Gaussian graphical model

DSA5103 Lecture 10

Yangjing Zhang

23-Mar-2023

NUS

Today's content

1. Basics of graphical models
2. Gaussian graphical models and graphical Lasso
3. Neighbourhood selection
4. Applications

Basics of graphical models

Joint distribution

Toy example Given a coin (possibly biased). Toss the coin twice.

- x_1: H(head)/T(tail). x_2: H/T. x_3: F(fair coin)/B(biased coin)
- Joint distribution $p(x_1, x_2, x_3)$
- Conditional distribution $p(x_1, x_2 \mid x_3 = B)$

Table 1: Joint distribution

x_1	x_2	x_3	Probability
H	H	F	0.125
H	T	F	0.125
T	H	F	0.125
T	T	F	0.125
H	H	B	0.32
H	T	B	0.08
T	H	B	0.08
T	T	B	0.02

Joint distribution

Toy example Given a coin (possibly biased). Toss the coin twice.

- x_1: H(head)/T(tail). x_2: H/T. x_3: F(fair coin)/B(biased coin)
- Joint distribution $p(x_1, x_2, x_3)$
- Conditional distribution $p(x_1, x_2 \mid x_3 = B)$

Table 1: Joint distribution

x_1	x_2	x_3	Probability
H	H	F	0.125
H	T	F	0.125
T	H	F	0.125
T	T	F	0.125
H	H	B	0.32
H	T	B	0.08
T	H	B	0.08
T	T	B	0.02

Table 2: Conditioning on $x_3 = B$

x_1	x_2	x_3	Probability
H	H	B	0.64
H	T	B	0.16
T	H	B	0.16
T	T	B	0.04

Conditional independence

Toy example

- Marginalization

Table 3: Conditioning on $x_3 = B$

x_1	x_2	x_3	Probability
H	H	B	0.64
H	T	B	0.16
T	H	B	0.16
T	T	B	0.04

Table 4: Marginalization

x_1	x_3	Probability
H	B	0.8
T	B	0.2

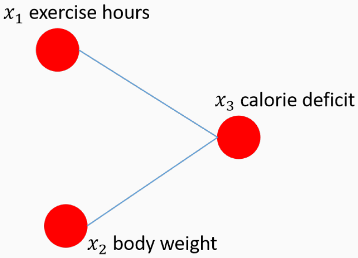
x_2	x_3	Probability
H	B	0.8
T	B	0.2

- x_1 and x_2 are conditionally independent given x_3 if
  $p(x_1, x_2 \mid x_3) = p(x_1 \mid x_3)p(x_2 \mid x_3)$

Conditional independence

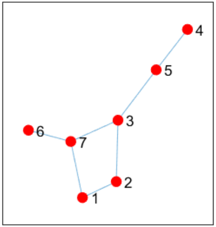
Toy example

- x_1: the number of hours a person spends exercising in a week
- x_2: the person's body weight
- x_3: the person's calorie deficit in a week
  — calorie deficit = calorie received (eating) − calorie spent (breathing, walking, exercising, etc)
- they are correlated
- given x_3, x_1 and x_2 are conditionally independent



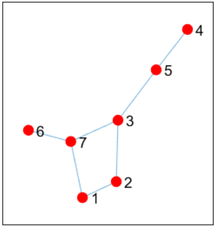
Undirected graph

- A **undirected graph** $G = (V, E)$ consists of a vertex set $V = \{1, 2, \ldots, p\}$, and an edge set $E \subset V \times V$
- "Undirected": the edge $(s, t) \in E =$ the edge $(t, s) \in E$.
- Assume no self-loop



Undirected graph

- A **undirected graph** $G = (V, E)$ consists of a vertex set $V = \{1, 2, \ldots, p\}$, and an edge set $E \subset V \times V$
- "Undirected": the edge $(s, t) \in E$ = the edge $(t, s) \in E$.
- Assume no self-loop



Connected

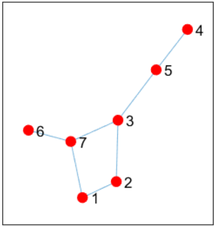
Vertex set $V = \{1, 2, 3, 4, 5, 6, 7\}$

