, S., Witten, D., Lee, S., and Fazel, M. (2012).
Structured learning of Gaussian graphical models.
In *Proceedings of NIPS (Neural Information Processing Systems) 2012*, Lake Tahoe, Nevada, USA.

Osborne, M., Presnell, B., and Turlach, B. (2000).
On the LASSO and its dual.
Journal of Computational and Graphical Statistics, 9(2):319–337.

Schäfer, J. and Strimmer, K. (2005a).
An empirical bayes approach to inferring large-scale gene association networks.
Bioinformatics, 21(6):754–764.

Schäfer, J. and Strimmer, K. (2005b).
A shrinkage approach to large-scale covariance matrix estimation and implication for functional genomics.
Statistical Applications in Genetics and Molecular Biology, 4:1–32.

Viguerie, N., Montastier, E., Maoret, J., Roussel, B., Combes, M., Valle, C., Villa-Vialaneix, N., Iacovoni, J., Martinez, J., Holst, C., Astrup, A., Vidal, H., Clément, K., Hager, J., Saris, W., and Langin, D. (2012).
Determinants of human adipose tissue gene expression: impact of diet, sex, metabolic status and cis genetic regulation.
PLoS Genetics, 8(9):e1002959.