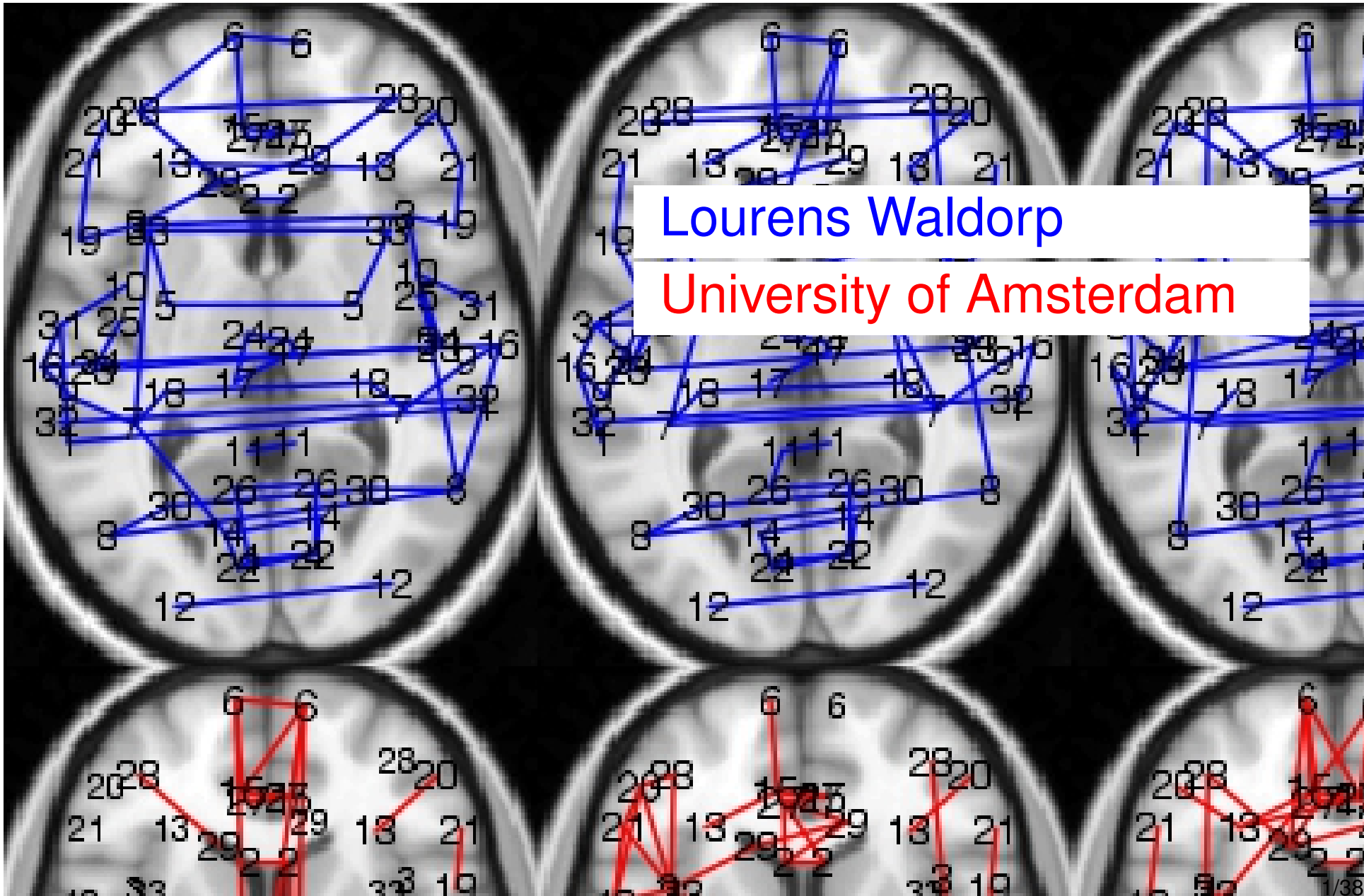
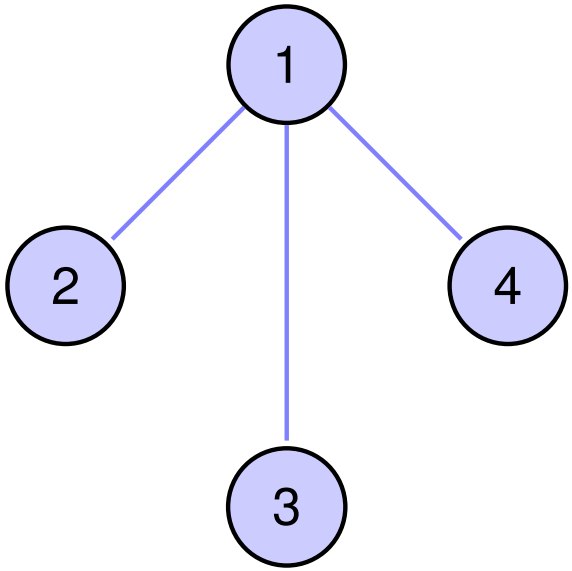
inference in large-scale networks

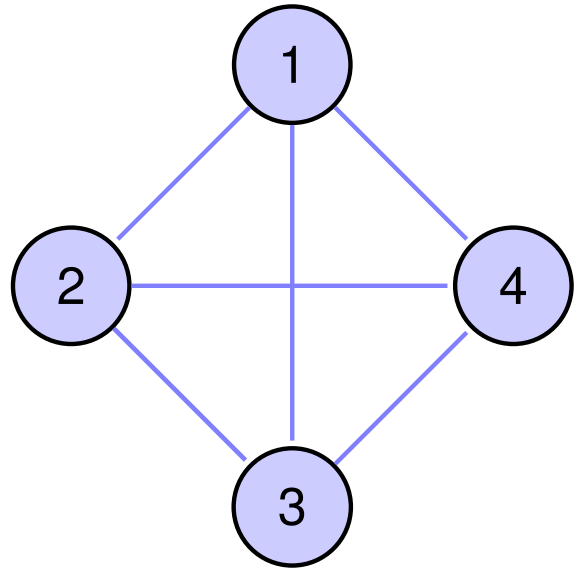


Lourens Waldorp

University of Amsterdam

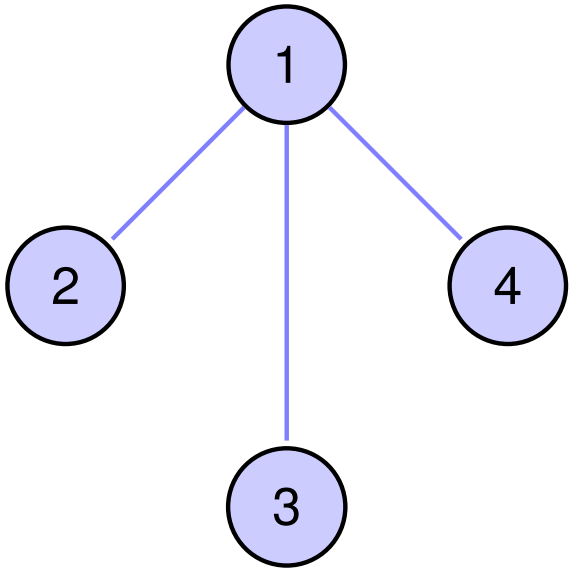
graphical models and networks

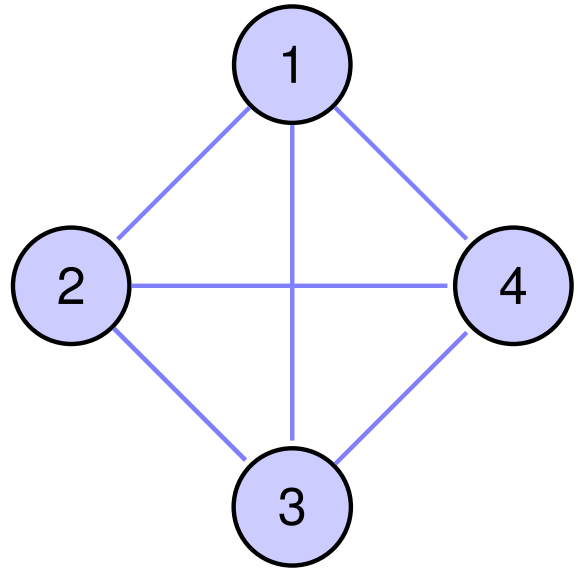




neighborhood $\text{ne}(i) = \{j \in V \backslash \{i\} : (i, j) \in E\}$

graphical models and networks





neighborhood $\text{ne}(i) = \{j \in V \backslash \{i\} : (i, j) \in E\}$

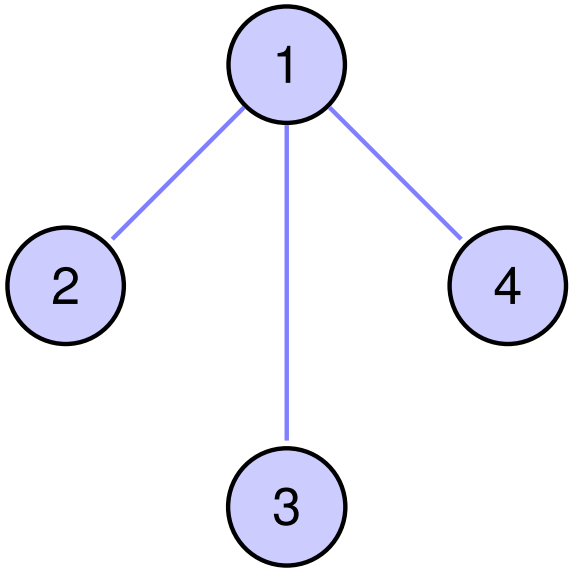
$\text{ne}(1) = \{2, 3, 4\}$

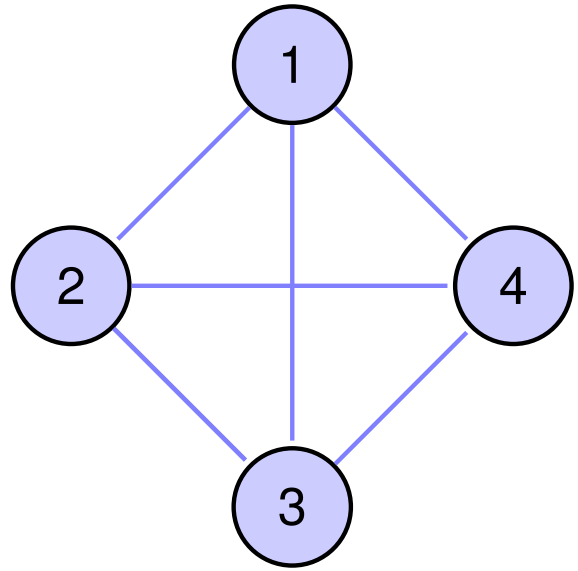
$\text{ne}(2) = \{1\}$

$\text{ne}(3) = \{1\}$

$\text{ne}(4) = \{1\}$

graphical models and networks



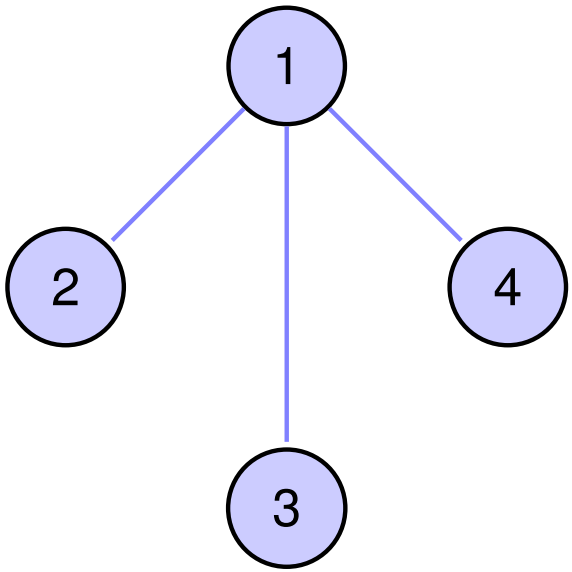


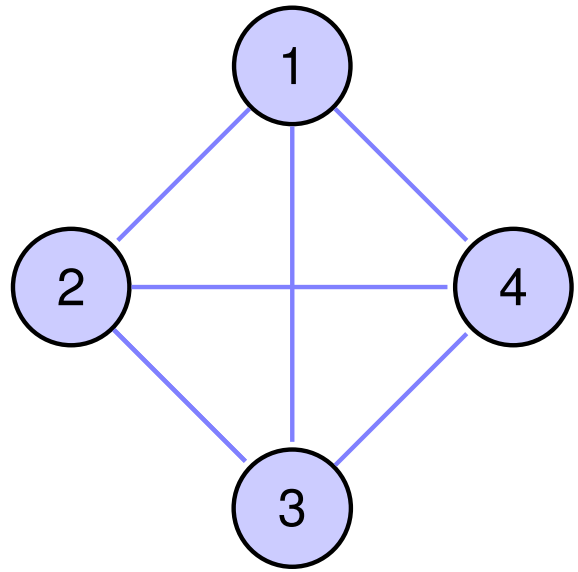
neighborhood $\text{ne}(i) = \{j \in V \backslash \{i\} : (i, j) \in E\}$

$\text{ne}(1) = \{2, 3, 4\}$
$\text{ne}(2) = \{1\}$
$\text{ne}(3) = \{1\}$
$\text{ne}(4) = \{1\}$