Edge set $E =$

$\{(6, 7), (7, 3), (7, 1), (1, 2), (2, 3), (3, 5), (5, 4)\}$

Undirected graph

- A **undirected graph** $G = (V, E)$ consists of a vertex set $V = \{1, 2, \ldots, p\}$, and an edge set $E \subset V \times V$
- "Undirected": the edge $(s, t) \in E$ = the edge $(t, s) \in E$.
- Assume no self-loop

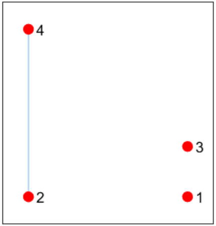


Connected

Vertex set $V = \{1, 2, 3, 4, 5, 6, 7\}$

Edge set $E =$

$\{(6,7), (7,3), (7,1), (1,2), (2,3), (3,5), (5,4)\}$



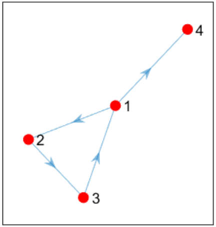
Disconnected

Vertex set $V = \{1, 2, 3, 4\}$

Edge set $E = \{(2, 4)\}$

Directed graphs

- In directed graphs, the edges have directionality. Directed graphs are more difficult to handle
- In our context, we only consider undirected graphs without self-loop



Vertex set $V = \{1, 2, 3, 4\}$

Edge set $E = \{(2,3), (3,1), (1,2), (1,4)\}$

The edge $(2,3)$ is from node 2 to node 3, different from the edge $(3,2)$

Markov property

- Consider a cut set S that separates the graph into disconnected components A and B
- Use $\perp\!\!\!\perp$ to denote the relation "is conditionally independent of"
- We say that the random variable x is Markov with respect to $G = (V, E)$ if

$$x_A \perp\!\!\!\perp x_B \mid x_S \quad \text{for all cut sets } S \subset V$$

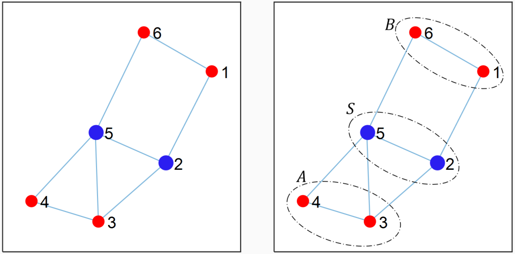
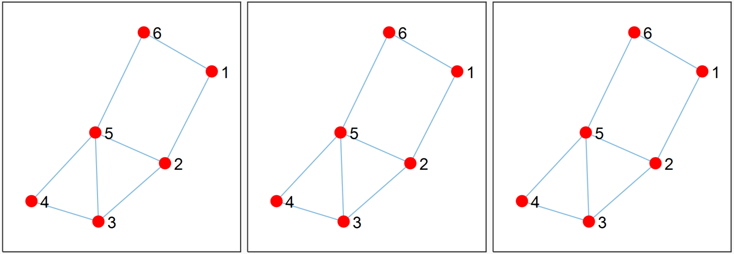


Figure 1: When the vertices in the cut set S are removed, the graph is broken into two sub-components A and B.

Illustration of Markov property



(a) **(b)** **(c)**

For example, we have the conditional independence

(a) $x_6 \perp\!\!\!\perp x_2, x_3, x_4 \mid x_1, x_5$

(b) $x_6, x_1, x_2 \perp\!\!\!\perp x_4 \mid x_3, x_5$

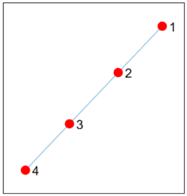
(c) $x_1 \perp\!\!\!\perp x_5 \mid x_2, x_3, x_5, x_6$

Illustration of Markov property

- Consider a chain-structured graph with edge set

$$E = \{(1,2), (2,3), \ldots, (p-1, p)\}$$

- Any single vertex $s \in \{2, 3, \ldots, p-1\}$ forms a cut set



- The cut set $\{s\}$ separates the graph into two sub-components

$$\text{past } P = \{1, \ldots, s-1\}$$
$$\text{furture } F = \{s+1, \ldots, p\}$$

- Markov property: for a Markov chain, the future x_F is conditionally independent of the past x_P given the present x_s.

