Inférence de réseaux pour les données RNA-seq

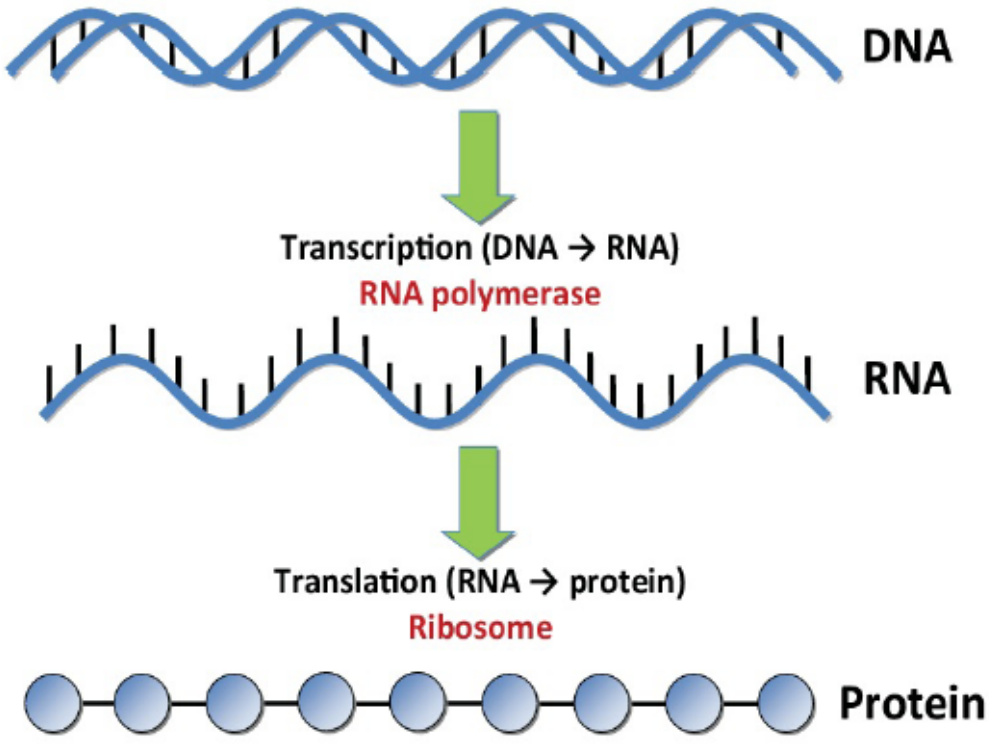
Mélina Gallopin

Université Paris Descartes

Séminaire MIAT (INRA, Auzeville)

Vendredi 26 mars 2016

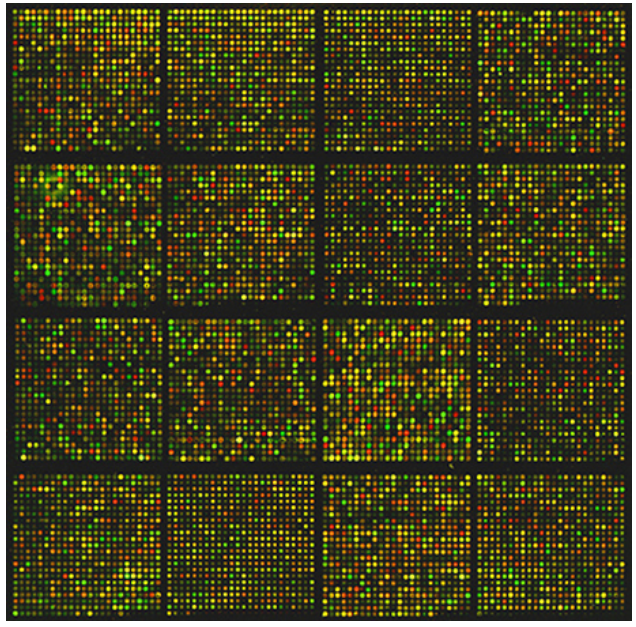
Mesurer l'expression des gènes



Mesurer l'expression des gènes

Comment?

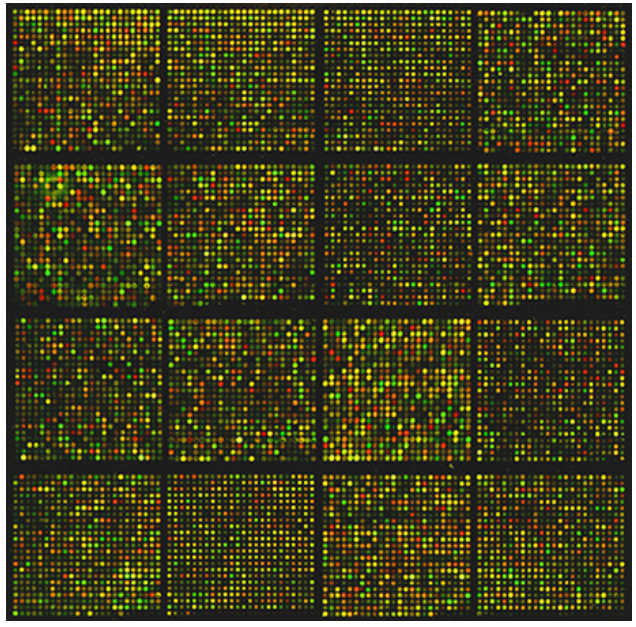
- les puces de ADN (1995) $\Rightarrow$ hybridation des séquences d'ADNc



Mesurer l'expression des gènes

Comment?

- les puces de ADN (1995) $\Rightarrow$ hybridation des séquences d'ADNc



- la technologie RNA-seq (2008) $\Rightarrow$ lecture des séquences d'ADNc

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

GATTACA, GTTTTTAGCTG, TAATTAG

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

génome de référence —TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

génome de référence —TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

génome de référence —TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$\textit{GATTACA}, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

read aligné	GATTACA
génome de référence	—TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

read aligné	TTAGCTC
read aligné	GATTACA
génome de référence	—TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

read aligné	GCTCTGAT
read aligné	TTAGCTC
read aligné	GATTACA
génome de référence	—TATTTAGCTCTGATTACAATG—

La technologie de séquencage RNA-seq

Pour un échantillon :

1. Extraction de l'ARN
2. Retranscription ARN $\Rightarrow$ ADNc
3. **Lecture des brins d'ADNc, appelés *reads***

$$GATTACA, \text{GTTTTTAGCTG}, \text{TAATTAG}$$

4. **Alignement des *reads***

read aligné	GCTCTGAT
read aligné	TTAGCTC
read aligné	GATTACA
génome de référence	—TATTTAGCTCTGATTACAATG—

5. **Comptage des *reads***

nombre de *reads*	45	17685	0	15
génome de référence	– gène 1 ———	gène 2 –	gène 3 –	gène 4 —

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

$\mathbf{y}_i$: expression pour l'échantillon i pour $i = 1, \ldots, n$

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

$\mathbf{y}_{ij}$: expression du gène j pour l'échantillon i

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

$\mathbf{y}_i$: expression pour l'échantillon i pour $i = 1, \ldots, n$

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

$\mathbf{y}_{ij}$: expression du gène j pour l'échantillon i

$\mathbf{y}_{ij}$ *est un comptage* $\rightarrow$ *Modélisations discrètes*

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

$\mathbf{y}_i$: expression pour l'échantillon i pour $i = 1, \ldots, n$

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

$\mathbf{y}_{ij}$: expression du gène j pour l'échantillon i

$\mathbf{y}_{ij}$ *est un comptage* $\rightarrow$ *Modélisations discrètes*

$$p \text{ gènes } > 1000$$
$$n \text{ échantillons } < 100$$

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

$\mathbf{y}_i$: expression pour l'échantillon i pour $i = 1, \ldots, n$

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

$\mathbf{y}_{ij}$: expression du gène j pour l'échantillon i

$\mathbf{y}_{ij}$ *est un comptage* $\rightarrow$ *Modélisations discrètes*

p *gènes* > 1000

n *échantillons* < 100

Technologie RNA-seq couteuse $\rightarrow$ *Manque d'échantillons*

Les données RNA-seq de Mach *et al.*, 2014

		gène 1	gène 2	gène 3	gène 4	gène 5	...
duodenum	porcelet 1	4	199	2987	0	65	...
	porcelet 2	0	189	1806	0	29	...
	porcelet 3	6	201	1752	48	599	...
	porcelet 4	4	198	2987	0	65	...
jejunum	porcelet 1	0	0	1296	0	49	...
	porcelet 2	6	0	2298	0	119	...
	porcelet 3	4	0	2987	0	651	...
	porcelet 4	0	0	1876	0	219	...
ileum	porcelet 1	0	19931	1837	0	388	...
	porcelet 2	2	18319	8786	0	861	...
	porcelet 3	7	23101	2237	0	76	...
	porcelet 4	1	34198	9828	0	65	...