$\text{ne}(1) = \{2, 3, 4\}$
$\text{ne}(2) = \{1, 3, 4\}$
$\text{ne}(3) = \{1, 2, 4\}$
$\text{ne}(4) = \{1, 2, 3\}$

graphical models and networks



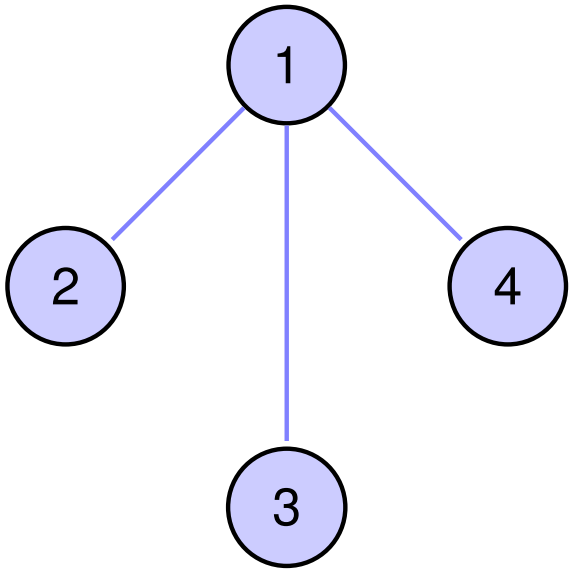


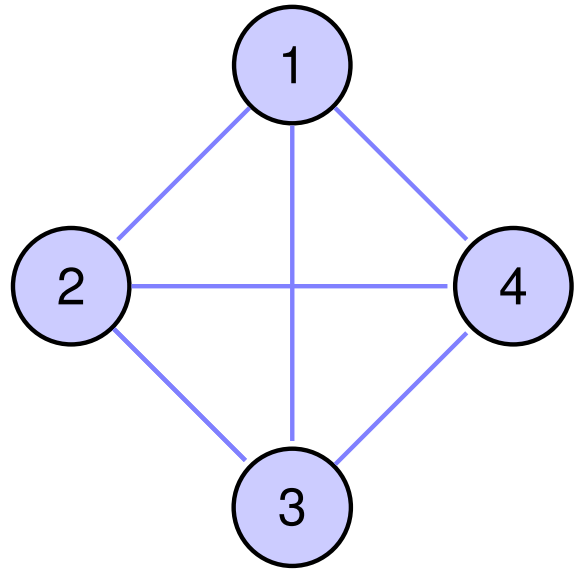
$$\mathbb{P}(1,2)\mathbb{P}(1,3)\mathbb{P}(1,4)$$

$$\mathbb{P}(1,2,3,4)$$

description in terms of cliques

graphical models and networks



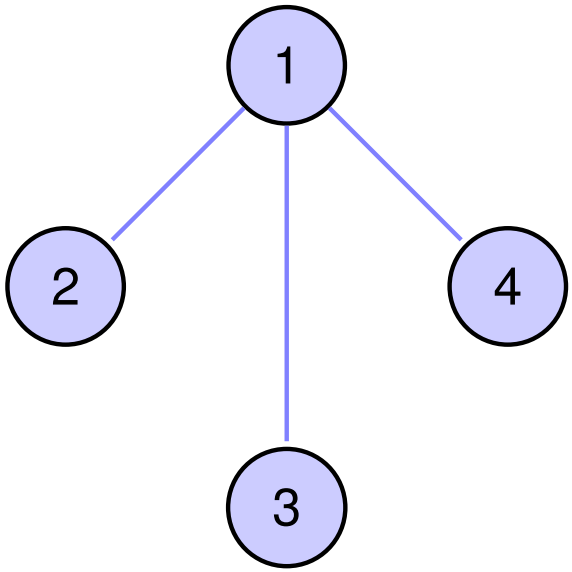


conditional independencies of graph

$$X_2 \perp\!\!\!\perp X_3 \mid X_1$$
$$X_2 \perp\!\!\!\perp X_4 \mid X_1$$
$$X_3 \perp\!\!\!\perp X_4 \mid X_1$$

$\varnothing$

Gaussian graphical models

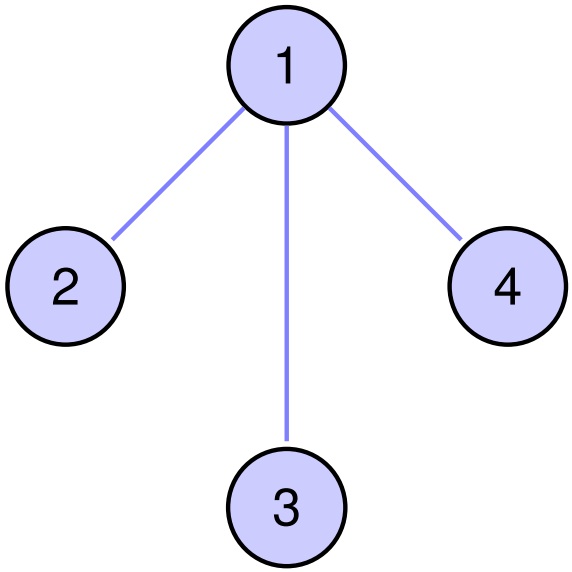


true graph

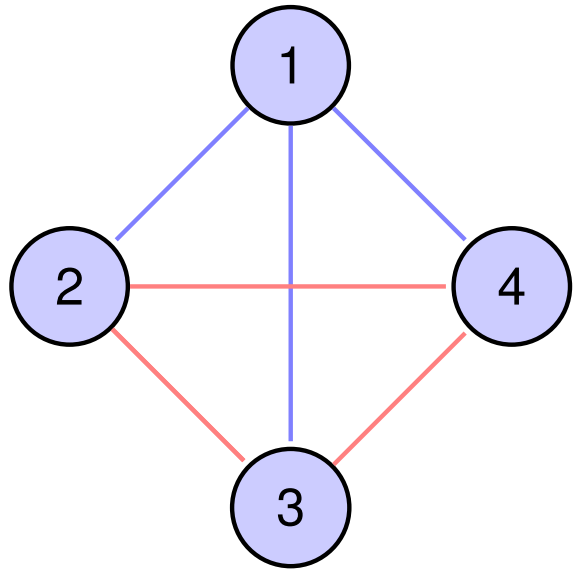
$$K = \Sigma^{-1} =$$

$$\begin{pmatrix} 1 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 1 \end{pmatrix}$$

Gaussian graphical models



true graph

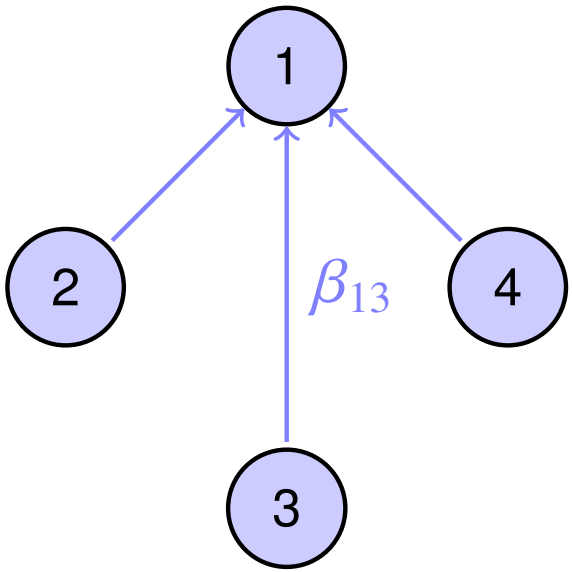


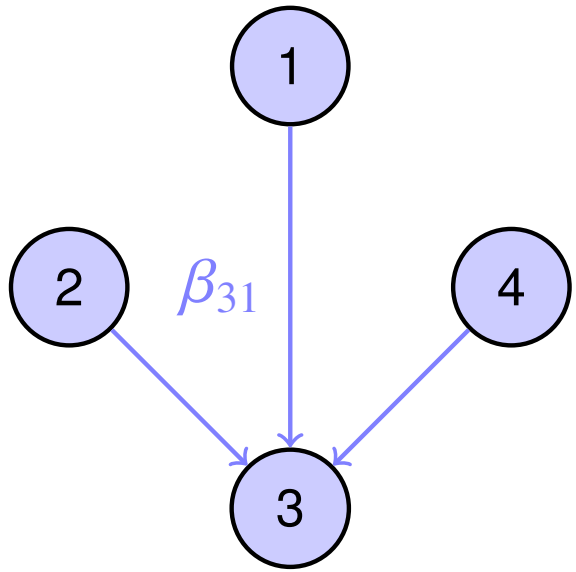
correlation graph

$$K = \Sigma^{-1} = \begin{pmatrix} 1 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 1 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 4 & -2 & -2 & -2 \\ -2 & 2 & 1 & 1 \\ -2 & 1 & 2 & 1 \\ -2 & 1 & 1 & 2 \end{pmatrix}$$

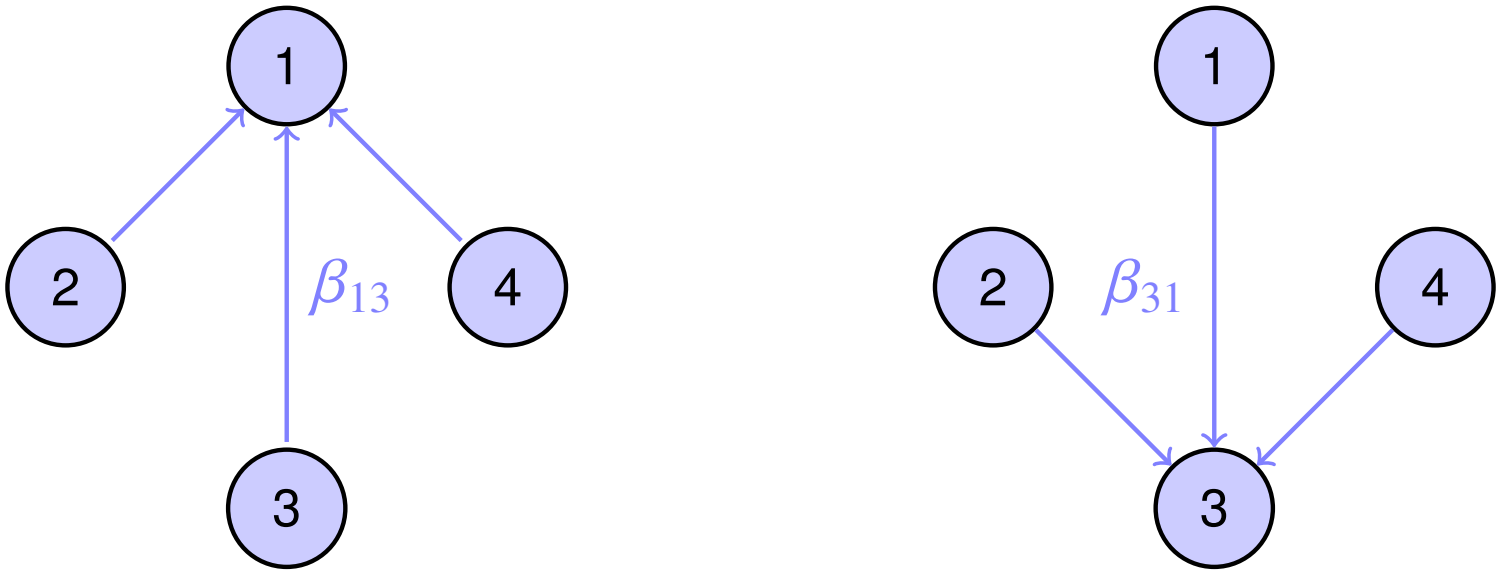
Gaussian graphical models





combine neighborhoods

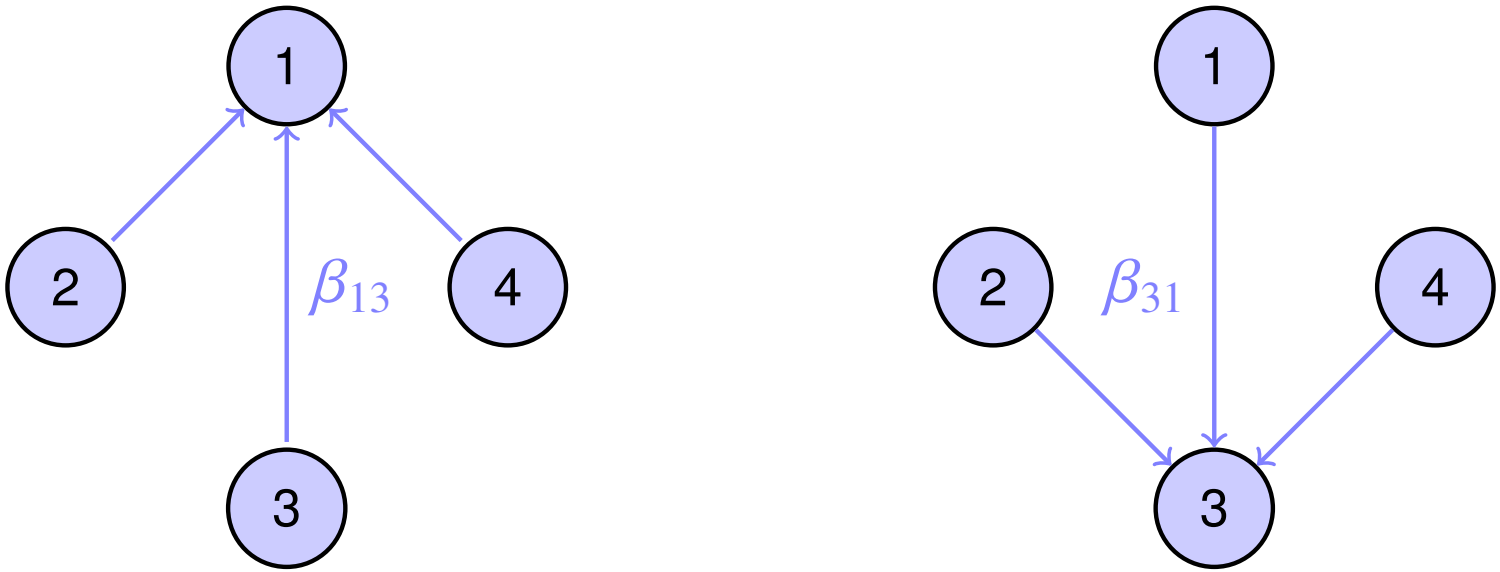
Gaussian graphical models



combine neighborhoods

- and node X_3 is in ne(1) if $\beta_{13} \neq 0$ and $\beta_{31} \neq 0$

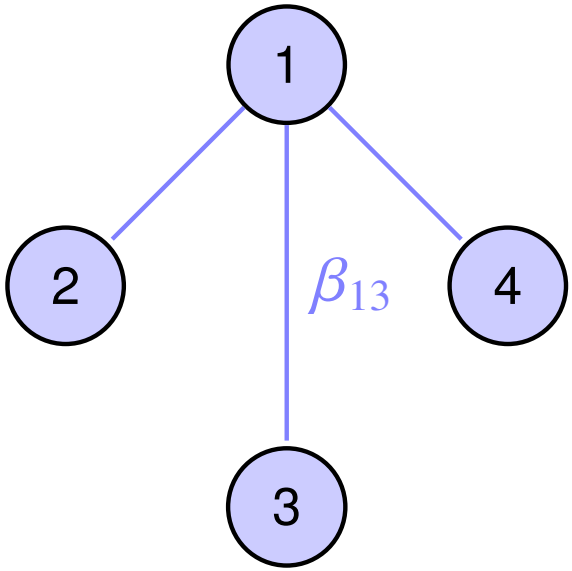
Gaussian graphical models



combine neighborhoods

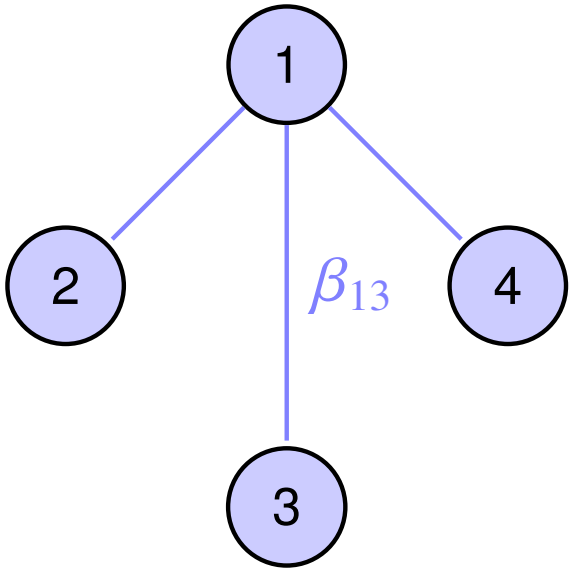
- and node X_3 is in ne(1) if $\beta_{13} \neq 0$ and $\beta_{31} \neq 0$
- or node X_3 is in ne(1) if $\beta_{13} \neq 0$ or $\beta_{31} \neq 0$