Multivariate Gaussians

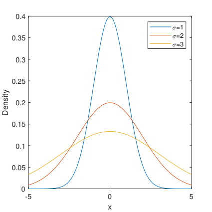
The probability density function of a multivariate Gaussian distribution with mean vector $\mu \in \mathbb{R}^p$ and covariance matrix $\Sigma \in \mathbb{S}^p_{++}$ is

$$f(x) = \frac{1}{(2\pi)^{p/2} \det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

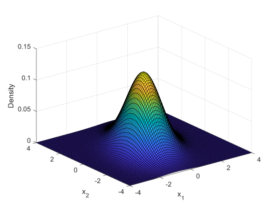
- $x \sim N(\mu, \Sigma)$ in $\mathbb{R}^p$
- $\det(\Sigma)$ denotes the determinant of the covariance matrix
  - ▷ $\det(\Sigma) = \prod_{j=1}^{p} \lambda_j(\Sigma)$, where $\lambda_j(\Sigma)$ is the j-the eigenvalue of $\Sigma \in \mathbb{S}^p$
- $\Theta = \Sigma^{-1}$ denotes the inverse covariance matrix or precision matrix
  - ▷ $\det(\Theta) = \dfrac{1}{\det(\Sigma)}$
- For simplicity, assume $\mu = 0$

Multivariate Gaussians

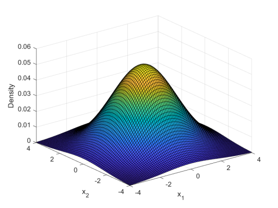
$$f(x) = \frac{1}{(2\pi)^{p/2}\det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}x^T\Sigma^{-1}x\right)$$



(d)



(e)



(f)

(d) $p = 1$. Let $\Sigma = \sigma^2$, the probability density function reduces to

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right), \qquad x \sim N(0, \sigma^2)$$

(e) $p = 2$, $\Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix}$ (f) $p = 2$, $\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$

Gaussian graphical models

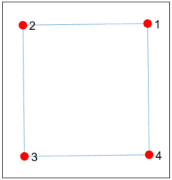
Gaussian graphical models

The essential idea of Gaussian graphical models are

- to represent a collection of Gaussian random variables $x = (x_1, \ldots, x_p)^T$ by the vertex set $V = \{1, \ldots, p\}$
- to represent the "relationships" (conditional independence) of variables by the edge set E
  - ▷ For any pair of vertices s and t,

$$(s,t) \notin E \iff x_s \perp\!\!\!\perp x_t \mid x_{V \setminus \{s,t\}}$$

where $V \setminus \{s,t\} = \{k \mid 1 \le k \le p,\ k \ne s,\ k \ne t\}$



If random variable $x = (x_1, x_2, x_3, x_4)^T$ is Markov w.r.t. the left graph, then

$x_1 \perp\!\!\!\perp x_3 \mid x_2, x_4$

$x_2 \perp\!\!\!\perp x_4 \mid x_1, x_3$

Gaussian graphical models

Fact [4] Consider a Gaussian vector $x \sim N(0, \Sigma)$. For any s and t,

$$x_s \perp\!\!\!\perp x_t \mid x_{V \setminus \{s,t\}}$$

if and only if

$$\Theta_{st} = 0, \quad \Theta = \Sigma^{-1}.$$

That is, we can characterize the conditional independence of Gaussian random variables by the graph with edge set $E = \{(s,t) \mid \Theta_{st} \neq 0, s < t\}$

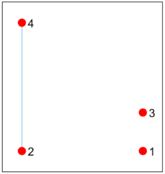
	conditional independence	$x_s \perp\!\!\!\perp x_t \mid x_{V \setminus \{s,t\}}$
$\iff$	zero in precision	$\Theta_{st} = 0$
$\iff$	no edge in graph	$(s,t) \notin E$

Example

Example. Represent by a graph the conditional independence of a Gaussian vector $(x_1, x_2, x_3, x_4) \sim N(0, \Sigma)$, $\Sigma = \begin{bmatrix} 1 & & & \\ & 4/3 & & -2/3 \\ & & 1 & \\ & -2/3 & & 4/3 \end{bmatrix}$.

Solution.

1. Compute the precision matrix $\Theta = \Sigma^{-1} = \begin{bmatrix} 1 & & & \\ & 1 & & 0.5 \\ & & 1 & \\ & 0.5 & & 1 \end{bmatrix}$
2. Edge set $E = \{(2,4)\}$ since $\Theta_{24} \neq 0$. Vertex set $V = \{1,2,3,4\}$
3. Plot



Example

Example. Represent by a graph the conditional independence of a Gaussian vector $x \sim N(0, I)$, $x \in \mathbb{R}^p$.