$\mathbf{y}_i$: expression pour l'échantillon i pour $i = 1, \ldots, n$

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

$\mathbf{y}_{ij}$: expression du gène j pour l'échantillon i

$\mathbf{y}_{ij}$ *est un comptage* $\rightarrow$ *Modélisations discrètes*

p *gènes* > 1000

n *échantillons* < 100

Technologie RNA-seq couteuse $\rightarrow$ *Manque d'échantillons*

Analyses statistiques des données d'expression

- **Analyse différentielle**
  Déterminer si un gène est plus ou moins exprimé dans deux conditions

Analyses statistiques des données d'expression

- **Analyse différentielle**
  Déterminer si un gène est plus ou moins exprimé dans deux conditions

- **Analyse de co-expression**
  Grouper les gènes ayant des profils d'expression similaires

Analyses statistiques des données d'expression

- **Analyse différentielle**
  Déterminer si un gène est plus ou moins exprimé dans deux conditions

- **Analyse de co-expression**
  Grouper les gènes ayant des profils d'expression similaires

- **Inférence de réseaux**
  Reconstituer les réseaux de régulation génique

Analyses statistiques des données d'expression

- **Analyse différentielle**
  Déterminer si un gène est plus ou moins exprimé dans deux conditions

- **Analyse de co-expression**
  Grouper les gènes ayant des profils d'expression similaires

- **Inférence de réseaux**
  Reconstituer les réseaux de régulation génique

Inférence de réseaux

Principe

Reconstruire un graphe $G = (V, E)$ où :

- $V = \{1, \ldots, p\}$ ensemble des nœuds représentant les variables
- E ensemble des arrêtes modélisant les dépendances entre les variables

Inférence de réseaux

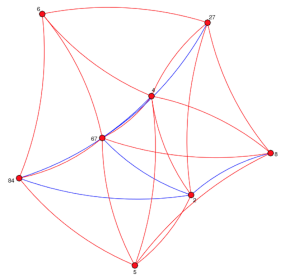
Principe

Reconstruire un graphe $G = (V, E)$ où :

- $V = \{1, \ldots, p\}$ ensemble des nœuds représentant les variables
- E ensemble des arrêtes modélisant les dépendances entre les variables

Objectif

Reconstituer des réseaux de régulation de gènes



Inférence de réseaux

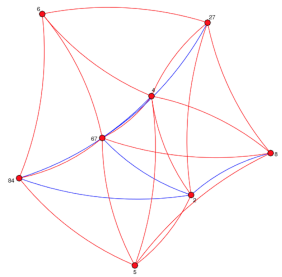
Principe

Reconstruire un graphe $G = (V, E)$ où :

- $V = \{1, \ldots, p\}$ ensemble des nœuds représentant les variables
- E ensemble des arrêtes modélisant les dépendances entre les variables

Objectif

Reconstituer des réseaux de régulation de gènes



Outils

- Les modèles graphiques orientés
- Les modèles graphiques non-orientés

Inférence de réseaux

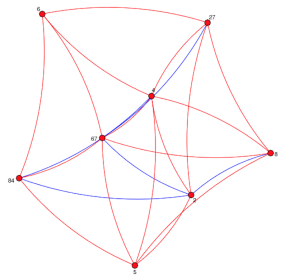
Principe

Reconstruire un graphe $G = (V, E)$ où :

- $V = \{1, \ldots, p\}$ ensemble des nœuds représentant les variables
- E ensemble des arrêtes modélisant les dépendances entre les variables

Objectif

Reconstituer des réseaux de régulation de gènes



Outils

- Les modèles graphiques orientés
- Les modèles graphiques non-orientés

Les modèles graphiques non-orientés

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

Les modèles graphiques non-orientés

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

Théorème (Hammersley-Clifford) La densité p des données se factorise ainsi

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \frac{1}{Z} \prod_{\mathcal{C} \in \mathfrak{C}} \psi_{\mathcal{C}}(\mathbf{y}^{\mathcal{C}}).$$

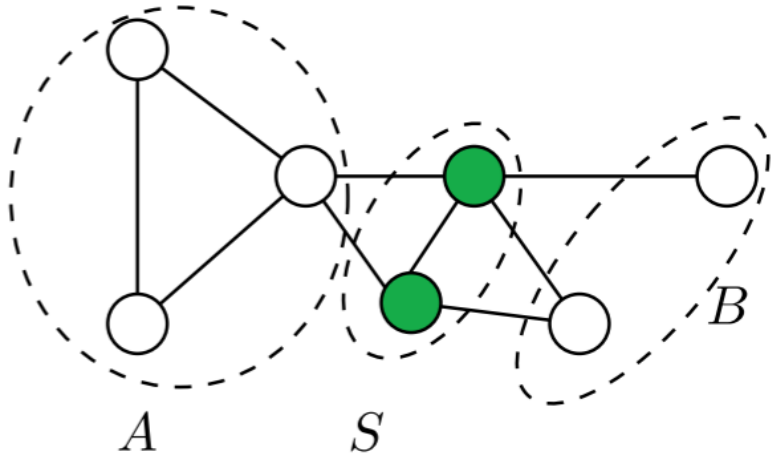
Les modèles graphiques non-orientés

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

Théorème (Hammersley-Clifford) La densité p des données se factorise ainsi

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \frac{1}{Z} \prod_{\mathcal{C} \in \mathfrak{C}} \psi_{\mathcal{C}}(\mathbf{y}^{\mathcal{C}}).$$

ssi $(\mathbf{y}^1, \ldots, \mathbf{y}^p)$ vérifie la propriété de Markov par rapport au graphe G :



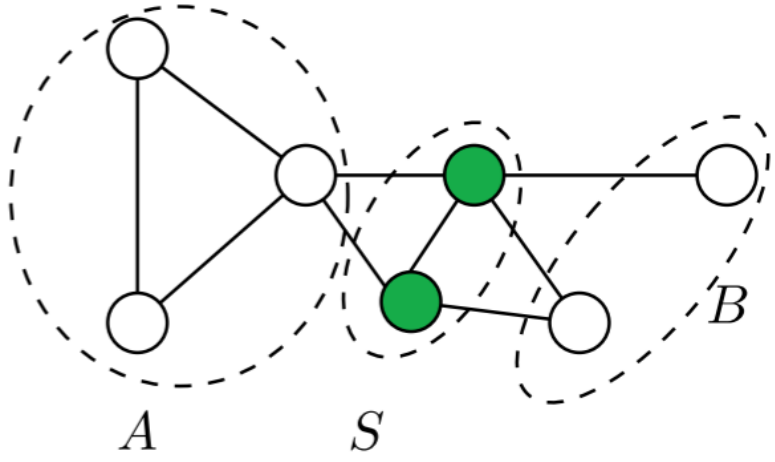
Les modèles graphiques non-orientés

$\mathbf{y}^j$: expression du gène j pour $j = 1, \ldots, p$

Théorème (Hammersley-Clifford) La densité p des données se factorise ainsi

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \frac{1}{Z} \prod_{\mathcal{C} \in \mathfrak{C}} \psi_{\mathcal{C}}(\mathbf{y}^{\mathcal{C}}).$$

ssi $(\mathbf{y}^1, \ldots, \mathbf{y}^p)$ vérifie la propriété de Markov par rapport au graphe G :



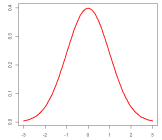
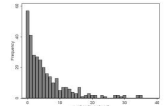
Exemple : Le modèle graphique gaussien (GGM) $\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma)$

1. Les données RNA-seq

2. Modélisation

3. Réduction de dimension

Etat de l'art des méthodes développées

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne: `limma` ***(Smyth et al., 2005)***	binomiale négative: `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***,
Analyse de co-expression (modèle de mélange)	gaussienne: `Rmixmod` ***(Biernacki et al., 2006)***	Poisson: `HTSCluster` ***(Rau et al., 2015)***
Inférence de réseaux (modèle non-orienté)	gaussienne: `SIMoNe` ***(Chiquet et al., 2010)***	?