Gaussian graphical models



$$K = \Sigma^{-1} = \begin{pmatrix} 1 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 1 \end{pmatrix}$$

Gaussian graphical models

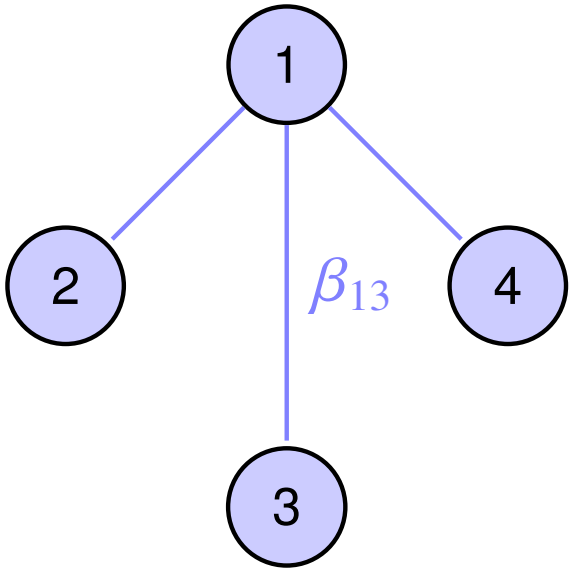


$$K = \Sigma^{-1} = \begin{pmatrix} 1 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 1 \end{pmatrix}$$

- let $K = \Sigma^{-1}$ then the neighborhood of X_1 is

$$\text{ne}(1) = \{i \in V \backslash \{1\} : K_{1i} \neq 0\} = \{2, 3, 4\}$$

Gaussian graphical models



$$K = \Sigma^{-1} = \begin{pmatrix} 1 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1 & 0 & 0 \\ 1/2 & 0 & 1 & 0 \\ 1/2 & 0 & 0 & 1 \end{pmatrix}$$

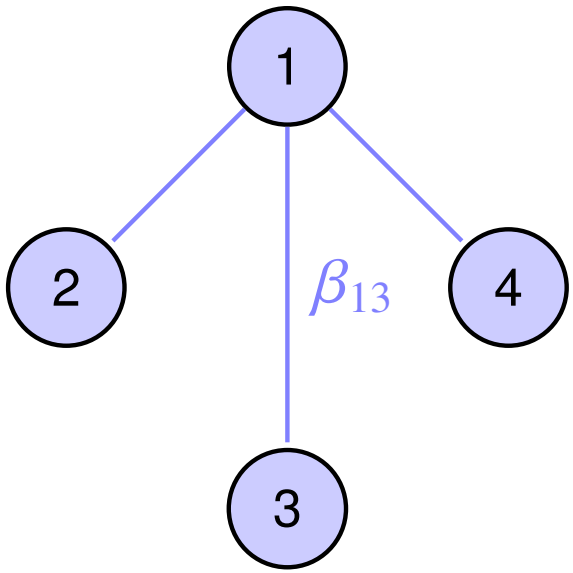
- let $K = \Sigma^{-1}$ then the neighborhood of X_1 is

$$\text{ne}(1) = \{i \in V \backslash \{1\} : K_{1i} \neq 0\} = \{2, 3, 4\}$$

- but the regression coefficient is $\beta_{13} = -K_{13}/K_{11}$ [Lauritzen, 1996, chap. 5], and so

$$\text{ne}(1) = \{i \in V \backslash \{1\} : \beta_{1i} \neq 0\} = \{2, 3, 4\}$$

nodewise estimation $n > p$



- estimate regression coefficients β_{ij} with LS

$$L_\beta := \|Y - X\beta\|_2^2 = (Y - X\beta)'(Y - X\beta)$$

- do this for each node separately and combine coefficients (and or or)

algorithm Meinshausen-Bühlmann

Algorithm 1 network $\hat{\Gamma}$ with MB lasso

1: **for** $i \in V$ **do**

2: split data Z in $Y = Z_i$ and $X = Z_{V \setminus i}$

3: obtain lasso $\hat{\beta}_L$ for $Y = X\beta + \varepsilon$

4: let $\hat{\tau}_i^2 = \|Y - X\hat{\beta}_L\|_2^2 + \lambda\|\hat{\beta}_L\|_1$ and

$$\hat{\gamma}_{ij} = -\hat{\beta}_{Lj}/\hat{\tau}_i^2 \qquad \text{for } i \neq j$$
$$\hat{\gamma}_{ij} = 1/\hat{\tau}_i^2 \qquad \text{for } i = j$$

5: **end for**

6: output $\hat{\Gamma}$ (perhaps using and or or rule)

relevance of selection/regularisation

linear regression model

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- increase prediction accuracy (decrease prediction error) by selecting a subset of predictors
- increase model interpretability by selecting only 'relevant' (correlated to Y) predictors

- Gaussian graphical models

1. stepwise procedures
   - problem
2. regularization
   - lasso variants
3. desparsified lasso

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: for each number of predictors k, test $\binom{p}{k}$ models, select the best from all

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: for each number of predictors k, test $\binom{p}{k}$ models, select the best from all
- forward selection: start with no predictors, add the best predictor one at a time

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: for each number of predictors k, test $\binom{p}{k}$ models, select the best from all
- forward selection: start with no predictors, add the best predictor one at a time
- backward selection: start with all predictors, remove the worst predictor one at a time

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: complexity is 2^p, e.g., 20 predictors gives 1.048.576 models

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: complexity is 2^p, e.g., 20 predictors gives 1.048.576 models
- forward selection: complexity is $p(p+1)/2 + 1$, e.g., 20 predictors gives 211 models

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: complexity is 2^p, e.g., 20 predictors gives 1.048.576 models
- forward selection: complexity is $p(p+1)/2 + 1$, e.g., 20 predictors gives 211 models
- backward selection: complexity is $p(p+1)/2 + 1$

subset selection

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon$$

- best subset selection: complexity is 2^p, e.g., 20 predictors gives 1.048.576 models
- forward selection: complexity is $p(p+1)/2 + 1$, e.g., 20 predictors gives 211 models
- backward selection: complexity is $p(p+1)/2 + 1$
- no guarantees that the true model is selected (asymptotically)

problem when $p > n$

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon = X\beta_0 + \varepsilon$$

problem when $p > n$

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon = X\beta_0 + \varepsilon$$

problem when $p > n$

- design matrix X is not of full rank

problem when $p > n$

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon = X\beta_0 + \varepsilon$$

problem when $p > n$

$$Y = X(\beta_0 + u) + \varepsilon$$

- design matrix X is not of full rank
- where u is in the null-space of X, i.e.

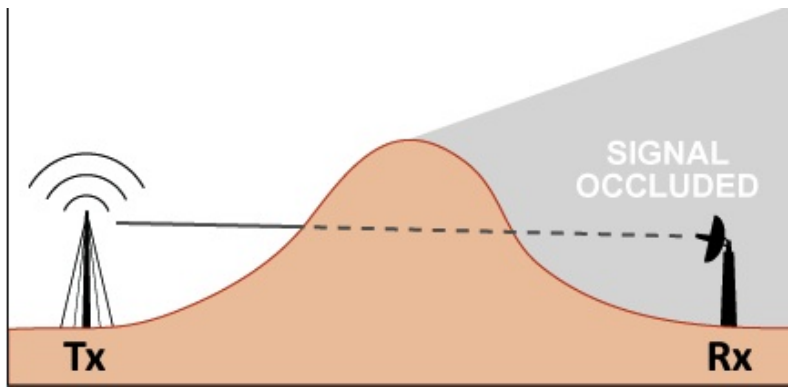
$$\text{null}(X) = \{u \in \mathbb{R}^p : Xu = 0\}$$

problem when $p > n$

$$Y = \beta_{0,0} + X_1\beta_{0,1} + X_2\beta_{0,2} + \cdots + X_p\beta_{0,p} + \epsilon = X\beta_0 + \varepsilon$$

problem when $p > n$

$$Y = X(\beta_0 + u) + \varepsilon$$



- design matrix X is not of full rank
- where u is in the null-space of X, i.e.

$$\text{null}(X) = \{u \in \mathbb{R}^p : Xu = 0\}$$

solution when $p > n$

penalized LS

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

solution when $p > n$

penalized LS

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge ℓ_2
ψ	$\sum_{i=1} \beta_i^2$
bias	all β_i
treats βs	unequally

solution when $p > n$

penalized LS

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge ℓ_2	lasso ℓ_1
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$
bias	all β_i	small β_i
treats βs	unequally	equally

solution when $p > n$

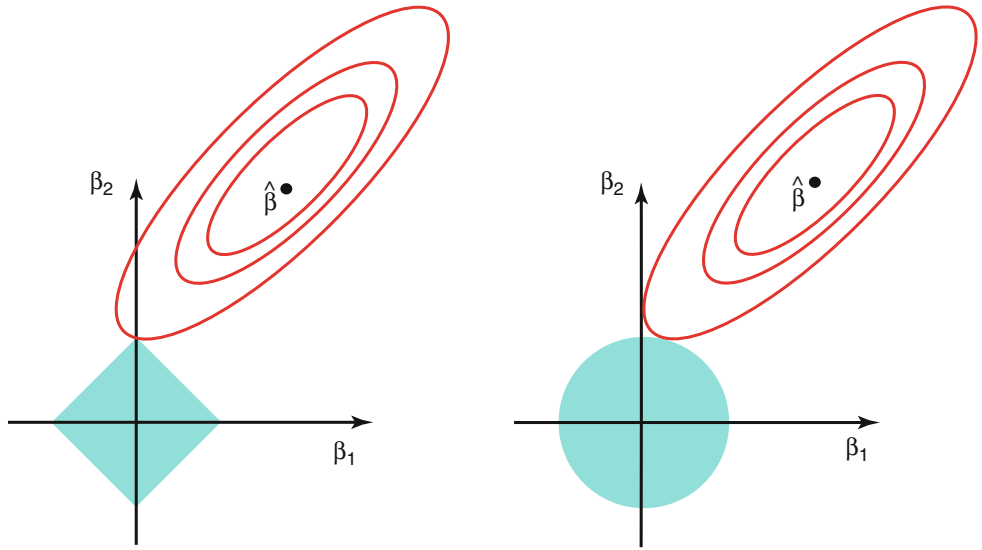
penalized LS

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge ℓ_2	lasso ℓ_1	ℓ_0
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$	p
bias	all β_i	small β_i	all β_i
treats βs	unequally	equally	equally

solution when $p > n$



types of penalization

	ridge ℓ_2	lasso ℓ_1	ℓ_0
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$	p
bias	all β_i	small β_i	all β_i
treats βs	unequally	equally	equally

adaptive lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda \sum_i \frac{|\beta_i|}{|\hat{\beta}_i^{\text{init}}|}}_{\text{penalty}}$$

where β_j^{init} is an initial estimate (prediction, with CV λ).

[Zou, 2006]

adaptive lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda \sum_i \frac{|\beta_i|}{|\hat{\beta}_i^{\text{init}}|}}_{\text{penalty}}$$

where β_j^{init} is an initial estimate (prediction, with CV λ).

properties of adaptive lasso

- when $\hat{\beta}_i^{\text{init}} = 0$ then $\hat{\beta}_j^{\text{ad}} = 0$

[Zou, 2006]

adaptive lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda \sum_i \frac{|\beta_i|}{|\hat{\beta}_i^{\text{init}}|}}_{\text{penalty}}$$

where β_j^{init} is an initial estimate (prediction, with CV λ).

properties of adaptive lasso

- when $\hat{\beta}_i^{\text{init}} = 0$ then $\hat{\beta}_j^{\text{ad}} = 0$
- when $\hat{\beta}_i^{\text{init}}$ is large, $\hat{\beta}_j^{\text{ad}}$ has small penalty, resulting in less bias

[Zou, 2006]

thresholded lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

and select for some constant $c > 0$

$$\hat{\beta}_j 1\{\hat{\beta}_j > c\}$$

[Bühlmann and van der Geer, 2011]

thresholded lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

and select for some constant $c > 0$

$$\hat{\beta}_j 1\{\hat{\beta}_j > c\}$$

properties of thresholded lasso

- screening property $S_0 \subseteq \hat{S}$

[Bühlmann and van der Geer, 2011]

thresholded lasso

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

and select for some constant $c > 0$

$$\hat{\beta}_j 1\{\hat{\beta}_j > c\}$$

properties of thresholded lasso

- screening property $S_0 \subseteq \hat{S}$
- more 'accurate' than adaptive lasso (smaller predictive and estimation error)

[Bühlmann and van der Geer, 2011]

scaled lasso

iterate as long as $L_{\hat{\beta}(\hat{\sigma}\lambda_0)} \leq L_{\hat{\beta},\lambda}$

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

$$\hat{\beta} = \hat{\beta}(\hat{\sigma}\lambda_0) \text{ and } \hat{\sigma} = \text{LS}^{1/2}/n$$

[Sun and Zhang, 2012]

scaled lasso

iterate as long as $L_{\hat{\beta}(\hat{\sigma}\lambda_0)} \leq L_{\hat{\beta},\lambda}$

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

$$\hat{\beta} = \hat{\beta}(\hat{\sigma}\lambda_0) \text{ and } \hat{\sigma} = \text{LS}^{1/2}/n$$

properties of scaled lasso

- screening property $S_0 \subseteq \hat{S}$

[Sun and Zhang, 2012]

scaled lasso

iterate as long as $L_{\hat{\beta}(\hat{\sigma}\lambda_0)} \leq L_{\hat{\beta},\lambda}$

$$L_{\beta,\lambda} = \underbrace{(Y - X\beta)^{\mathsf{T}}(Y - X\beta)}_{\text{LS fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

$$\hat{\beta} = \hat{\beta}(\hat{\sigma}\lambda_0) \text{ and } \hat{\sigma} = \text{LS}^{1/2}/n$$

properties of scaled lasso

- screening property $S_0 \subseteq \hat{S}$
- selects automatically a value for λ depending on the noise level $\hat{\sigma}$