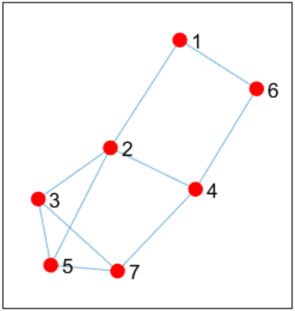
Solution.

1. Compute the precision matrix $\Theta = \Sigma^{-1} = I$
2. Edge set $E = \{(s,t) \mid \Theta_{st} \neq 0, s < t\} = \emptyset$ since all off-diagonal entries are zero. Vertex set $V = \{1, 2, \ldots, p\}$
3. The graph has p nodes and no edge.

- The graph interprets that $x_s \perp\!\!\!\perp x_t \mid \{1, \ldots, p\} \backslash \{s, t\}$
- In fact, $x \sim N(0, I) \iff x_i \sim N(0, 1) \, \forall \, i = 1, \ldots, p$. The random variable x_s and x_t are independent for $s \neq t$.

Gaussian graphical models

Correspondence between the zero pattern of Θ and the edge structure E of the graph



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Zero pattern of the precision matrix Θ. Here 1 correspond to $\Theta_{st} \neq 0$, $s \neq t$. Diagonals are 0.

Maximum likelihood estimation

- Given n samples $x^{(1)}, \ldots, x^{(n)} \sim N(0, \Sigma)$ independently
- Suppose the true covariance matrix Σ is unknown
- Aim to learn the covariance matrix Σ or the precision matrix $\Theta = \Sigma^{-1}$
- This problem is called covariance selection, precision estimation, or inverse covariance estimation

$$x \sim N(0, \Sigma), \quad f(x) = \frac{1}{(2\pi)^{p/2} \det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x\right)$$

Maximum likelihood estimation (MLE)

$$\max \quad \frac{1}{n} \sum_{i=1}^{n} \log f(x^{(i)})$$

It is equivalent to maximize the likelihood $\prod_{i=1}^{n} f(x^{(i)})$

Likelihoods

$$x \sim N(0, \Sigma), \quad f(x) = \frac{1}{(2\pi)^{p/2} \det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x\right)$$

Likelihoods

$$x \sim N(0, \Sigma), \quad f(x) = \frac{1}{(2\pi)^{p/2} \det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x\right)$$

$$\begin{aligned}\log f(x) &= \underbrace{-\log\left((2\pi)^{p/2}\right)}_{\text{constant}} - \frac{1}{2}\log\det(\Sigma) - \frac{1}{2}\langle \Sigma^{-1}, xx^T \rangle \\ &= \frac{1}{2}\log\det(\Theta) - \frac{1}{2}\langle \Theta, xx^T \rangle + \text{constant}\end{aligned}$$

The log-likelihood (up to additive constant) is

$$\frac{1}{n}\sum_{i=1}^{n} \log f(x^{(i)}) = \frac{1}{2}\log\det(\Theta) - \frac{1}{2n}\sum_{i=1}^{n}\langle \Theta, x^{(i)}(x^{(i)})^T \rangle$$

$$= \frac{1}{2}\log\det(\Theta) - \frac{1}{2}\langle S, \Theta \rangle$$

- $S = \frac{1}{n}\sum_{i=1}^{n} x^{(i)}(x^{(i)})^T$ is the sample covariance matrix
- $\Theta = \Sigma^{-1} \Rightarrow \det(\Theta) = \frac{1}{\det(\Sigma)} \Rightarrow \log\det(\Theta) = -\log\det(\Sigma)$

Maximum likelihood estimation

Maximum likelihood estimation (MLE)

$$
\begin{aligned}
&\max_{\Theta \in \mathbb{S}^p_{++}} \quad \log\det(\Theta) - \langle S, \Theta \rangle \\
\iff &\min_{\Theta \in \mathbb{S}^p_{++}} \quad -\log\det(\Theta) + \langle S, \Theta \rangle
\end{aligned}
\tag{MLE}
$$

- The log-determinant function

$$
h(\Theta) = -\log\det(\Theta) = \begin{cases} -\sum_{j=1}^{p} \log(\lambda_j(\Theta)), & \text{if } \Theta \in \mathbb{S}^p_{++} \\ +\infty, & \text{otherwise.} \end{cases}
$$