Inférence de réseaux pour les données RNA-seq

Le modèle graphique gaussien (GGM)

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \exp\left[-\frac{1}{2}\sum_{(j,j')\in E} \theta_{jj'}\mathbf{y}^j\mathbf{y}^{j'} - A(\Theta)\right]$$

$$\text{où } A(\Theta) = -\frac{1}{2}\log\det\left[\frac{\Theta}{2\pi}\right]$$

Inférence de réseaux pour les données RNA-seq

Le modèle graphique gaussien (GGM)

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \exp\left[-\frac{1}{2}\sum_{(j,j')\in E} \theta_{jj'}\mathbf{y}^j\mathbf{y}^{j'} - A(\Theta)\right]$$

$$\text{où } A(\Theta) = -\frac{1}{2}\log\det\left[\frac{\Theta}{2\pi}\right]$$

Le modèle graphique de Poisson

$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \exp\left[\sum_{j\in V}(\beta_j\mathbf{y}^j - \log\left(\mathbf{y}^j!\right)) + \sum_{(j,j')\in E}\beta_{jj'}\mathbf{y}^j\mathbf{y}^{j'} - A(\boldsymbol{\beta})\right]$$

Calcul du terme de normalisation $A(\boldsymbol{\beta})$ difficile

$\rightarrow$ modélise uniquement des dépendances négatives

Inférence de réseaux pour les données RNA-seq

- Inférence par sélection de voisinage

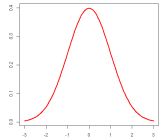
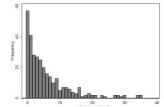
$$p(\mathbf{y}^1, \ldots, \mathbf{y}^p) = \exp\left[-\frac{1}{2}\sum_{(j,j')\in E} \theta_{jj'}\mathbf{y}^j\mathbf{y}^{j'} - A(\Theta)\right]$$

Meinshausen and Buhlmann (2006):

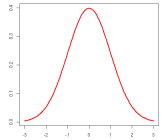
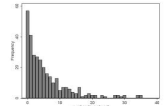
$$\mathbf{y}^j = \sum_{j'\neq j} \eta_{jj'}\mathbf{y}^{j'} + \epsilon_j,\ \epsilon_j \sim \mathcal{N}(0, \sigma_j),\ \eta_{jj'} = \frac{\theta_{jj'}}{\theta_{jj}}.$$

- Sélection de voisinage pour les modèles discrets
  - Basée sur des lois de Poisson (Allen *et al.*, 2013)
  - Basée sur des lois de Poisson sur-dispersées (Gallopin *et al.*, 2013)

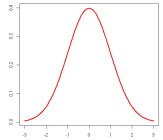
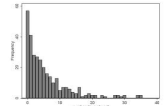
Modélisation des données RNA-seq

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne : `limma` ***(Smyth et al., 2005)***	binomiale négative : `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***,
Analyse de co-expression (modèle de mélange)	gaussienne : `Rmixmod` ***(Biernacki et al., 2006)***	Poisson : `HTSCluster` ***(Rau et al., 2015)***
Inférence de réseaux (modèle non-orienté)	gaussienne : `SIMoNe` ***(Chiquet et al., 2010)***	Poisson; Poisson sur-dispersée ***(Allen et al., 2012; Gallopin et al., 2013)***

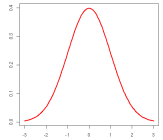
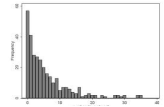
Modélisation des données RNA-seq

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne : `limma` ***(Smyth et al., 2005)***	binomiale négative : `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***, ou gaussienne *sur données transformées* : `limma` + "voom" ***(Law et al., 2014)***
Analyse de co-expression (modèle de mélange)	gaussienne : `Rmixmod` ***(Biernacki et al., 2006)***	Poisson : `HTSCluster` ***(Rau et al., 2015)***
Inférence de réseaux (modèle non-orienté)	gaussienne : `SIMoNe` ***(Chiquet et al., 2010)***	Poisson; Poisson sur-dispersée ***(Allen et al., 2012; Gallopin et al., 2013)***

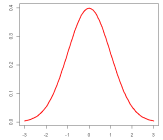
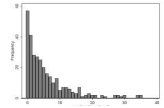
Modélisation des données RNA-seq

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne : `limma` ***(Smyth et al., 2005)***	binomiale négative : `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***, ou gaussienne *sur données transformées* : `limma` + "voom" ***(Law et al., 2014)***
Analyse de co-expression (modèle de mélange)	gaussienne : `Rmixmod` ***(Biernacki et al., 2006)***	Poisson : `HTSCluster` ***(Rau et al., 2015)*** ou gaussienne *sur données transformées*
Inférence de réseaux (modèle non-orienté)	gaussienne : `SIMoNe` ***(Chiquet et al., 2010)***	Poisson; Poisson sur-dispersée ***(Allen et al., 2012; Gallopin et al., 2013)***

Modélisation des données RNA-seq

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne : `limma` ***(Smyth et al., 2005)***	binomiale négative : `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***, ou gaussienne *sur données transformées* : `limma` + "voom" ***(Law et al., 2014)***
Analyse de co-expression (modèle de mélange)	gaussienne : `Rmixmod` ***(Biernacki et al., 2006)***	Poisson : `HTSCluster` ***(Rau et al., 2015)*** ou gaussienne *sur données transformées*
Inférence de réseaux (modèle non-orienté)	gaussienne : `SIMoNe` ***(Chiquet et al., 2010)***	Poisson; Poisson sur-dispersée ***(Allen et al., 2012; Gallopin et al., 2013)*** ou gaussienne *sur données transformées*

Modélisation des données RNA-seq

	Données de puces (1995) *continues* 	Données RNA-seq (2008) *discrètes* 
Analyse différentielle	gaussienne : `limma` ***(Smyth et al., 2005)***	binomiale négative : `DESeq`; `EdgeR` ***(Anders et al., 2010; Robinson et al., 2010)***, ou gaussienne *sur données transformées* : `limma` + "voom" ***(Law et al., 2014)***
Analyse de co-expression (modèle de mélange)	gaussienne : `Rmixmod` ***(Biernacki et al., 2006)***	Poisson : `HTSCluster` ***(Rau et al., 2015)*** ou gaussienne *sur données transformées*
Inférence de réseaux (modèle non-orienté)	gaussienne : `SIMoNe` ***(Chiquet et al., 2010)***	Poisson; Poisson sur-dispersée ***(Allen et al., 2012; Gallopin et al., 2013)*** ou gaussienne *sur données transformées*

$\rightarrow$ modélisations gaussiennes sont des alternatives raisonnables aux modélisations discrètes

Le modèle graphique gaussien

Modèle

$$\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma) \text{ pour } i = 1, \ldots, n$$