[Sun and Zhang, 2012]

glasso

obtain inverse covariance matrix $K = \Sigma^{-1}$ from

$$L_{\beta,\lambda} = \log|K| - \mathrm{tr}S\,K - \lambda \sum_{i<j} |K_{ij}|$$

where S is sample covariance

[Friedman et al., 2008]

glasso

obtain inverse covariance matrix $K = \Sigma^{-1}$ from

$$L_{\beta,\lambda} = \log|K| - \mathrm{tr}SK - \lambda \sum_{i<j} |K_{ij}|$$

where S is sample covariance

properties of glasso

- screening property $S_0 \subseteq \hat{S}$

[Friedman et al., 2008]

glasso

obtain inverse covariance matrix $K = \Sigma^{-1}$ from

$$L_{\beta,\lambda} = \log|K| - \mathrm{tr}S\,K - \lambda \sum_{i<j} |K_{ij}|$$

where S is sample covariance

properties of glasso

- screening property $S_0 \subseteq \hat{S}$
- more restrictive than MB approach

[Friedman et al., 2008]

glasso

obtain inverse covariance matrix $K = \Sigma^{-1}$ from

$$L_{\beta,\lambda} = \log|K| - \mathrm{tr}SK - \lambda \sum_{i<j} |K_{ij}|$$

where S is sample covariance

properties of glasso

- screening property $S_0 \subseteq \hat{S}$
- more restrictive than MB approach
- more accurate that MB apraoch (slightly)

[Friedman et al., 2008]

dependent measures of effect

dependent measures of effect

correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

dependent measures of effect

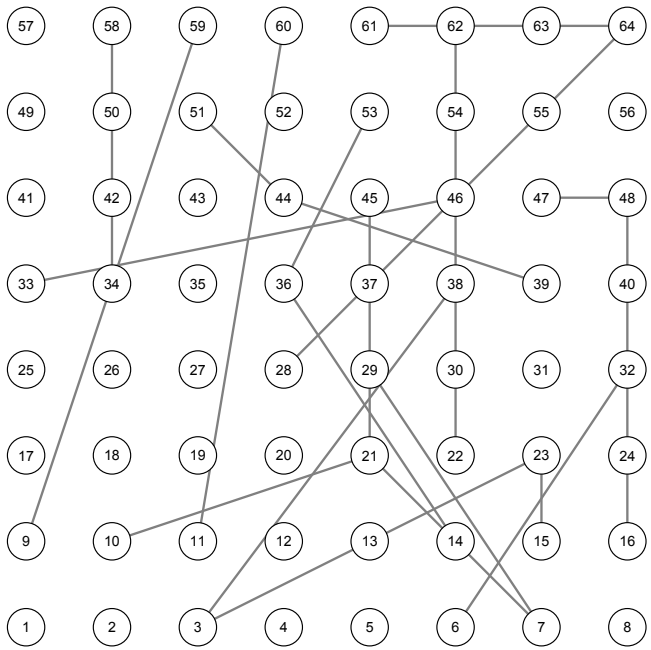
correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

correct rejections/rejections (positive predictive value)

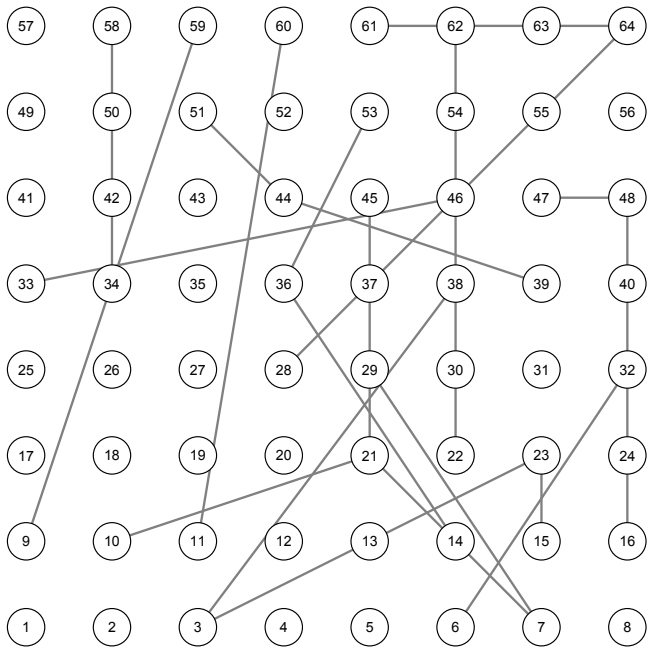
$$\text{precision} := \frac{|\hat{E} \cap E_0|}{|\hat{E}|}$$

ER graph

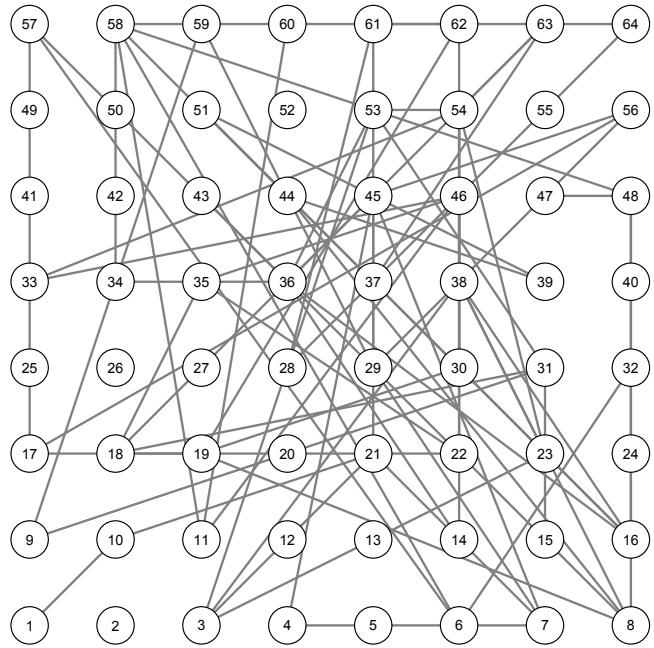


true graph

ER graph

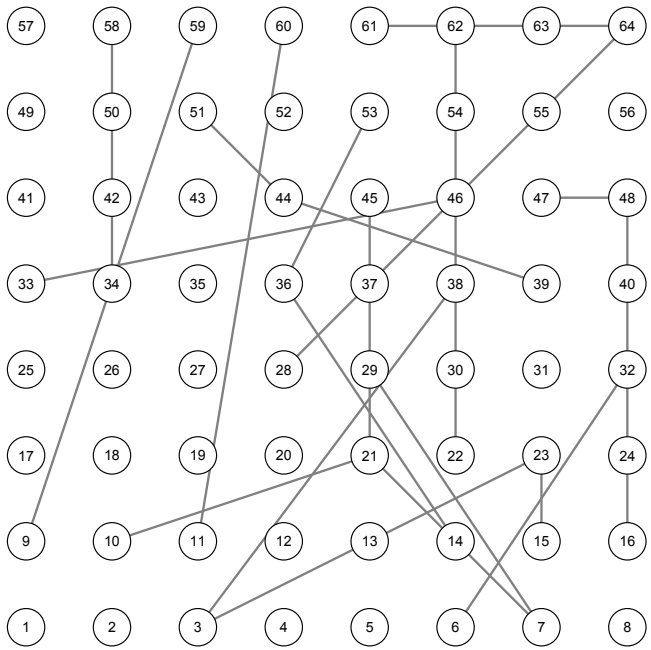


true graph

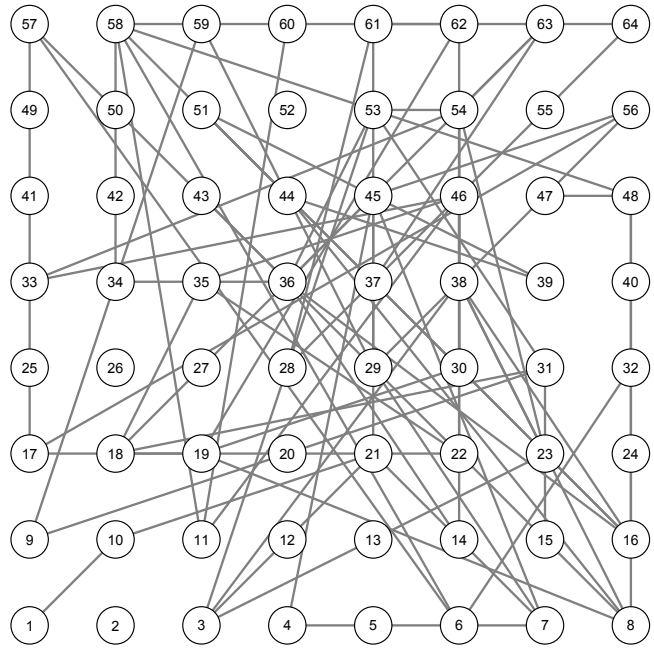


MB lasso

ER graph



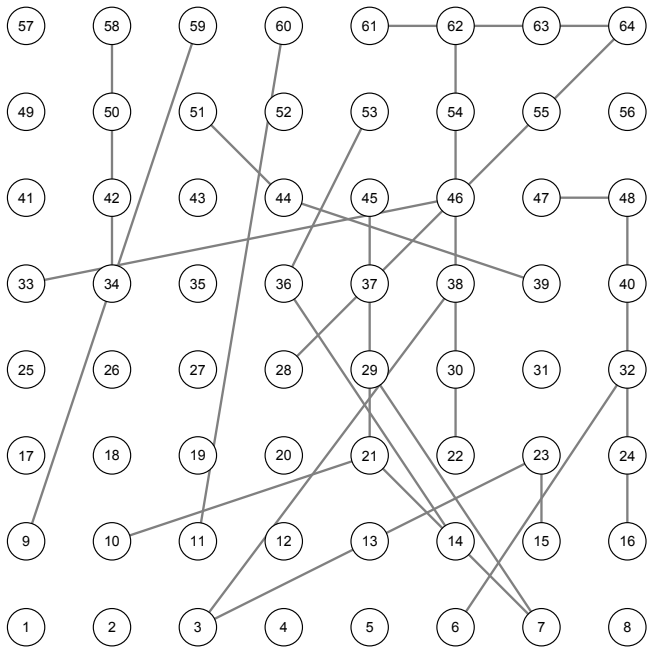
true graph



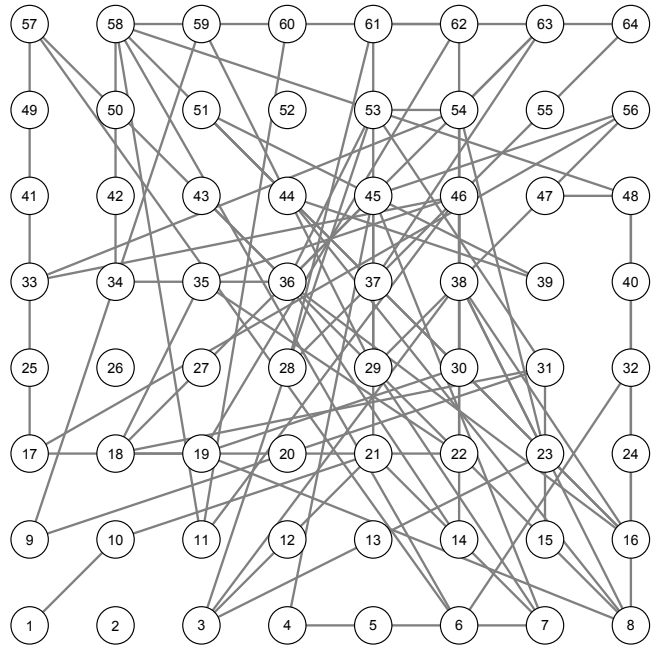
MB lasso

precision	recall	rejection rate	density
0.26	0.98	0.077	0.010

ER graph



true graph

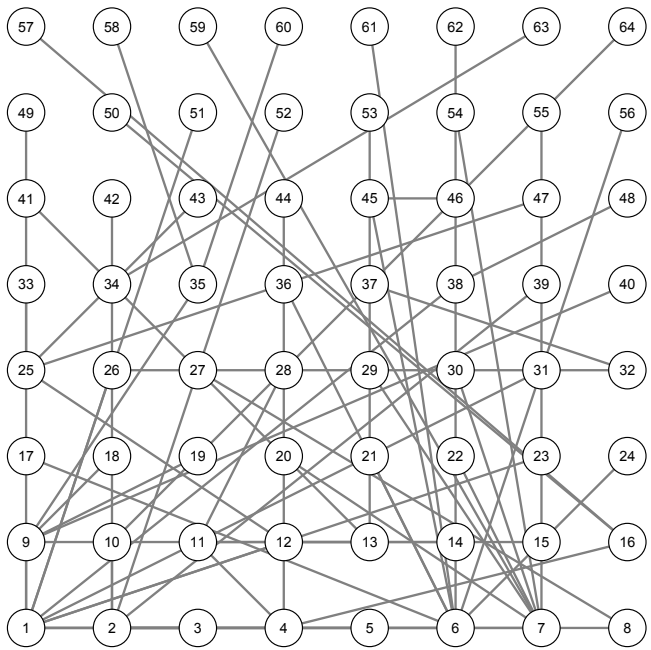


MB lasso

precision	recall	rejection rate	density
0.26	0.98	0.077	0.010

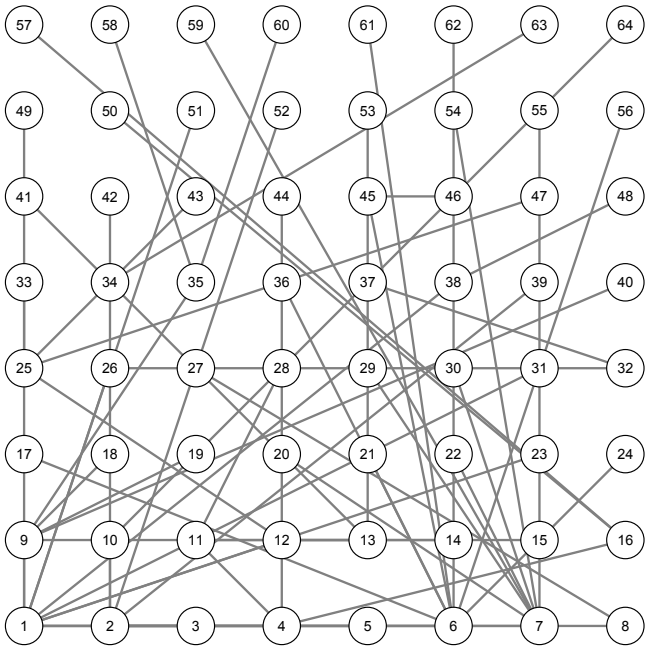
- often with lasso $S_0 \subseteq S(\hat{\beta}_L)$ when penalty is 'small'
  [see Bühlmann & van de Geer, 2011]