  Here $\lambda_j(\Theta)$ is the j-th eigenvalue of Θ
- $h(\Theta) = -\log\det(\Theta)$ is convex on $\mathbb{S}^p_{++}$
- $\nabla h(\Theta) = -\Theta^{-1}$ for any $\Theta \in \mathbb{S}^p_{++}$
- If the solution $\widehat{\Theta}$ to (MLE) exists, then $\widehat{\Theta} = S^{-1}$

Understand log-determinant function

It is nontrivial to prove

- $h(\Theta) = -\log\det(\Theta)$ is convex on $\mathbb{S}^p_{++}$
- $\nabla h(\Theta) = -\Theta^{-1}$ for any $\Theta \in \mathbb{S}^p_{++}$

Let's understand them on a diagonal $\Theta = \mathrm{Diag}(\Theta_{jj})$, $\Theta_{jj} > 0$

- $h(\Theta) = -\log\det(\Theta) = -\sum_{j=1}^p \log(\Theta_{jj})$ is convex since $-\log(\cdot)$ is convex
- $\nabla h(\Theta) = \begin{bmatrix} \frac{\partial}{\partial \Theta_{11}} h & & \\ & \ddots & \\ & & \frac{\partial}{\partial \Theta_{pp}} h \end{bmatrix} = \begin{bmatrix} -\frac{1}{\Theta_{11}} & & \\ & \ddots & \\ & & -\frac{1}{\Theta_{pp}} \end{bmatrix} = -\Theta^{-1}$

Challenge in high-dimensional regime

- MLE coverges to the truth as sample size $n \to +\infty$
- Practically, we are often in the regime where $n < p$
- In this regime, S is rank-deficient, and the solution to (MLE) does not even exist
- Consider regularization
  - ▷ Guarantee that a solution exists
  - ▷ The estimated precision matrix tends to be sparse and easy to interpret

Consider the following MLE with regularization (often reffered to as graphical Lasso)

$$\min_{\Theta \in \mathbb{S}^p_{++}} \quad -\log\det(\Theta) + \langle S, \Theta \rangle + \lambda \|\Theta\|_{1,\text{off}}$$

$\|\Theta\|_{1,\text{off}} = \sum_{s \neq t} |\Theta_{ij}|$ the ℓ_1-norm of the off-diagonal entries

Algorithm: ADMM

Apply ADMM for solving the graphical Lasso problem

$$\min_{\Theta \in \mathbb{S}^p_{++}} \quad -\log\det(\Theta) + \langle S, \Theta \rangle + \lambda \|\Theta\|_{1,\text{off}}$$

- Transform it into the form that ADMM can handle

$$\begin{aligned} \min_{\Theta, Y} \quad & -\log\det(\Theta) + \langle S, \Theta \rangle + \delta_{\mathbb{S}^p_{++}}(\Theta) + \lambda \|Y\|_{1,\text{off}} \\ \text{s.t.} \quad & \Theta - Y = 0 \end{aligned}$$

- The augmented Lagrangian

$$\begin{aligned} L_\sigma(\Theta, Y, Z) = & -\log\det(\Theta) + \langle S, \Theta \rangle + \delta_{\mathbb{S}^p_{++}}(\Theta) + \lambda \|Y\|_{1,\text{off}} \\ & + \langle Z, \Theta - Y \rangle + \frac{\sigma}{2} \|\Theta - Y\|_F^2 \end{aligned}$$

- ADMM iterations $\begin{cases} \Theta \leftarrow \arg\min_{\Theta} L_\sigma(\Theta, Y, Z) \\ Y \leftarrow \arg\min_{Y} L_\sigma(\Theta, Y, Z) \\ Z \leftarrow Z + \tau\sigma(\Theta - Y) \end{cases}$

Subproblem-Θ

$$L_\sigma(\Theta, Y, Z) = -\log\det(\Theta) + \langle S, \Theta\rangle + \delta_{\mathbb{S}^p_{++}}(\Theta) + \lambda\|Y\|_{1,\text{off}}$$
$$+ \langle Z, \Theta - Y\rangle + \frac{\sigma}{2}\|\Theta - Y\|_F^2$$