Chaque arête du réseau $\Leftrightarrow$ Coefficients non nuls de $\Theta = \Sigma^{-1}$

Le modèle graphique gaussien

Modèle

$$\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma) \text{ pour } i = 1, \ldots, n$$

Chaque arête du réseau $\Leftrightarrow$ Coefficients non nuls de $\Theta = \Sigma^{-1}$

Maximisation en Θ par *Graphical lasso* (Friedman *et al.*, 2008) de

$$\mathcal{L}_\lambda(\Theta; S) \propto \log\det(\Theta) - \text{tr}(S\Theta) - \lambda||\Theta||_1$$

avec S matrice de covariance empirique

Le modèle graphique gaussien

Modèle

$$\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma) \text{ pour } i = 1, \ldots, n$$

Chaque arête du réseau $\Leftrightarrow$ Coefficients non nuls de $\Theta = \Sigma^{-1}$

Maximisation en Θ par *Graphical lasso* (Friedman *et al.*, 2008) de

$$\mathcal{L}_\lambda(\Theta; S) \propto \log\det(\Theta) - \text{tr}(S\Theta) - \lambda||\Theta||_1$$

avec S matrice de covariance empirique

Algorithme *Block Diagonal Screening Rule* (Mazumder et Hastie, 2012)

Pour un paramètre de régularisation fixé λ

Etape 1 Seuillage de $|\ S\ |$ au niveau $\lambda \Rightarrow$ structure en blocs

Etape 2 Graphical lasso au paramètre λ dans chaque bloc

Le modèle graphique gaussien

Modèle

$$\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma) \text{ pour } i = 1, \ldots, n$$

Chaque arête du réseau $\Leftrightarrow$ Coefficients non nuls de $\Theta = \Sigma^{-1}$

Maximisation en Θ par *Graphical lasso* (Friedman *et al.*, 2008) de

$$\mathcal{L}_\lambda(\Theta; S) \propto \log\det(\Theta) - \text{tr}(S\Theta) - \lambda||\Theta||_1$$

avec S matrice de covariance empirique

Algorithme *Block Diagonal Screening Rule* (Mazumder et Hastie, 2012)

Pour un paramètre de régularisation fixé λ

Etape 1 Seuillage de $|\ S\ |$ au niveau $\lambda \Rightarrow$ structure en blocs

Etape 2 Graphical lasso au paramètre λ dans chaque bloc

$$n < p$$

Modèle graphique gaussien et grande dimension

Phénomène d'*ultra-grande dimension* (Verzelen, 2012)

$$\frac{d \log(\frac{p}{d})}{n} \geq \frac{1}{2}$$

où d est le degré maximal du réseau

Exemple : $n = 50, p = 200, d \geq 8$

Modèle graphique gaussien et grande dimension

Phénomène d'*ultra-grande dimension* (Verzelen, 2012)

$$\frac{d \log(\frac{p}{d})}{n} \geq \frac{1}{2}$$

où d est le degré maximal du réseau

Exemple : $n = 50, p = 200, d \geq 8$

Solutions

1. Restreindre le nombre de gènes à l'aide d'informations externes

Modèle graphique gaussien et grande dimension

Phénomène d'*ultra-grande dimension* (Verzelen, 2012)

$$\frac{d \log(\frac{p}{d})}{n} \geq \frac{1}{2}$$

où d est le degré maximal du réseau

Exemple : $n = 50, p = 200, d \geq 8$

Solutions

1. Restreindre le nombre de gènes à l'aide d'informations externes
2. Se focaliser uniquement sur les gènes les plus variables

Modèle graphique gaussien et grande dimension

Phénomène d'*ultra-grande dimension* (Verzelen, 2012)

$$\frac{d \log(\frac{p}{d})}{n} \geq \frac{1}{2}$$

où d est le degré maximal du réseau

Exemple : $n = 50, p = 200, d \geq 8$

Solutions

1. Restreindre le nombre de gènes à l'aide d'informations externes
2. Se focaliser uniquement sur les gènes les plus variables
3. Sélectionner automatiquement les gènes clefs

Modèle graphique gaussien et grande dimension

Phénomène d'*ultra-grande dimension* (Verzelen, 2012)

$$\frac{d \log(\frac{p}{d})}{n} \geq \frac{1}{2}$$

où d est le degré maximal du réseau

Exemple : $n = 50, p = 200, d \geq 8$

Solutions

1. Restreindre le nombre de gènes à l'aide d'informations externes
2. Se focaliser uniquement sur les gènes les plus variables
3. Sélectionner automatiquement les gènes clefs

Sélection automatique des gènes clefs

Cluster graphical lasso (Tan *et al.*, 2015)

1. Détection de K groupes par classification hiérarchique des variables

2. Glasso dans chaque groupe avec différents paramètres de régularisation

Sélection automatique des gènes clefs

Cluster graphical lasso (Tan *et al.*, 2015)

1. Détection de K groupes par classification hiérarchique des variables
   $\Rightarrow$ Choix de K par validation croisée
2. Glasso dans chaque groupe avec différents paramètres de régularisation

Sélection automatique des gènes clefs

Cluster graphical lasso (Tan *et al.*, 2015)

1. Détection de K groupes par classification hiérarchique des variables
   $\Rightarrow$ Choix de K par validation croisée
2. Glasso dans chaque groupe avec différents paramètres de régularisation

Notre procédure (Devijver & Gallopin, soumis)

1. Choix automatique des groupes par un critère de sélection de modèles
2. Glasso dans chaque groupe avec différents paramètres de régularisation

Détection des groupes de gènes

Hypothèse: $\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma_B)$ avec $\Sigma_B = \begin{pmatrix} \Sigma^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Sigma^K \end{pmatrix}$

La structure en K blocs de Σ_B définit les groupes de gènes $B = (B_1, \ldots, B_K)$

Détection des groupes de gènes

$$\textbf{Hypothèse: } \mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma_B) \text{ avec } \Sigma_B = \begin{pmatrix} \Sigma^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Sigma^K \end{pmatrix}$$

La structure en K blocs de Σ_B définit les groupes de gènes $B = (B_1, \ldots, B_K)$

$$F_B = \{f_B = \mathcal{N}_p(0, \Sigma_B) \text{ with } \Sigma_B \in S_B\}$$

$$S_B = \left\{ \Sigma_B \in \mathbb{S}_p^{++}(\mathbb{R}) \middle| \Sigma_B = P_\sigma \begin{pmatrix} \Sigma^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Sigma^K \end{pmatrix} P_\sigma^{-1}, \Sigma^k \in \mathbb{S}_{p_k}^{++}(\mathbb{R}), \forall k \in 1, \ldots, p \right\}$$

Détection des groupes de gènes

Hypothèse: $\mathbf{y}_i \overset{\text{iid}}{\sim} \mathcal{N}_p(\mathbf{0}, \Sigma_B)$ avec $\Sigma_B = \begin{pmatrix} \Sigma^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Sigma^K \end{pmatrix}$

La structure en K blocs de Σ_B définit les groupes de gènes $B = (B_1, \ldots, B_K)$

$$F_B = \{f_B = \mathcal{N}_p(0, \Sigma_B) \text{ with } \Sigma_B \in S_B\}$$

$$S_B = \left\{ \Sigma_B \in \mathbb{S}_p^{++}(\mathbb{R}) \middle| \Sigma_B = P_\sigma \begin{pmatrix} \Sigma^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Sigma^K \end{pmatrix} P_\sigma^{-1}, \Sigma^k \in \mathbb{S}_{p_k}^{++}(\mathbb{R}), \forall k \in 1, \ldots, p \right\}$$