BA graph

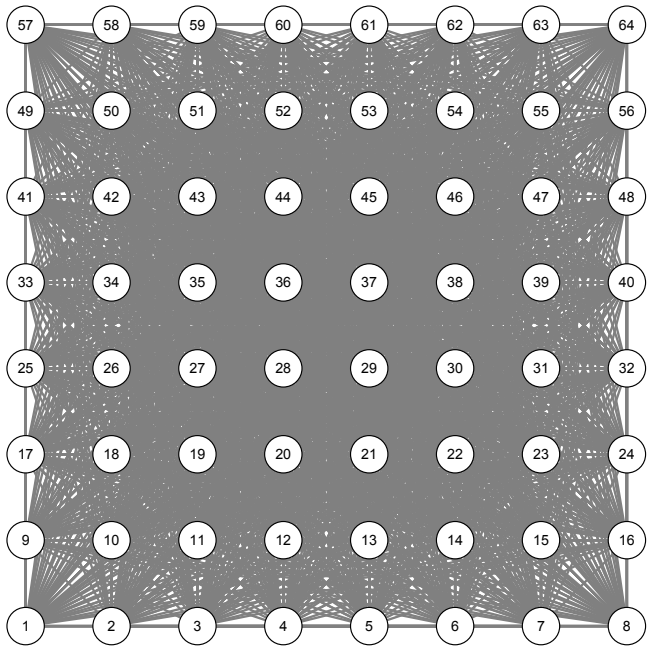


true graph

BA graph

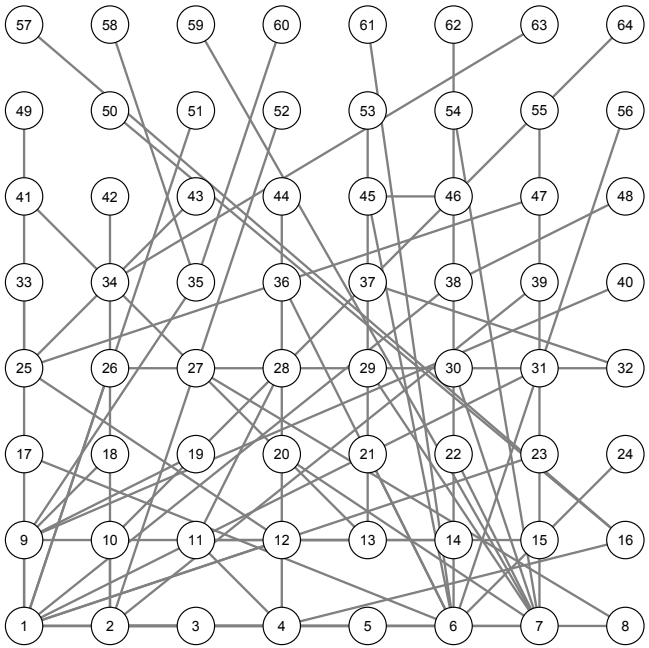


true graph

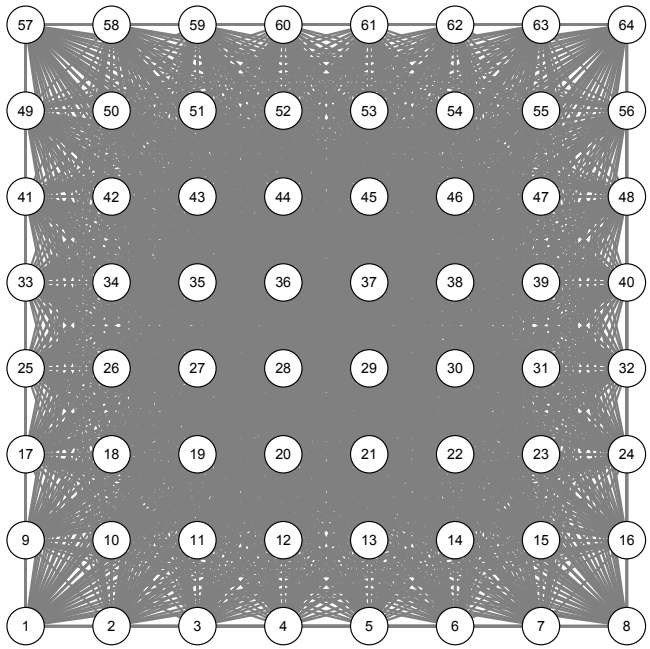


MB lasso

BA graph



true graph



MB lasso

precision	recall	rejection rate	density
0.032	1.00	0.99	0.0625

points of view

points of view

- prediction squared prediction loss (est – truth)^2

$$\mathbb{E}[(\hat{\beta} - \beta_0)^\mathsf{T} X_{\text{new}}^\mathsf{T} X_{\text{new}} (\hat{\beta} - \beta_0) \mid X]$$

points of view

- prediction squared prediction loss (est – truth)^2

$$\mathbb{E}[(\hat{\beta} - \beta_0)^\mathsf{T} X_{\text{new}}^\mathsf{T} X_{\text{new}} (\hat{\beta} - \beta_0) \mid X]$$

- variable screening estimated variables in $\hat{S} = \{j : \hat{\beta}_j \neq 0\}$ wrt true $S_0 = \{j : \beta_{0j} \neq 0\}$

$$\mathbb{P}[S_0 \subseteq \hat{S}] \to 1$$

points of view

- prediction squared prediction loss (est – truth)^2

$$\mathbb{E}[(\hat{\beta} - \beta_0)^\mathsf{T} X_{\text{new}}^\mathsf{T} X_{\text{new}} (\hat{\beta} - \beta_0) \mid X]$$

- variable screening estimated variables in $\hat{S} = \{j : \hat{\beta}_j \neq 0\}$ wrt true $S_0 = \{j : \beta_{0j} \neq 0\}$

$$\mathbb{P}[S_0 \subseteq \hat{S}] \to 1$$

allow for false positives, no false negatives

points of view

- prediction squared prediction loss (est – truth)^2

$$\mathbb{E}[(\hat{\beta} - \beta_0)^\mathsf{T} X_{\text{new}}^\mathsf{T} X_{\text{new}} (\hat{\beta} - \beta_0) \mid X]$$

- variable screening estimated variables in $\hat{S} = \{j : \hat{\beta}_j \neq 0\}$ wrt true $S_0 = \{j : \beta_{0j} \neq 0\}$

$$\mathbb{P}[S_0 \subseteq \hat{S}] \to 1$$

  allow for false positives, no false negatives

- variable selection

$$\mathbb{P}[S_0 = \hat{S}] \to 1$$

assumptions lasso

assumptions lasso

assumptions for variable selection

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

assumptions lasso

assumptions for variable selection

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

(b) consistency tuning λ, no collinearity in true set of variables S_0

assumptions lasso

assumptions for variable selection

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

(b) consistency tuning λ, no collinearity in true set of variables S_0

(c) beta-min the signal (edge weights) cannot be too small

$$\min_{j \in E} |\beta_{0j}| > c$$

today

- Gaussian graphical models

1. stepwise procedures
   - problem
2. regularization
   - lasso variants
3. desparsified lasso

desparsified lasso

the lasso estimate $\hat{\beta}_L$ is obtained by minimizing

$$L_{\beta,\lambda} = (Y - X\beta)^{\mathsf{T}}(Y - X\beta) + \lambda \sum_i |\beta_i|$$

desparsified lasso

the lasso estimate $\hat{\beta}_L$ is obtained by minimizing

$$L_{\beta,\lambda} = (Y - X\beta)^{\mathsf{T}}(Y - X\beta) + \lambda \sum_i |\beta_i|$$

Properties lasso estimate $\hat{\beta}_L$

$$\mathbb{E}(\hat{\beta}_L) = \beta_0 - \text{bias}$$

desparsified lasso

the lasso estimate $\hat{\beta}_L$ is obtained by minimizing

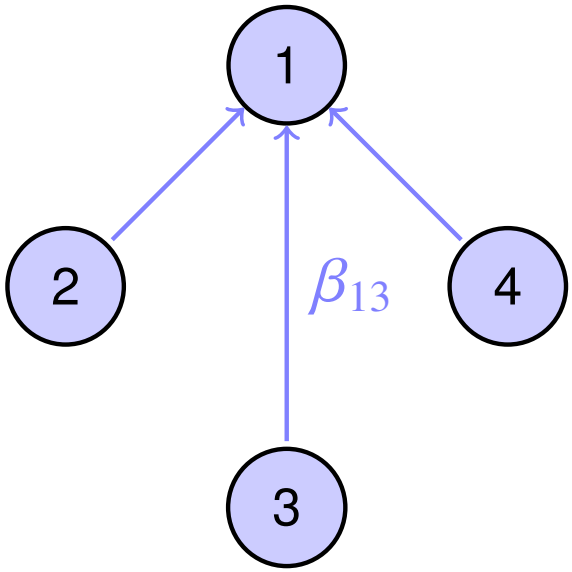
$$L_{\beta,\lambda} = (Y - X\beta)^{\mathsf{T}}(Y - X\beta) + \lambda \sum_i |\beta_i|$$

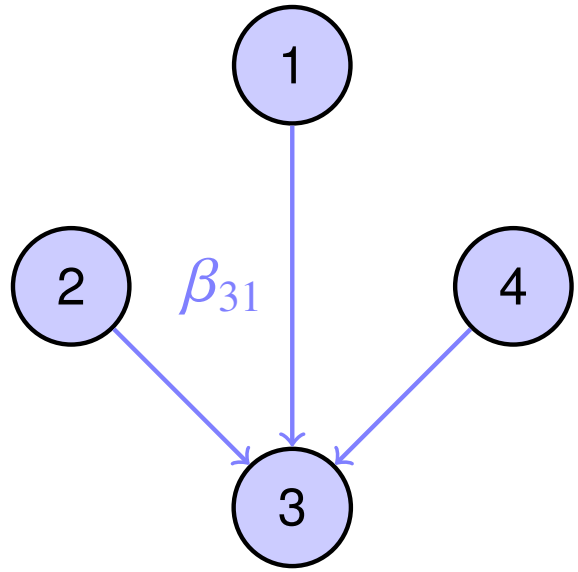
Properties lasso estimate $\hat{\beta}_L$

$$\mathbb{E}(\hat{\beta}_L) = \beta_0 - \text{bias}$$

$$\text{bias} = \hat{\Theta} X^{\mathsf{T}}(Y - X\hat{\beta}_L)/n$$

Gaussian graphical models





desparsified (debiased) lasso

$$\hat{\beta}_{dL} = \underbrace{\hat{\beta}_L}_{\text{normal lasso}} + \underbrace{\hat{\Theta} X^{\mathsf{T}} (Y - X\hat{\beta}_L)/n}_{\text{debias part}}$$

[van der Geer et al., 2013, Javanmard & Montanari, 2013]

desparsified lasso

$\hat{\Theta}$ is a 'relaxed' inverse of $X^\top X/n$, such that

$$\hat{\beta}_{dL} = \underbrace{\hat{\beta}_L}_{\text{normal lasso}} + \underbrace{\hat{\Theta} X^\top (Y - X\hat{\beta}_L)/n}_{\text{debias part}} - \Delta$$

[van de Geer et al., 2013, Javanmard & Montanari, 2013]

desparsified lasso

$\hat{\Theta}$ is a 'relaxed' inverse of $X^\top X/n$, such that

$$\hat{\beta}_{dL} = \underbrace{\hat{\beta}_L}_{\text{normal lasso}} + \underbrace{\hat{\Theta} X^\top (Y - X\hat{\beta}_L)/n}_{\text{debias part}} - \Delta$$

$$\Delta = \sqrt{n}(\hat{\Theta}(X^\top X/n) - I)(\hat{\beta}_L - \beta) \quad \text{(small)}$$

desparsified lasso

- is asymptotically unbiased

[van de Geer et al., 2013, Javanmard & Montanari, 2013]

desparsified lasso

$\hat{\Theta}$ is a 'relaxed' inverse of $X^\top X/n$, such that

$$\hat{\beta}_{dL} = \underbrace{\hat{\beta}_L}_{\text{normal lasso}} + \underbrace{\hat{\Theta} X^\top (Y - X\hat{\beta}_L)/n}_{\text{debias part}} - \Delta$$

$$\Delta = \sqrt{n}(\hat{\Theta}(X^\top X/n) - I)(\hat{\beta}_L - \beta) \quad \text{(small)}$$

desparsified lasso

- is asymptotically unbiased
- has standard, computable variance

[van de Geer et al., 2013, Javanmard & Montanari, 2013]

desparsified lasso

$\hat{\Theta}$ is a 'relaxed' inverse of $X^\mathsf{T}X/n$, such that

$$\hat{\beta}_{dL} = \underbrace{\hat{\beta}_L}_{\text{normal lasso}} + \underbrace{\hat{\Theta}X^\mathsf{T}(Y - X\hat{\beta}_L)/n}_{\text{debias part}} - \Delta$$

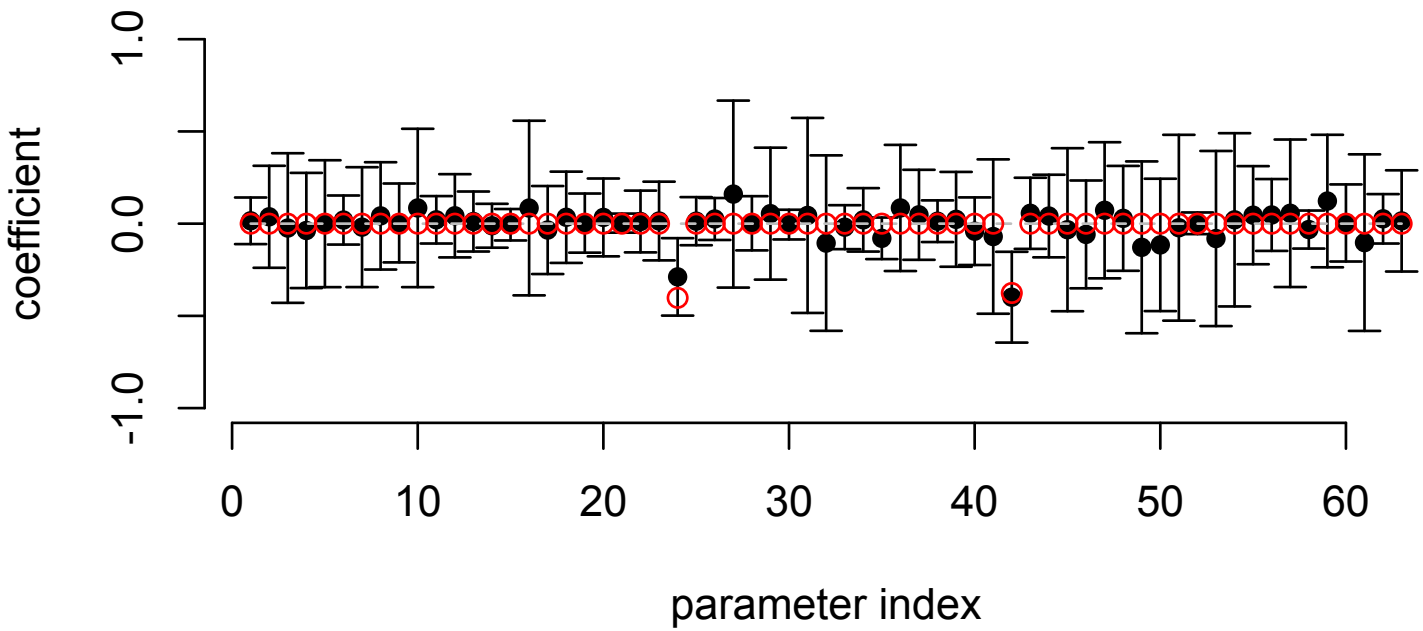
$$\Delta = \sqrt{n}(\hat{\Theta}(X^\mathsf{T}X/n) - I)(\hat{\beta}_L - \beta) \quad \text{(small)}$$

desparsified lasso

- is asymptotically unbiased
- has standard, computable variance
- is amenable to standard analysis