Consider minimizing w.r.t. Θ

$$\begin{aligned}
&\arg\min_{\Theta\in\mathbb{S}^p_{++}} -\log\det(\Theta) + \langle S+Z, \Theta\rangle + \frac{\sigma}{2}\|\Theta - Y\|_F^2 \\
&= \arg\min_{\Theta\in\mathbb{S}^p_{++}} -\log\det(\Theta) + \frac{\sigma}{2}\|\Theta - Y + \sigma^{-1}(S+Z)\|_F^2 \\
&= P_{\frac{1}{\sigma}h}\left(Y - \sigma^{-1}(S+Z)\right)
\end{aligned}$$

where $h(\Theta) = -\log\det(\Theta) = \begin{cases} -\sum_{j=1}^p \log(\lambda_j(\Theta)), & \text{if } \Theta \in \mathbb{S}^p_{++} \\ +\infty, & \text{otherwise.} \end{cases}$

Proximal mapping of log-determinant function

Theorem The proximal mapping

$$P_{\frac{1}{\sigma}h}(Y) = \operatorname*{arg\,min}_{\Theta \in \mathbb{S}^p_{++}} \left\{ -\log\det(\Theta) + \frac{\sigma}{2}\|\Theta - Y\|_F^2 \right\}$$

is obtained by

$$\begin{aligned} Y &= Q\mathrm{Diag}(\rho)Q^T \\ \gamma_j &= \tfrac{1}{2}\left(\rho_j + \sqrt{\rho_j^2 + 4/\sigma}\right) \\ P_{\frac{1}{\sigma}h}(Y) &= Q\mathrm{Diag}(\gamma)Q^T \end{aligned}$$

Proof The proof is based on

1. The log-determinant function and Frobenius norms are orthogonally invariant: $\det(U\Sigma V^T) = \det(\Sigma)$, $\|U\Sigma V^T\|_F = \|\Sigma\|_F$ for any Σ and orthogonal U, V
2. Therefore, we can work with diagonal matrices Θ and Y

Proximal mapping of log-determinant function

Proof For $\Theta = \mathrm{Diag}(\gamma)$, $Y = \mathrm{Diag}(\sigma)$ (we need $\gamma > 0$, but no constraints for ρ), solve the problem

$$\min_{\Theta \in \mathbb{S}^p_{++}} \left\{ -\log\det(\Theta) + \frac{\sigma}{2}\|\Theta - Y\|_F^2 \right\}$$

Compute the gradient

$$0 = -\Theta^{-1} + \sigma(\Theta - Y) = -\begin{bmatrix} \frac{1}{\gamma_1} & & \\ & \ddots & \\ & & \frac{1}{\gamma_p} \end{bmatrix} + \sigma \begin{bmatrix} \gamma_1 - \rho_1 & & \\ & \ddots & \\ & & \gamma_p - \rho_p \end{bmatrix}$$

$$\Rightarrow -\frac{1}{\gamma_j} + \sigma(\gamma_j - \rho_j) = 0 \Rightarrow \gamma_j = \frac{1}{2}\left(\rho_j + \sqrt{\rho_j^2 + 4/\sigma}\right) > 0$$

Subproblem-Y

$$
\begin{aligned}
L_\sigma(\Theta, Y, Z) = & -\log\det(\Theta) + \langle S, \Theta\rangle + \delta_{\mathbb{S}^p_{++}}(\Theta) + \lambda\|Y\|_{1,\text{off}} \\
& + \langle Z, \Theta - Y\rangle + \frac{\sigma}{2}\|\Theta - Y\|_F^2
\end{aligned}
$$

Consider minimizing w.r.t. Y

$$
\begin{aligned}
& \arg\min_Y \lambda\|Y\|_{1,\text{off}} - \langle Z, Y\rangle + \frac{\sigma}{2}\|Y - \Theta\|_F^2 \\
= & \arg\min_Y \lambda\|Y\|_{1,\text{off}} + \frac{\sigma}{2}\|Y - \Theta - \sigma^{-1}Z\|_F^2 \\
= & P_{\frac{\lambda}{\sigma}\|\cdot\|_{1,\text{off}}}\left(Y - \sigma^{-1}(S + Z)\right) \\
= & S^{\text{off}}_{\frac{\lambda}{\sigma}}\left(Y - \sigma^{-1}(S + Z)\right)
\end{aligned}
$$