Sélection par l'heuristique de pente (Birgé and Massart, 2007)

$$\hat{B} = \underset{B}{\text{argmin}} \left\{ -\frac{1}{n} \sum_{i=1}^{n} \log(\hat{f}_B(\mathbf{y}_i)) + \text{pen}(B) \right\},$$

$$\text{pen}(B) = \kappa D_B.$$

Une sous-collection aléatoire de modèle

$\mathcal{B}$: ensemble de toutes les partitions des variables possibles

$\Rightarrow$ Exploration exhaustive de $\mathcal{B}$ impossible

Une sous-collection aléatoire de modèle

$\mathcal{B}$: ensemble de toutes les partitions des variables possibles

$\Rightarrow$ Exploration exhaustive de $\mathcal{B}$ impossible

$\mathcal{B}^{\wedge}$: ensemble des partitions obtenues par seuillage de $|\, S \,|$

Une sous-collection aléatoire de modèle

$\mathcal{B}$: ensemble de toutes les partitions des variables possibles

$\Rightarrow$ Exploration exhaustive de $\mathcal{B}$ impossible

$\mathcal{B}^\wedge$: ensemble des partitions obtenues par seuillage de $|\, S \,|$

$$\hat{B} = \underset{B \in \mathcal{B}^\wedge}{\operatorname{argmin}} \left\{ -\frac{1}{n} \sum_{i=1}^{n} \log(\hat{f}_B(\mathbf{y}_i)) + \operatorname{pen}(B) \right\},$$

$$\operatorname{pen}(B) = \kappa D_B.$$

Une sous-collection aléatoire de modèle

$\mathcal{B}$: ensemble de toutes les partitions des variables possibles

$\Rightarrow$ Exploration exhaustive de $\mathcal{B}$ impossible

$\mathcal{B}^\wedge$: ensemble des partitions obtenues par seuillage de $|\, S \,|$

$$\hat{B} = \underset{B \in \mathcal{B}^\wedge}{\operatorname{argmin}} \left\{ -\frac{1}{n} \sum_{i=1}^{n} \log(\hat{f}_B(\mathbf{y}_i)) + \operatorname{pen}(B) \right\},$$

$$\operatorname{pen}(B) = \kappa D_B.$$

$\rightarrow$ **Procédure garantie par des résultats non-asymptotiques**

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\wedge} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\wedge} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

$\rightarrow$ Procédure de sélection de modèles optimale

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

$\rightarrow$ Procédure de sélection de modèles optimale

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

→ Procédure de sélection de modèles optimale
→ Calibration de la pénalité à partir des données

Garanties non-asymptotiques

Inégalité Oracle : On suppose qu'il existe une constante $\kappa' > 0$ telle que pour chaque partition $B \in \mathcal{B}$,

$$\mathrm{pen}(B) \geq \kappa' \frac{D_B}{n} \left[2c^2 + \rho \log \left(\frac{1}{D_B(\frac{D_B}{n} c^2 \wedge 1)} \right) + (1 \vee \tau) \log \left(\frac{0.792p}{\log(p+1)} \right) \right],$$

où c est une constante. Alors $\hat{f}_{\hat{B}}$ satisfait, pour une constante C :

$$\mathbb{E}(d_H^2(f^*, \hat{f}_{\hat{B}})) \leq C\mathbb{E} \left(\inf_{B \in \mathcal{B}^\Lambda} \left(\inf_{t \in F_B^{\text{borné}}} KL(f^*, t) + \mathrm{pen}(B) \right) + (1 \vee \tau) \frac{1}{n} \right).$$

Borne minimax : Pour tout $B \in \mathcal{B}$, il existe une constante $C_1 > 0$ telle que:

$$\inf_{\hat{f}_B} \sup_{f \in F_B^{\text{borné}}} \mathbb{E}(d_H^2(\hat{f}_B, f)) \geq C_1 \frac{D_B}{n} (1 + \log \left(\frac{C_2}{D_B^2} \right)).$$

→ Procédure de sélection de modèles optimale
→ Calibration de la pénalité à partir des données
→ Pénalité de la forme $\mathrm{pen}(B) = \kappa D_B$
à l'aide du package R capushe (Baudry *et al.*),

Calibration du coefficient κ dans $\mathrm{pen}(B) = \kappa D_B$

- Illustrations sur données simulées : $p = 100$, $n = 70$ et $K^\star = 15$

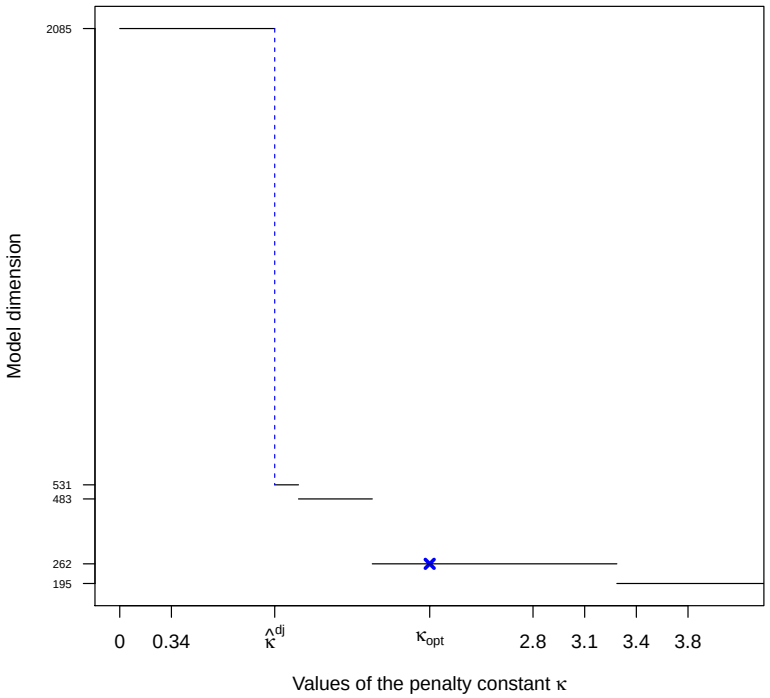
Calibration du coefficient κ dans $\mathrm{pen}(B) = \kappa D_B$

- Illustrations sur données simulées : $p = 100$, $n = 70$ et $K^\star = 15$
- Méthodes de calibration utilisées en pratique
  implémentées dans le package R `capushe` (Baudry *et al.*, 2012)

Calibration du coefficient κ dans $\mathrm{pen}(B) = \kappa D_B$

- Illustrations sur données simulées : $p = 100$, $n = 70$ et $K^\star = 15$
- Méthodes de calibration utilisées en pratique
  implémentées dans le package R `capushe` (Baudry *et al.*, 2012)

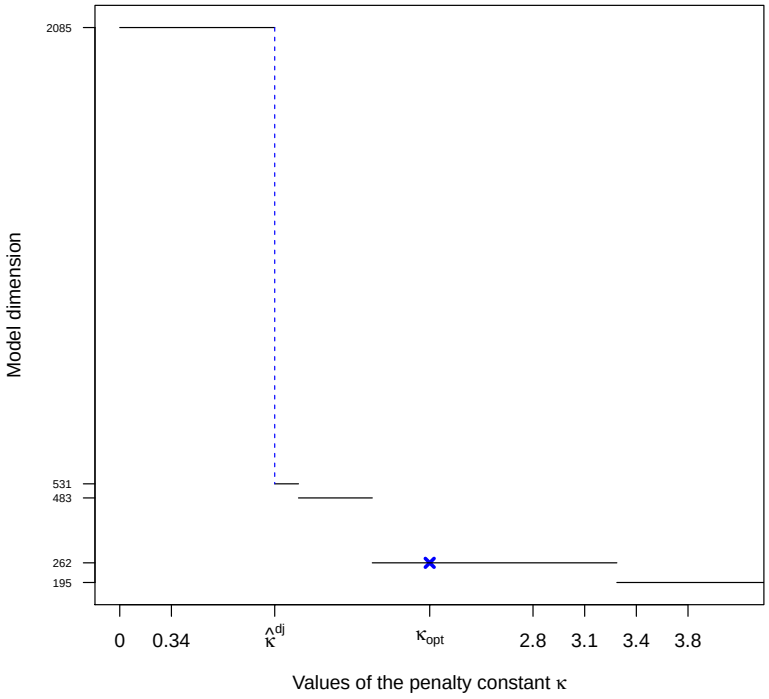
Méthode 1 : SHDJ
Slope Heuristics Dimension Jump