[van de Geer et al., 2013, Javanmard & Montanari, 2013]

nodewise desparsified lasso



tuning parameter λ estimated by 10-fold cv

estimate true value

simulations

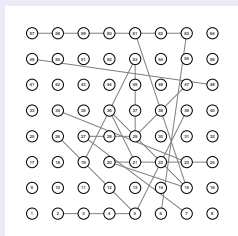
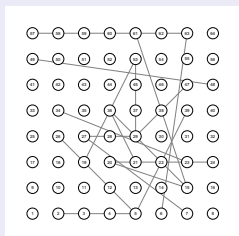
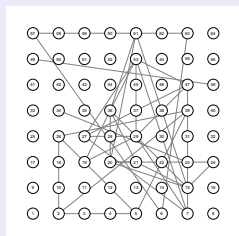
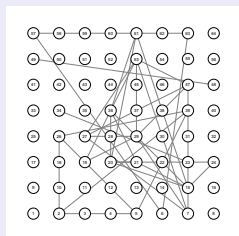
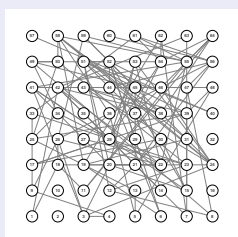
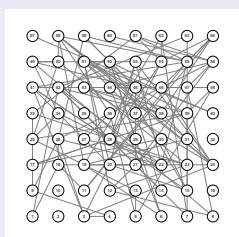
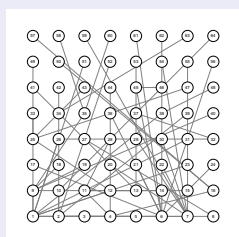
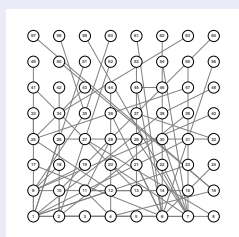
- generate data from different networks ER or BA

simulations

- generate data from different networks ER or BA
- estimate using desparsified lasso with MB or GL

simulations

- generate data from different networks ER or BA
- estimate using desparsified lasso with MB or GL

	ER		BA	
	MB	GL	MB	GL
sparse				
not sparse				

dependent measures of effect

dependent measures of effect

correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

dependent measures of effect

correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

correct rejections/rejections (positive predictive value)

$$\text{precision} := \frac{|\hat{E} \cap E_0|}{|\hat{E}|}$$

dependent measures of effect

correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

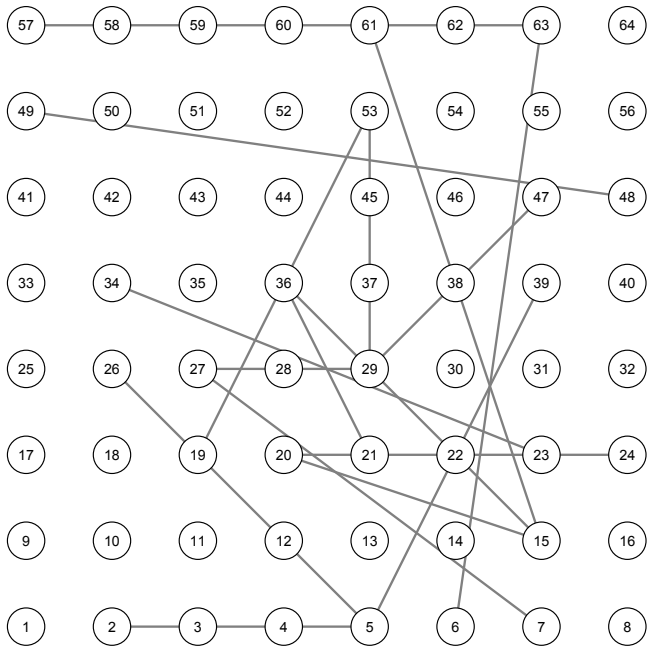
correct rejections/rejections (positive predictive value)

$$\text{precision} := \frac{|\hat{E} \cap E_0|}{|\hat{E}|}$$

probability of CI containing true parameter γ_{0ij}

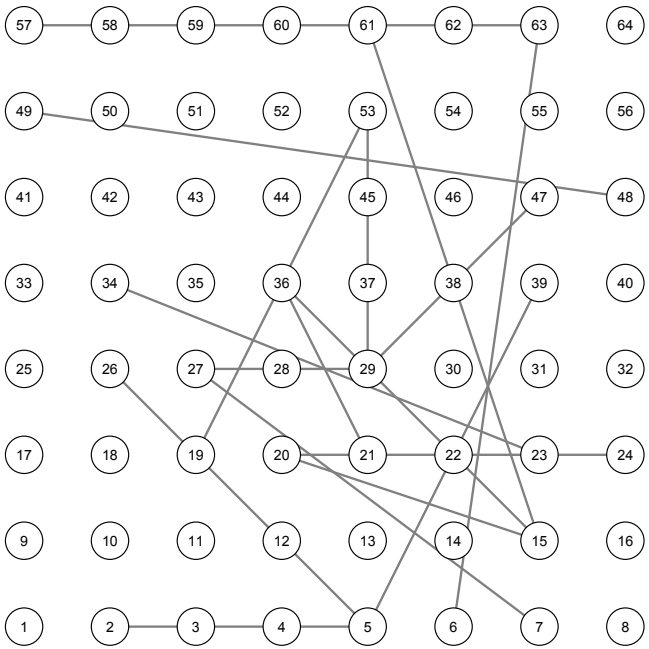
$$\text{coverage} := \frac{1}{|V|(|V|-1)} \sum_{i \neq j \in V} \mathbb{P}[\beta_{0ij} \in \text{CI}_{ij}]$$

ER graph

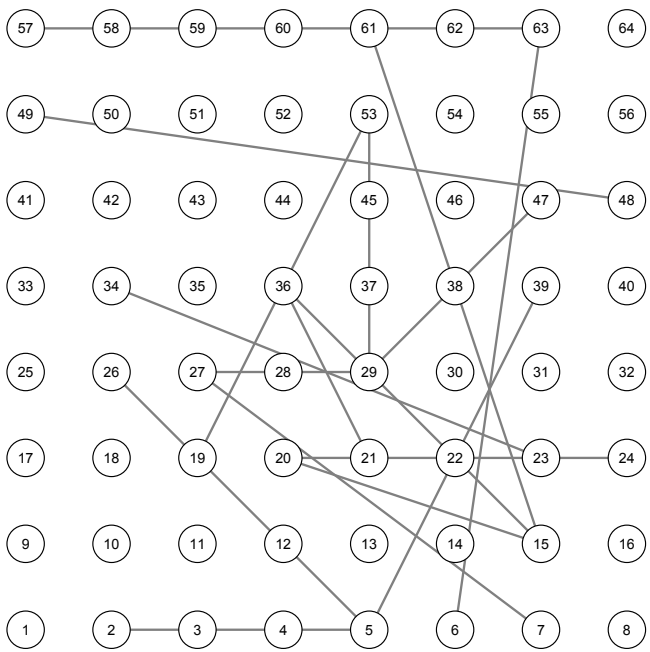


true graph

ER graph

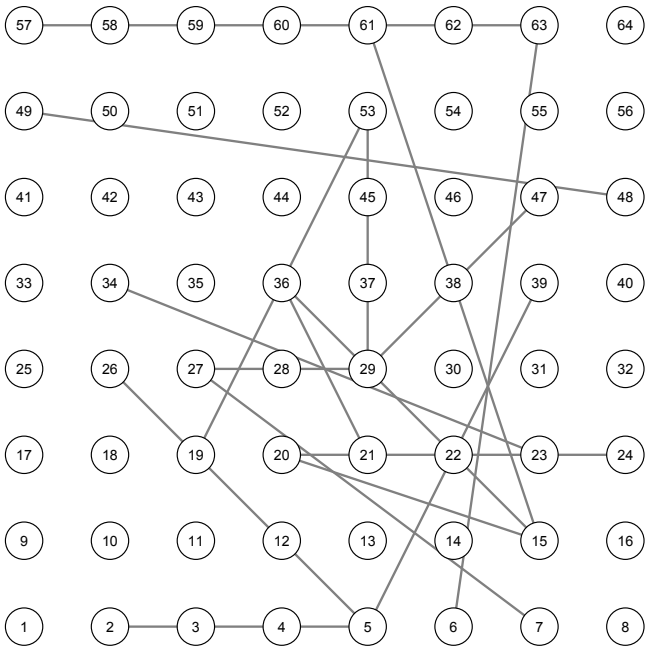


true graph

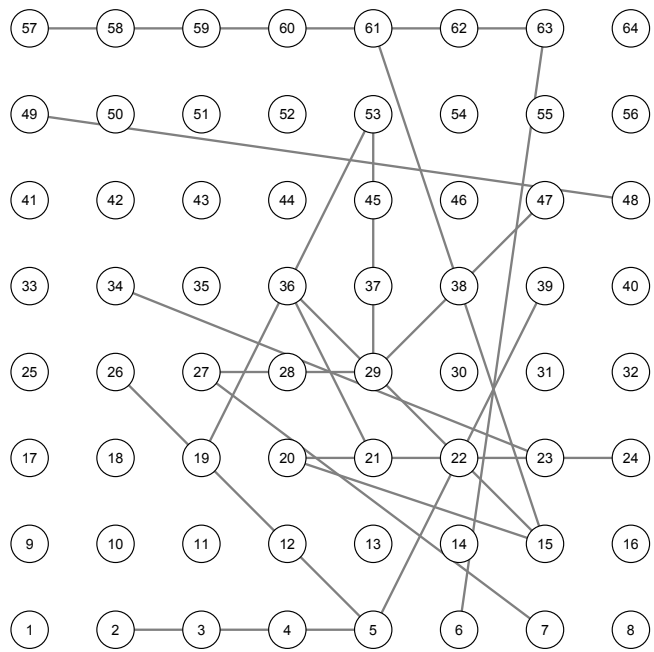


desparsified lasso

ER graph



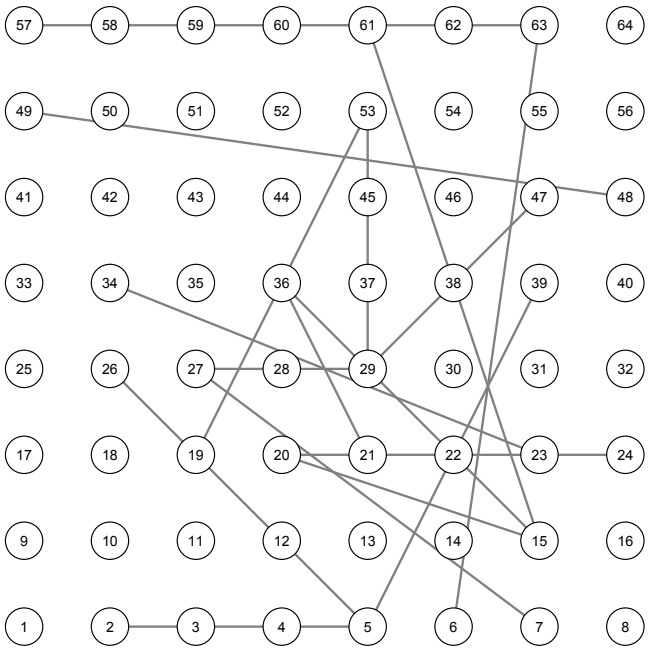
true graph



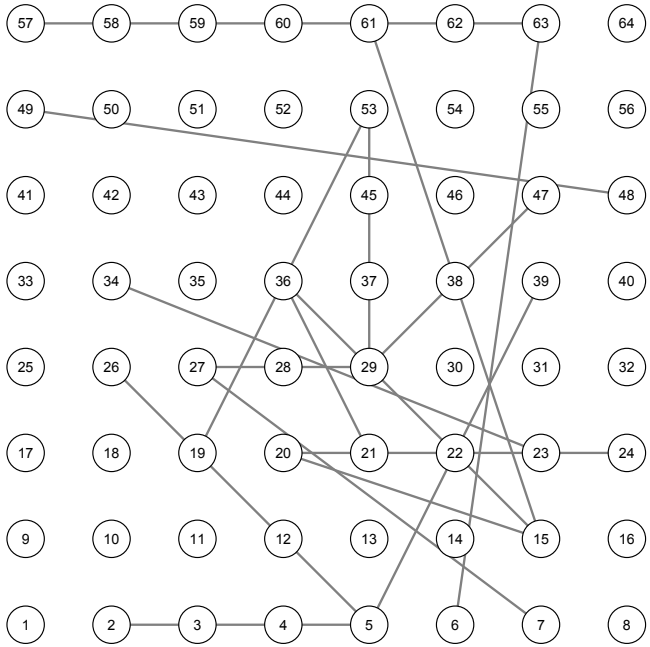
desparsified lasso

precision	recall	rejection rate	density
0.99	1.00	0.011	0.010

ER graph



true graph

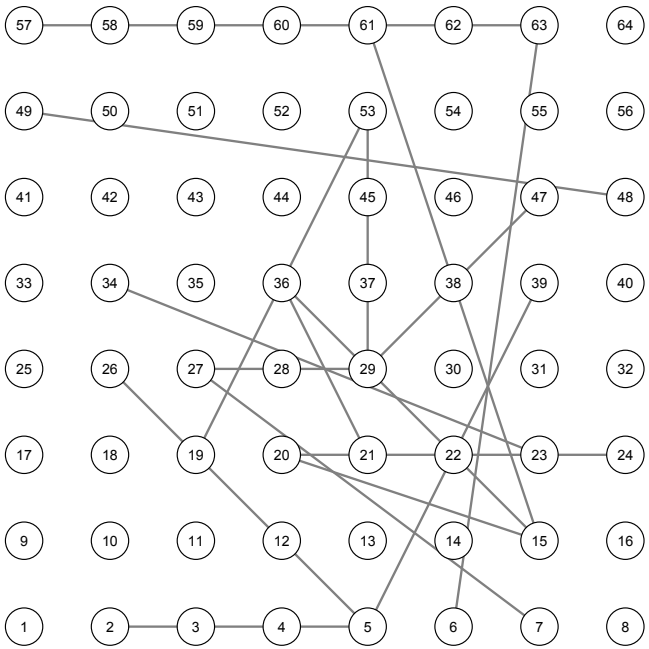


desparsified lasso

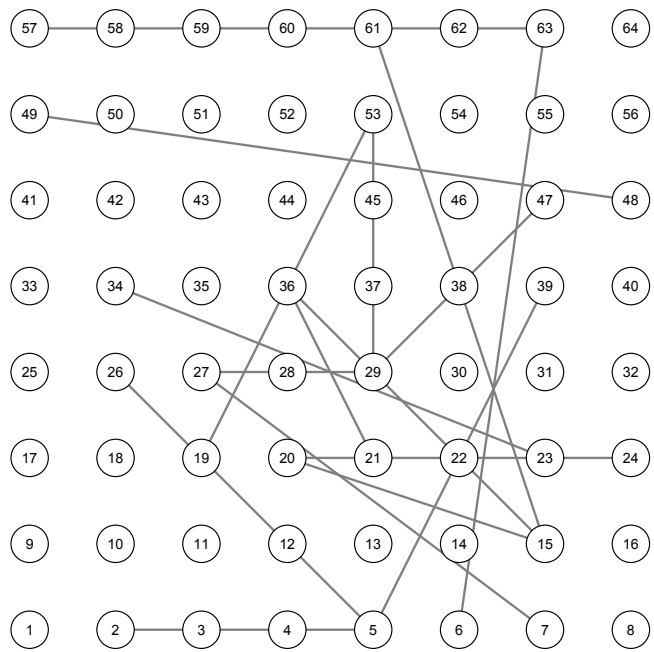
precision	recall	rejection rate	density
0.99	1.00	0.011	0.010