$S^{\text{off}}_{\frac{\lambda}{\sigma}}$ only thresholds the off-diagonal entries:

$$
\begin{cases}
Y_{ii} - \sigma^{-1}(S_{ii} + Z_{ii}) & \text{if } i = j \\
S_{\frac{\lambda}{\sigma}}\left(Y_{ij} - \sigma^{-1}(S_{ij} + Z_{ij})\right) & \text{if } i \neq j
\end{cases}
$$

ADMM framework

Algorithm (ADMM for graphical Lasso)

Choose $\sigma > 0$, $0 < \tau < \frac{1+\sqrt{5}}{2}$, $Y^{(0)} = (S + \lambda I)^{-1}$, $Z^{(0)} = I$. $k \leftarrow 0$

repeat until convergence

$T^{(k)} \leftarrow Y^{(k)} - \sigma^{-1}\left(S + Z^{(k)}\right)$

$T^{(k)} = Q^{(k)}\mathrm{Diag}(d^{(k)})(Q^{(k)})^T$ (eigen. decomp. of $T^{(k)}$)

for $j = 1, \ldots, p$ $\quad \gamma_j^{(k)} \leftarrow \dfrac{1}{2}\left(d_j^{(k)} + \sqrt{(d_j^{(k)})^2 + 4/\sigma}\right)$ **end(for)**

$\Theta^{(k+1)} \leftarrow Q^{(k)}\mathrm{Diag}(\gamma^{(k)})(Q^{(k)})^T$

$Y^{(k+1)} \leftarrow S^{\mathrm{off}}_{\frac{\lambda}{\sigma}}\left(Y^{(k)} - \sigma^{-1}(S + Z^{(k)})\right)$

$Z^{(k+1)} \leftarrow Z^{(k)} + \tau\sigma(\Theta^{(k+1)} - Y^{(k+1)})$

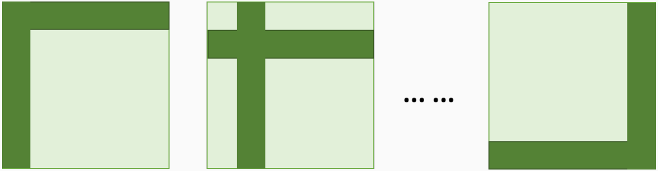
end(repeat)

return $\Theta^{(k)}, Y^{(k)}, Z^{(k)}$

Existing methods

In addition to ADMM, existing first order methods in the literature includes:

- [1] BCD, in each step, optimize a single column/row



- [5] APG

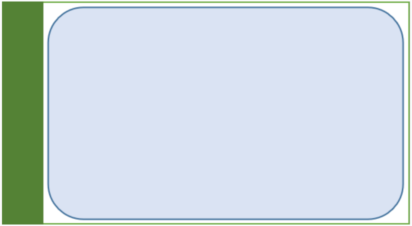
Neighbourhood selection

Neighbourhood selection

Recall we aim to learn the precision matrix $\Theta = \Sigma^{-1}$ given n samples $x^{(1)}, \ldots, x^{(n)} \sim N(0, \Sigma)$ independently, where the true covariance matrix Σ is unknown

- Data matrix $X = [x^{(1)}, \ldots, x^{(n)}]^T \in \mathbb{R}^{n \times p}$

Neighbourhood selection



- $X_{\cdot j}$: j-th column, $X_{\cdot -j}$: delete j-th column
- Sparse linear regression for each vertex

$$X_{\cdot j} \approx X_{\cdot -j} \beta^j$$

- Via solving Lasso

$$\beta^j = \underset{\beta \in \mathbb{R}^{p-1}}{\arg\min} \frac{1}{2} \|X_{\cdot -j}\beta - X_{\cdot j}\|^2 + \lambda \|\beta\|_1$$

- Determine the neighbourhood $\mathcal{N}(j)$ of vertex j

- Form a graph via the AND or OR rule
  - ▷ AND rule: edge (s, t) exists if $s \in \mathcal{N}(t)$ and $t \in \mathcal{N}(s)$
  - ▷ OR rule: edge (s, t) exists if $s \in \mathcal{N}(t)$ or $t \in \mathcal{N}(s)$

Example. Given the neighbourhood of four variables:

$$\mathcal{N}(1) = \{3, 4\}, \quad \mathcal{N}(2) = \{3\}, \quad \mathcal{N}(3) = \{1\}, \quad \mathcal{N}(4) = \{1, 2\}$$

Form a graph via the AND rule (and OR rule).