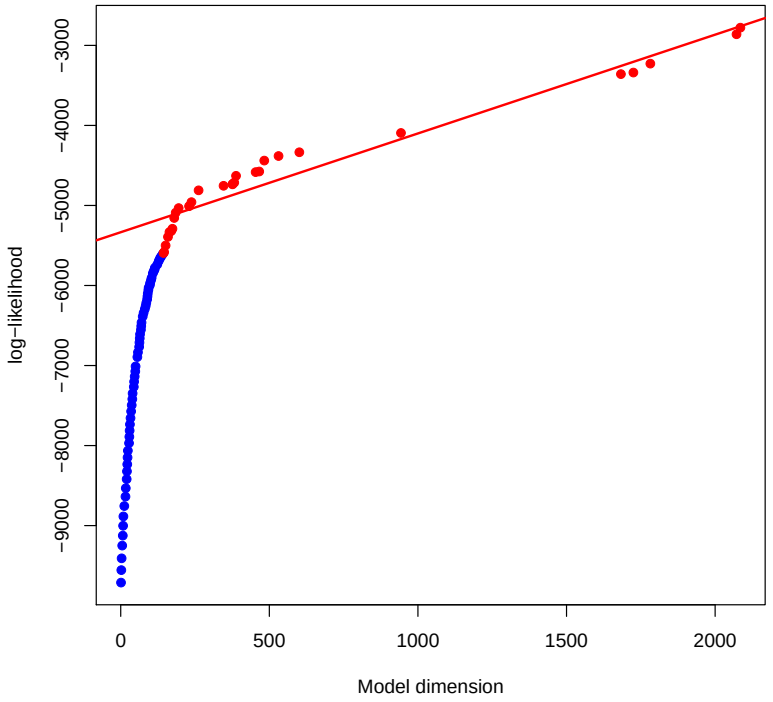
Calibration du coefficient κ dans $\mathrm{pen}(B) = \kappa D_B$

- Illustrations sur données simulées : $p = 100$, $n = 70$ et $K^\star = 15$
- Méthodes de calibration utilisées en pratique
  implémentées dans le package R `capushe` (Baudry *et al.*, 2012)

Méthode 1 : SHDJ
Slope Heuristics Dimension Jump

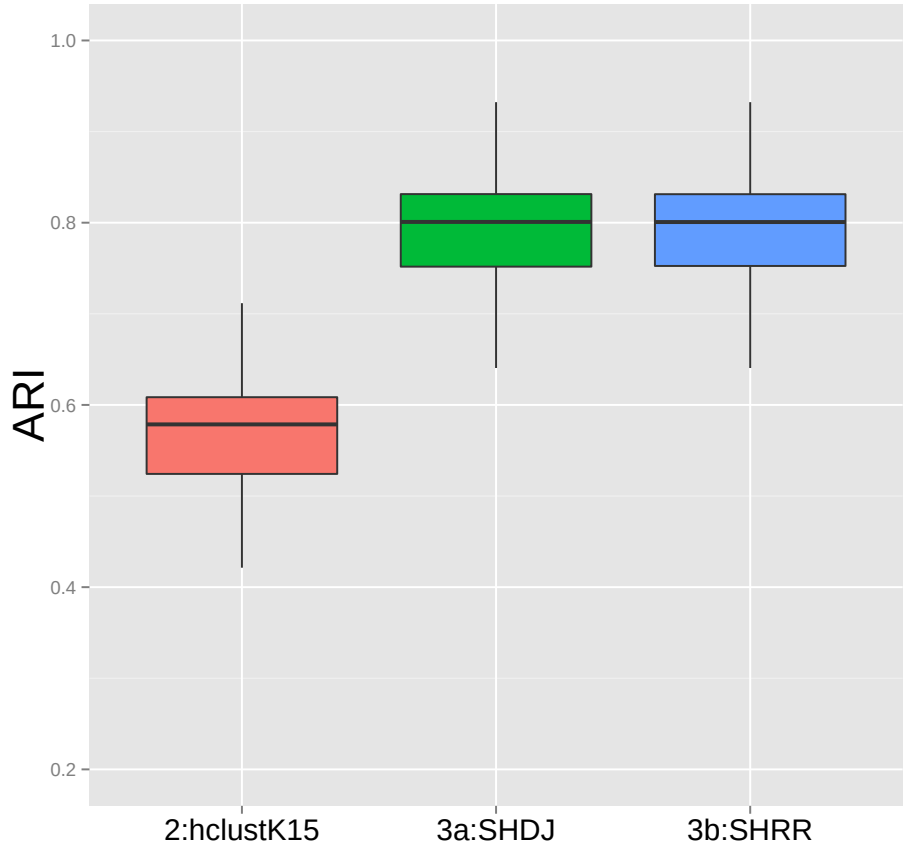


Méthode 2 : SHRR
Slope Heuristics Robust Regression



Données simulées

$p = 100$, $n = 70$ et Σ diagonale par blocs et $K^\star = 15$.



Adjusted Rand Index

entre la vraie partition et la partition détectée

Quatre stratégies d'inférence de réseaux

Données simulées : $p = 100$, $n = 70$ et Σ diagonale par blocs avec $K^\star = 15$.

- **(1) Glasso**
  graphical lasso sélectionné par BIC sur l'ensemble des variables

Quatre stratégies d'inférence de réseaux

Données simulées : $p = 100$, $n = 70$ et Σ diagonale par blocs avec $K^\star = 15$.

- **(1) Glasso**
  graphical lasso sélectionné par BIC sur l'ensemble des variables
- **(2) Cluster Graphical Lasso (Tan *et al.*, 2015)**

  Etape 1 : classification hiérarchique des variables, pour $K = K^\star$ fixé
  Etape 2 : $\rho_1, \ldots, \rho_{K^\star}$ d'après le corollaire de Tan 2015.

Quatre stratégies d'inférence de réseaux

Données simulées : $p = 100$, $n = 70$ et Σ diagonale par blocs avec $K^\star = 15$.

- **(1) Glasso**
  graphical lasso sélectionné par BIC sur l'ensemble des variables

- **(2) Cluster Graphical Lasso (Tan *et al.*, 2015)**

  Etape 1 : classification hiérarchique des variables, pour $K = K^\star$ fixé
  Etape 2 : $\rho_1, \ldots, \rho_{K^\star}$ d'après le corollaire de Tan 2015.

- **(3) Notre procédure**

  Etape 1 : (3a) SHRR partition
  (3b) SHDJ partition
  Etape 2 : graphical lassos sélectionnés par BIC

Quatre stratégies d'inférence de réseaux

Données simulées : $p = 100$, $n = 70$ et Σ diagonale par blocs avec $K^\star = 15$.

- **(1) Glasso**
  graphical lasso sélectionné par BIC sur l'ensemble des variables

- **(2) Cluster Graphical Lasso (Tan *et al.*, 2015)**

  Etape 1 : classification hiérarchique des variables, pour $K = K^\star$ fixé
  Etape 2 : $\rho_1, \ldots, \rho_{K^\star}$ d'après le corollaire de Tan 2015.