- sparsity $= \sqrt{n/\log p(p-1)} = 25$ and

ER graph



true graph

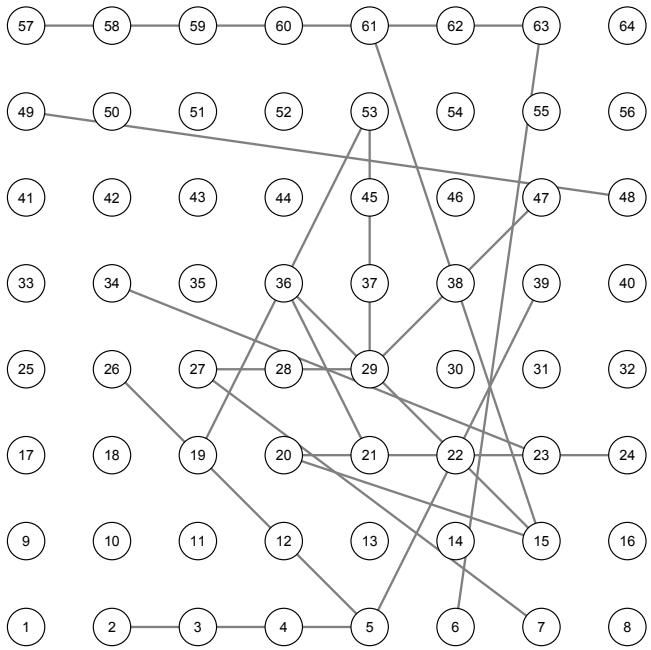


desparsified lasso

precision	recall	rejection rate	density
0.99	1.00	0.011	0.010

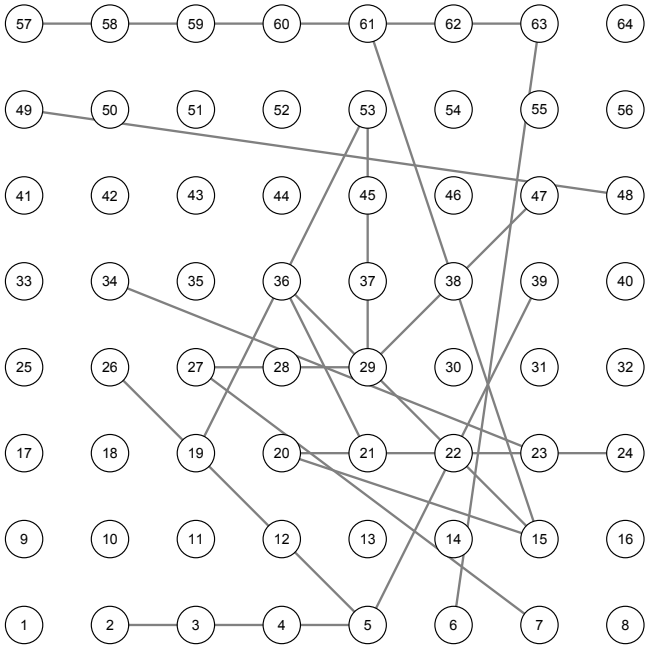
- sparsity $= \sqrt{n/\log p(p-1)} = 25$ and
- density $= 0.01\,p(p-1)/2 = 20$

ER graph

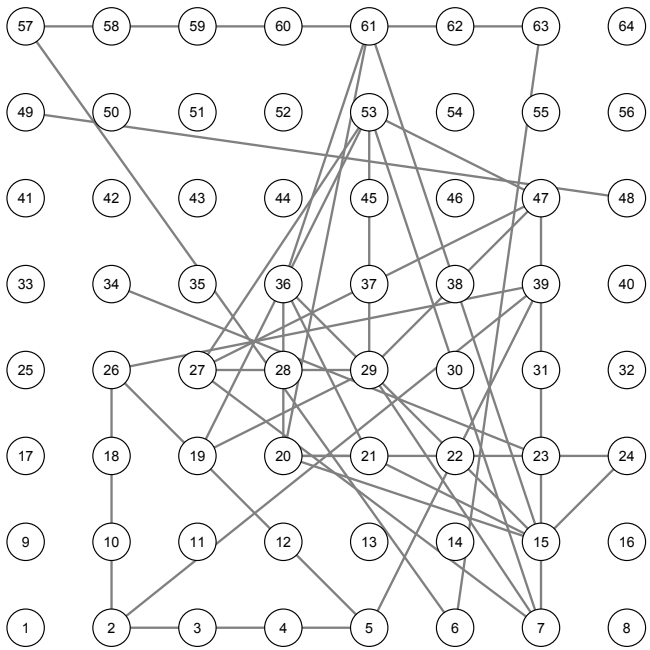


true graph

ER graph

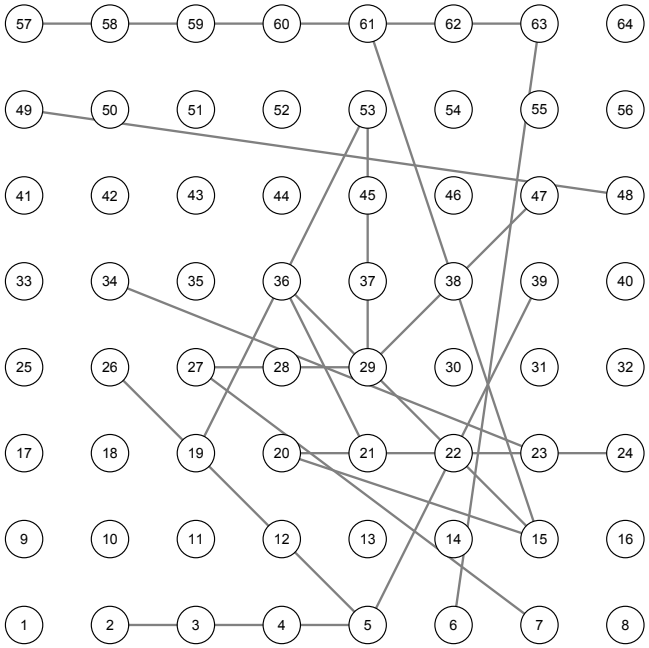


true graph

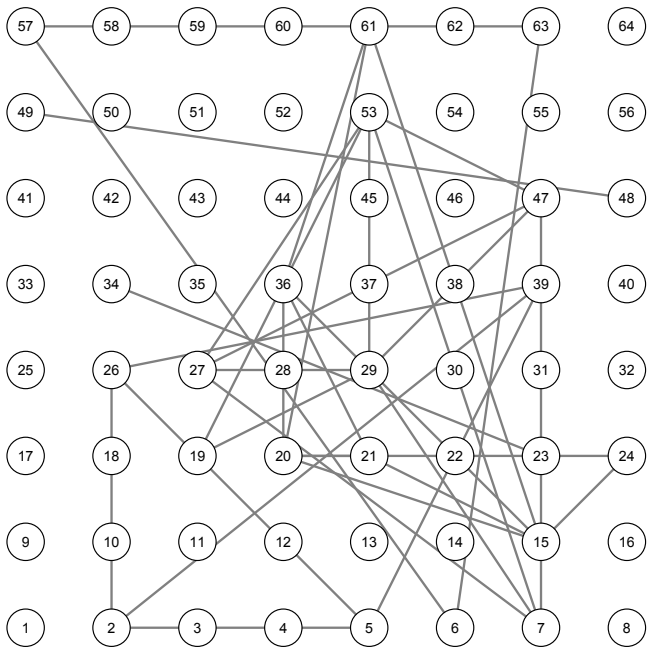


graphical lasso

ER graph



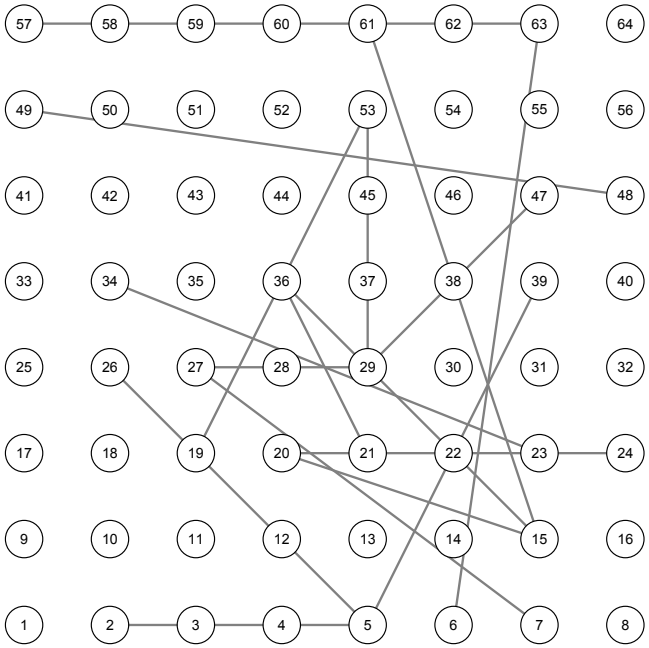
true graph



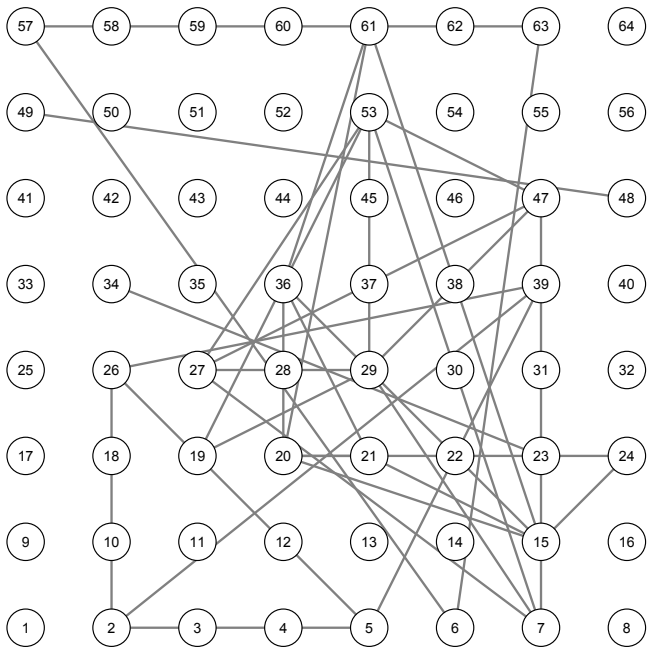
graphical lasso

precision	recall	rejection rate	density
0.50	1.00	0.011	0.010

ER graph



true graph

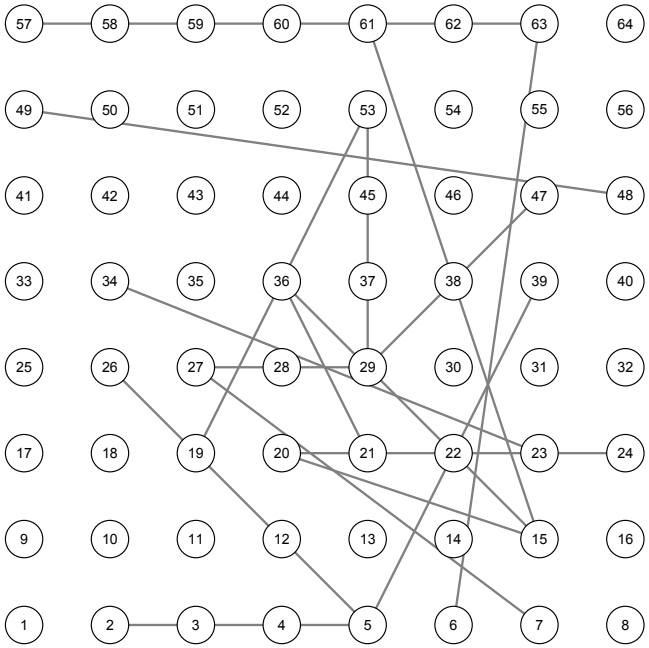


graphical lasso

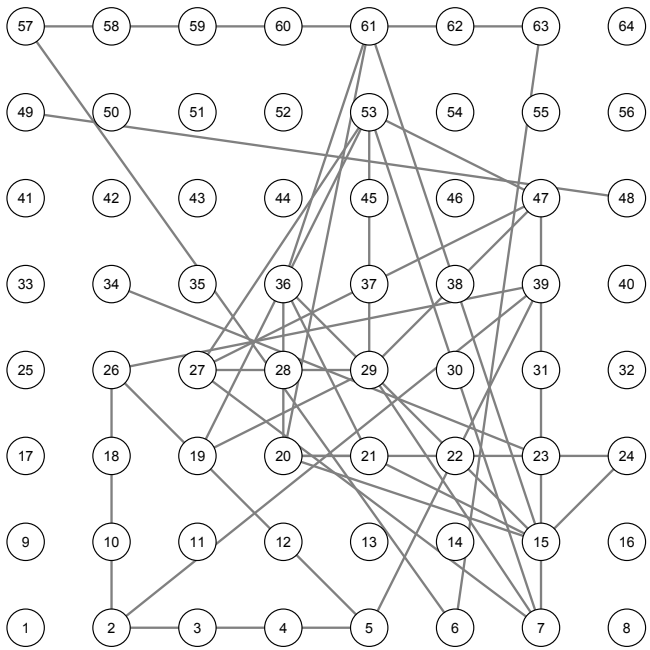
precision	recall	rejection rate	density
0.50	1.00	0.011	0.010