- **(3) Notre procédure**

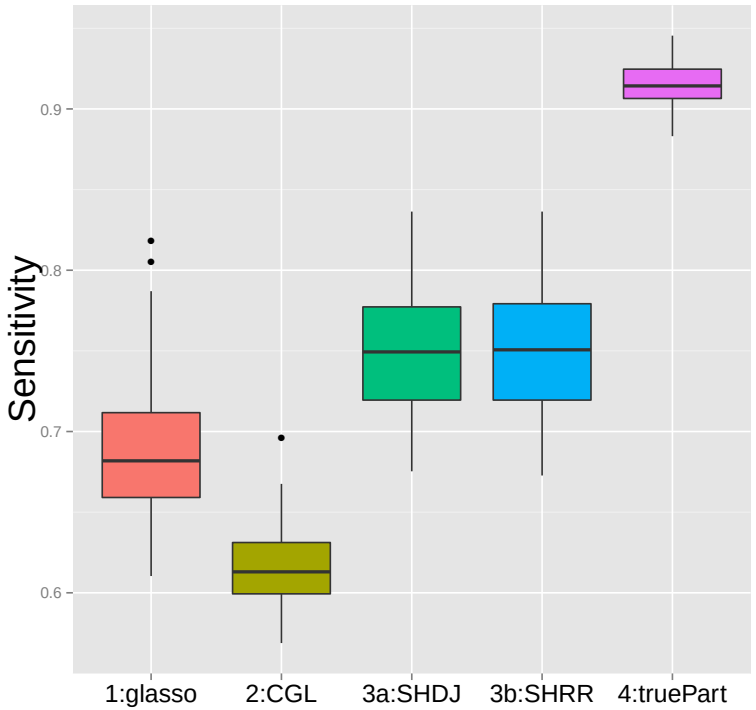
  Etape 1 : (3a) SHRR partition
  (3b) SHDJ partition
  Etape 2 : graphical lassos sélectionnés par BIC

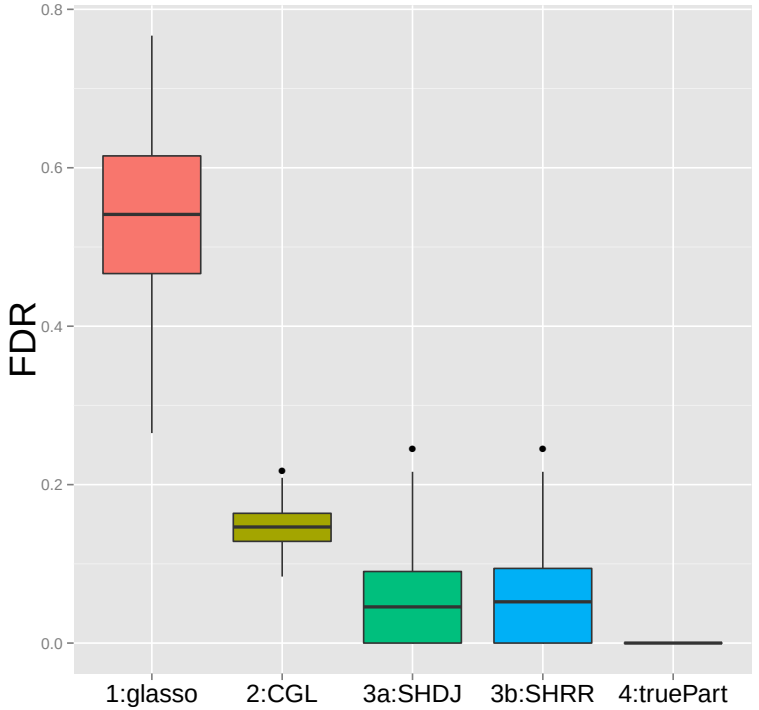
- **(4) Partition des variables connues**
  graphical lassos sélectionnés par BIC dans chaque bloc

Performance des stratégies sur données simulées

$p = 100$, $n = 70$ et Σ diagonale par blocs avec $K^\star = 15$.

$$\text{Sensibilité} = \frac{TP}{(TP + FN)} \qquad FDR = \frac{FP}{(TP + FP)}$$

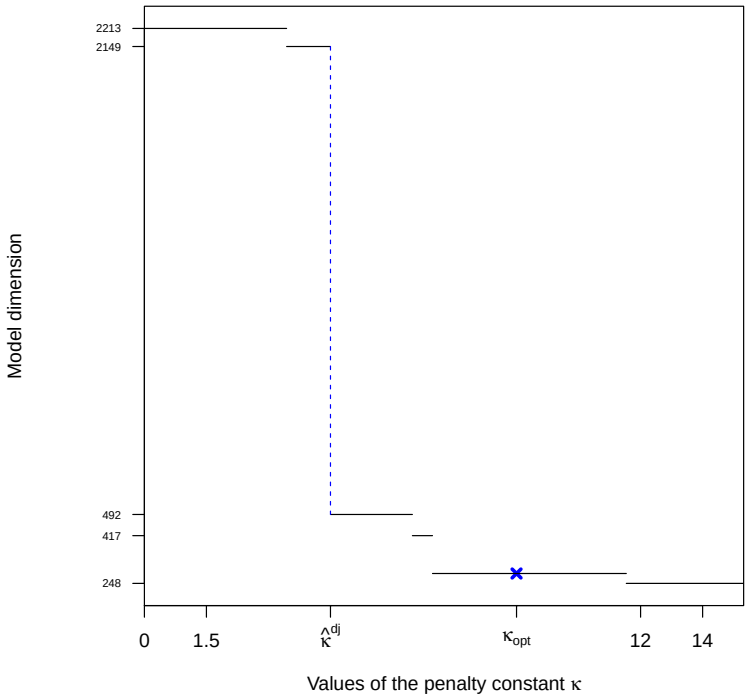


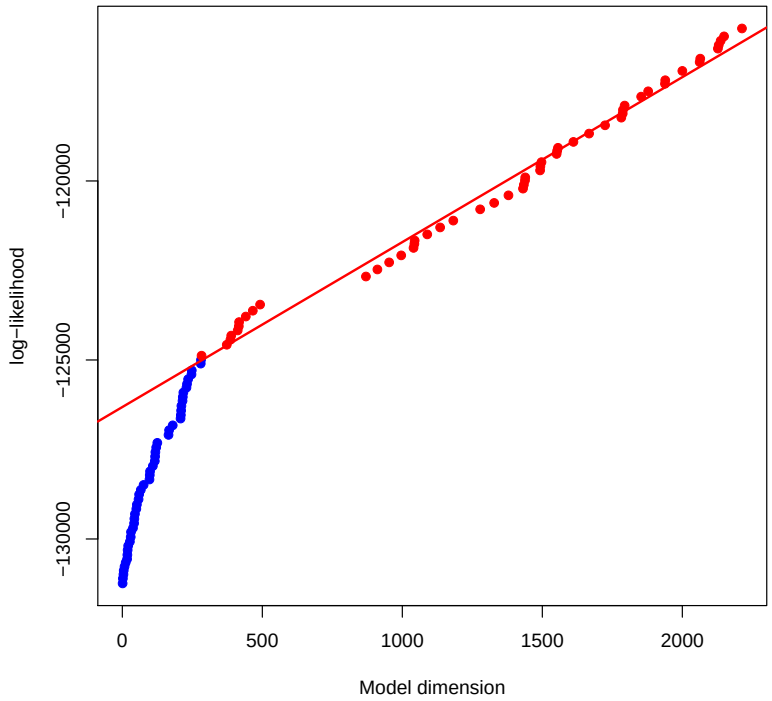


Sur 100 jeux de données simulés

Données réelles

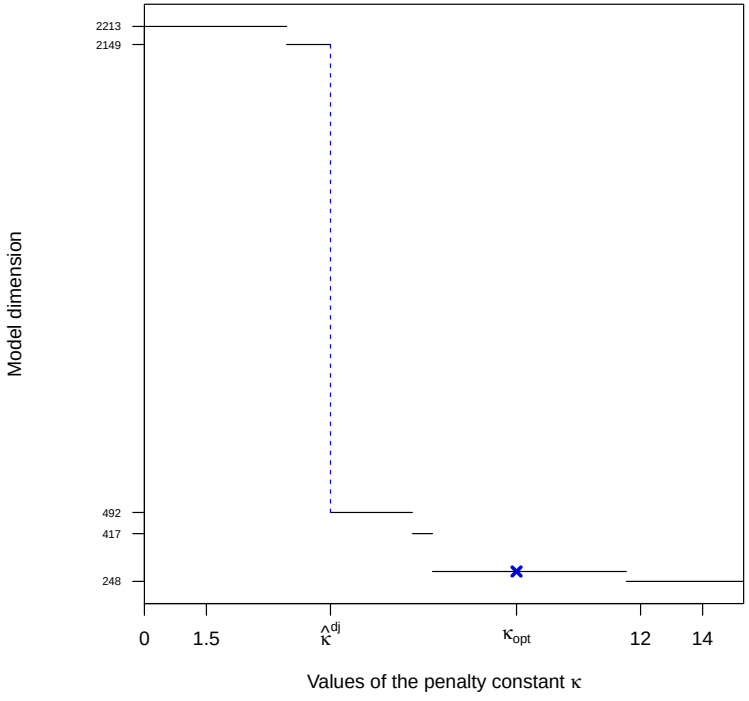
- Pickrell *et al.* (2010) : étude des lymphoblastes chez $n = 69$ individus
- Présélection des $p = 200$ gènes les plus variables

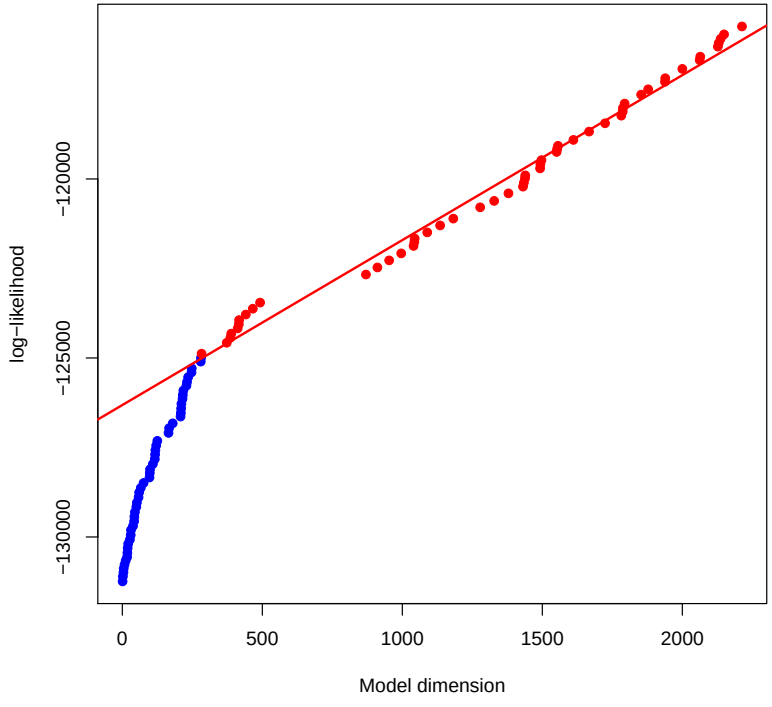




Données réelles

- Pickrell *et al.* (2010) : étude des lymphoblastes chez $n = 69$ individus
- Présélection des $p = 200$ gènes les plus variables





→ Les partitions SHRR et SHDJ coïncident

Inférence du réseau sur les données Pickrell *et al.*, (2010)

Graphical lasso

- $D = 19900$ paramètres à estimer

Inférence du réseau sur les données Pickrell *et al.*, (2010)

Graphical lasso

- $D = 19900$ paramètres à estimer

Partition détectée par l'heuristique des pentes $\hat{B}$

- $D_{\hat{B}_{\mathbf{SH}}} = 283$

Inférence du réseau sur les données Pickrell *et al.*, (2010)

Graphical lasso

- $D = 19900$ paramètres à estimer

Partition détectée par l'heuristique des pentes $\hat{B}$

- $D_{\hat{B}_{\mathbf{SH}}} = 283$
- $\hat{K}_{\mathsf{SH}} = 150$ blocs

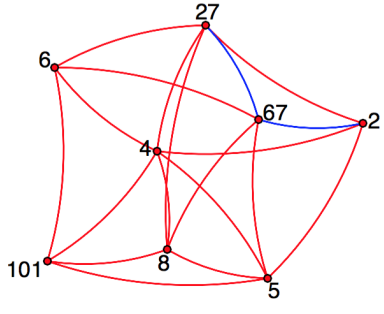
Inférence du réseau sur les données Pickrell *et al.*, (2010)

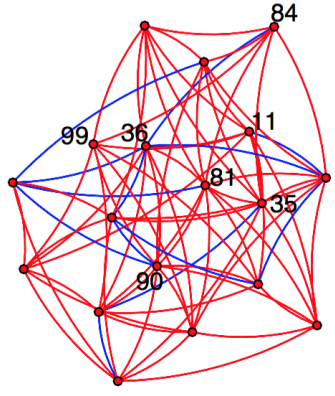
Graphical lasso

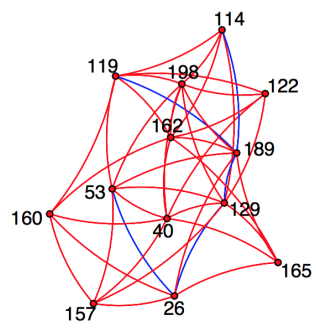
- $D = 19900$ paramètres à estimer

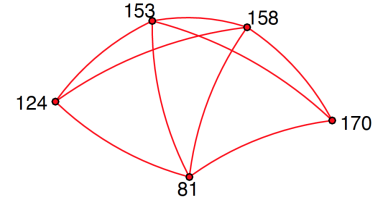
Partition détectée par l'heuristique des pentes $\hat{B}$

- $D_{\hat{B}_{\mathbf{SH}}} = 283$
- $\hat{K}_{\mathbf{SH}} = 150$ blocs
- 140 blocs de taille 1, 2 blocs de taille 2, 4 blocs de taille 3 et 4 blocs de taille $18, 13, 8$ et 5









Conclusion et perspectives

Conclusion

- Une procédure non-asymptotique pour réduire la dimension du problème d'inférence (Devijver et Gallopin, soumis)
- implémentée dans le package `shock` disponible sur le `CRAN`, développé sur Github `https://github.com/Gallopin/shock`

Conclusion et perspectives

Conclusion

- Une procédure non-asymptotique pour réduire la dimension du problème d'inférence (Devijver et Gallopin, soumis)
- implémentée dans le package `shock` disponible sur le `CRAN`, développé sur Github `https://github.com/Gallopin/shock`

Perspectives

- Applications sur des données réelles du département de génétique animale de l'INRA
- Inclusion d'informations externes pour améliorer la sélection des gènes : exemple, les données Hi-C

Merci pour votre attention

Bibliographie

- J. Whittaker, *Graphical Models in Applied Multivariate Statistics*, Wiley Publishing, 2009
- J. Friedman, T. Hastie, R. Tibshirani, *Sparse inverse covariance estimation with the Lasso*, Biostatistics, 2008
- R. Mazumder, T. Hastie, *Exact Covariance Thresholding into Connected Components for Large-Scale Graphical Lasso*, Journal of Machine Learning Reasearch, 2012
- D. Witten, J. Friedman, N. Simon, *New Insights and Faster Computations for the Graphical Lasso*, Journal of Computational and Graphical Statistics, 2011
- K. Tan, D. Witten, A. Shojaie, *The Cluster Graphical Lasso for improved estimation of Gaussian graphical models*, Computation Statistics & Data Analysis, 2015
- L. Birgé, P. Massart, *Minimal penalties for Gaussian model selection*, Probab. Theory Related Fields, 2007
- P. Massart, *Concentration inequalities and model selection*, Lecture Notes in Mathematics. Springer, 33, 2003, Saint-Flour, Cantal.