- sparsity $= \sqrt{n/\log p(p-1)} = 25$ and

ER graph



true graph



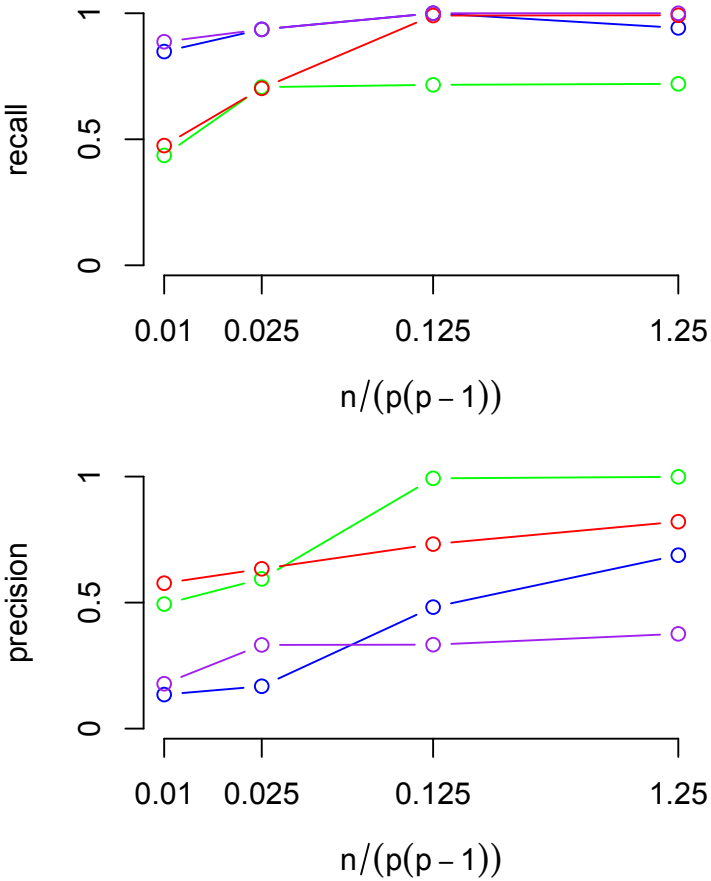
graphical lasso

precision	recall	rejection rate	density
0.50	1.00	0.011	0.010

- sparsity $= \sqrt{n/\log p(p-1)} = 25$ and
- density $= 0.01\,p(p-1)/2 = 20$

recall and precision for sparse graphs

ER



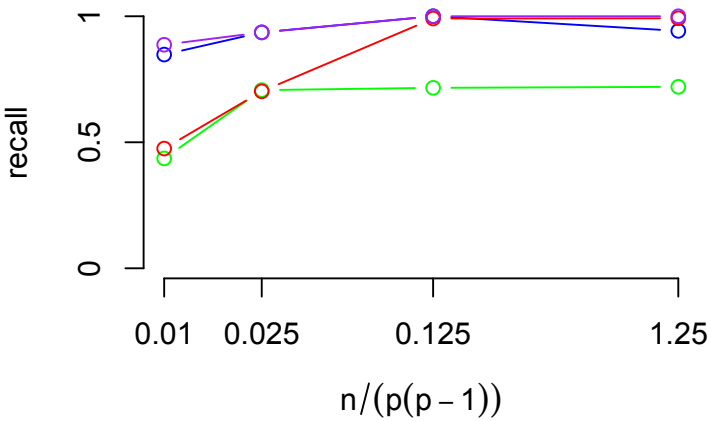
– MB – SL

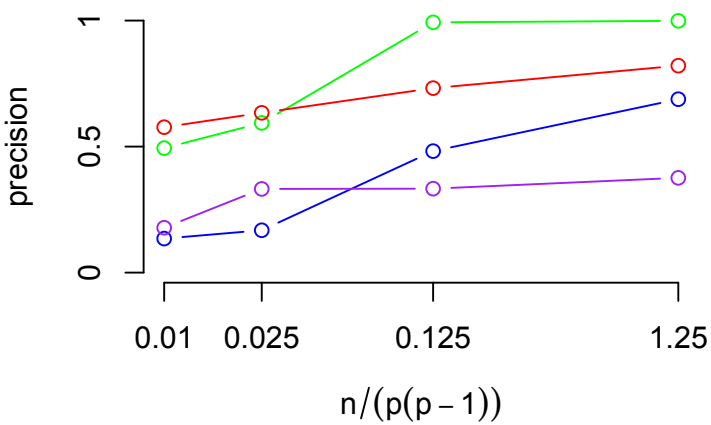
BA

– dL – GL

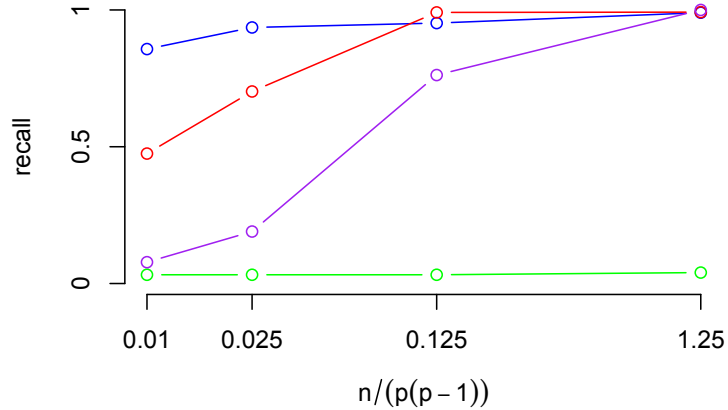
recall and precision for sparse graphs

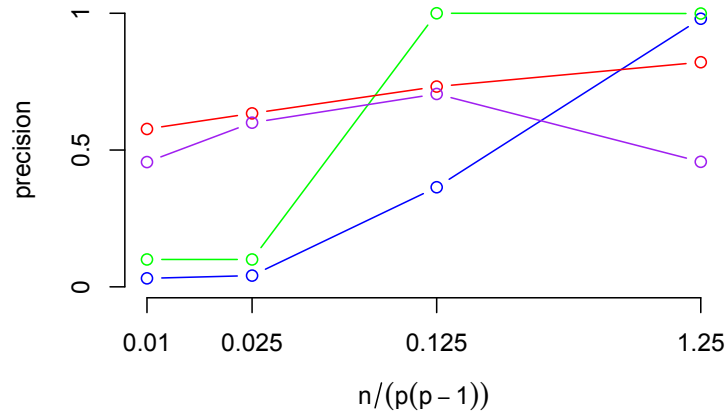
ER





BA

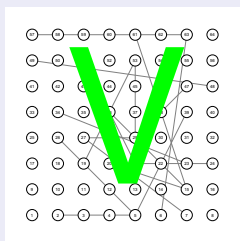
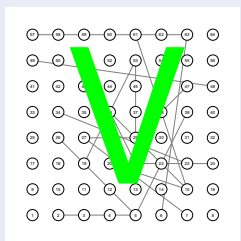
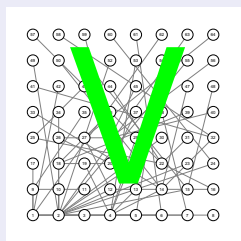
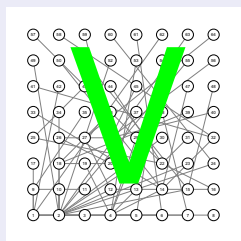
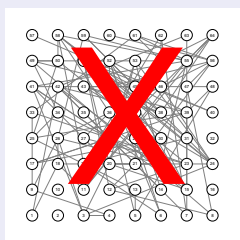
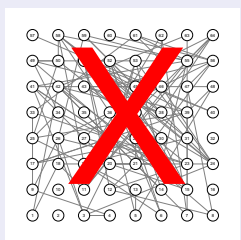
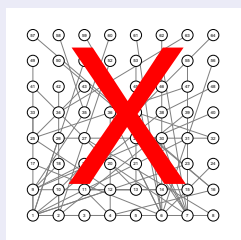
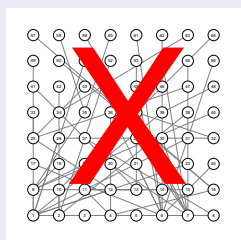




– MB – SL

– dL – GL

conclusions variable screening

	ER		BA	
	dL(MB)	other	dL(MB)	other
sparse				
not sparse				

coverage for sparse graphs

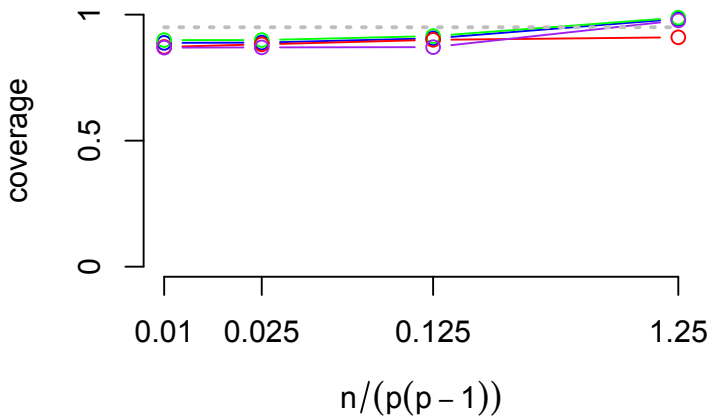
ER

– MB – SL

– dL – GL

coverage for sparse graphs

ER

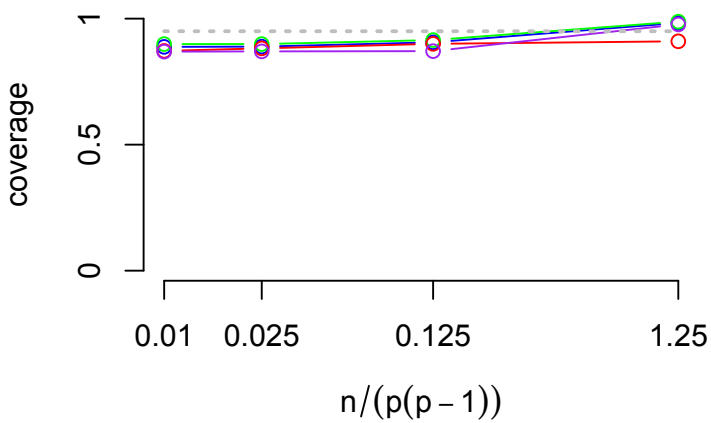


– MB – SL

– dL – GL

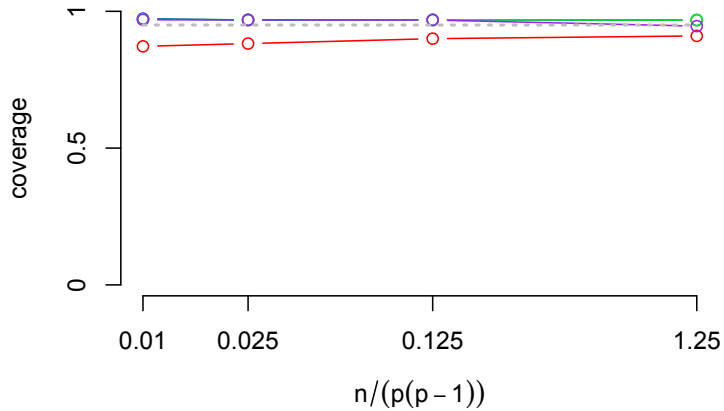
coverage for sparse graphs

ER



– MB – SL

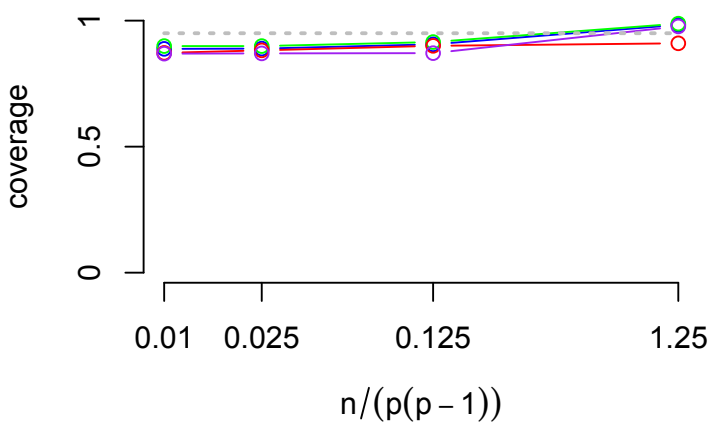
BA



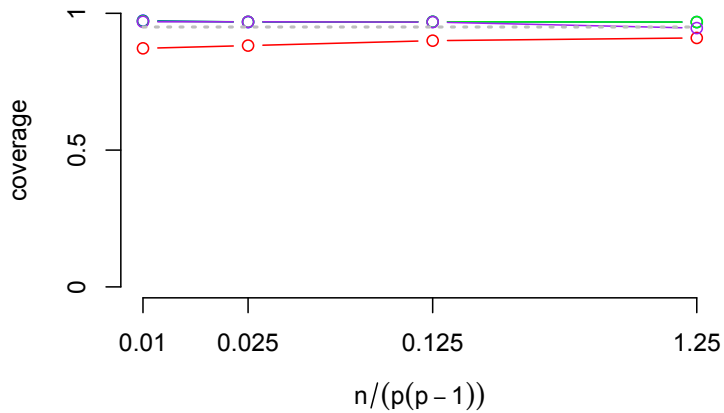
– dL – GL

coverage for sparse graphs

ER



BA

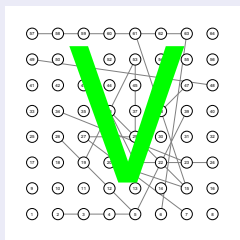
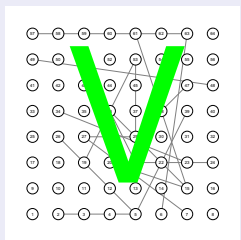
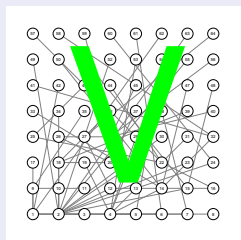
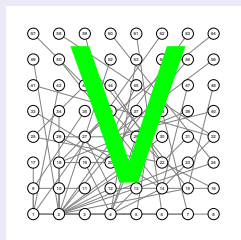


– MB – SL

– dL – GL

- coverage is close to nominal level (95%)

conclusions variable selection CI

	ER		BA	
	dL(MB)	dL(GL)	dL(MB)	dL(GL)
sparse				
not sparse