Solution.

AND rule gives $E = \{(1, 3), (1, 4)\}$

OR rule gives $E = \{(1, 3), (1, 4), (2, 4), (2, 3)\}$

Neighbourhood selection framework

Algorithm (Neighbourhood selection)

for $j = 1, \dots, p$

$$\beta^j \leftarrow \arg\min_{\beta \in \mathbb{R}^{p-1}} \frac{1}{2} \|X_{\cdot -j}\beta - X_{\cdot j}\|^2 + \lambda \|\beta\|_1$$

$$\mathcal{N}(j) = \{t \mid \beta^j_t \neq 0\}$$

end(for)

Form a graph $G = (V, E)$ from $\mathcal{N}(j), j = 1, \dots, p$ via AND/OR rule

return graph G

Neighbourhood selection

Example

$n = 3$, $p = 4$. Observe 4 Gaussian samples

$$x^{(1)} = (0.54, 0.95, -0.25, 2.39)^T \quad x^{(2)} = (1.85, 0.12, 0.40, -1.60)^T$$
$$x^{(3)} = (-2.28, -1.24, 3.61, 2.21)^T$$

Data matrix $X \in \mathbb{R}^{3\times 4}$

$$\begin{bmatrix} 0.54 & 0.95 & -0.25 & 2.39 \\ 1.85 & 0.12 & 0.40 & -1.60 \\ -2.28 & -1.24 & 3.61 & 2.21 \end{bmatrix}$$

```
X = [0.54  0.95  -0.25  2.39;
     1.85  0.12   0.40 -1.60;
    -2.28 -1.24   3.61  2.21]; % data matrix
% lasso for vertex 1
beta = lasso(X(:,2:4),X(:,1),'Lambda',0.5)
beta =
         0
   -0.5271
   -0.2449
```

For vertex 1, the neighbourhood is $\mathcal{N}(1) = \{3, 4\}$

```
X = [0.54  0.95  -0.25  2.39;
      1.85  0.12   0.40 -1.60;
     -2.28 -1.24   3.61  2.21]; % data matrix
% lasso for vertex 1
beta = lasso(X(:,2:4),X(:,1),'Lambda',0.5)
beta =
         0
   -0.5271
   -0.2449
```

For vertex 1, the neighbourhood is $\mathcal{N}(1) = \{3, 4\}$

```
X = [0.54  0.95  -0.25  2.39;
      1.85  0.12   0.40 -1.60;
     -2.28 -1.24   3.61  2.21]; % data matrix
% lasso for vertex 2
beta = lasso(X(:,[1,3,4]),X(:,2),'Lambda',0.5)
beta =
         0
   -0.2250
         0
```

For vertex 2, the neighbourhood is $\mathcal{N}(2) = \{3\}$

```
X = [0.54  0.95  -0.25  2.39;
      1.85  0.12   0.40 -1.60;
     -2.28 -1.24   3.61  2.21]; % data matrix
% lasso for vertex 3
beta = lasso(X(:,[1,2,4]),X(:,3),'Lambda',0.5)
beta =
   -0.1774
   -1.0086
         0
```

For vertex 3, the neighbourhood is $\mathcal{N}(3) = \{1, 2\}$

```
X = [0.54  0.95  -0.25  2.39;
      1.85  0.12   0.40 -1.60;
     -2.28 -1.24   3.61  2.21]; % data matrix
% lasso for vertex 4
beta = lasso(X(:,1:3),X(:,4),'Lambda',0.5)
beta =
   -0.4752
         0
         0
```

For vertex 4, the neighbourhood is $\mathcal{N}(4) = \{1\}$

Neighbourhood selection

Example Neighbourhood

$$\mathcal{N}(1) = \{3, 4\}, \quad \mathcal{N}(2) = \{3\}, \quad \mathcal{N}(3) = \{1, 2\}, \quad \mathcal{N}(4) = \{1\}$$

AND rule gives $E = \{(1, 3), (2, 3), (1, 4)\}$. OR rule gives the same graph.

Neighbourhood selection

Example Neighbourhood

$$\mathcal{N}(1) = \{3, 4\}, \quad \mathcal{N}(2) = \{3\}, \quad \mathcal{N}(3) = \{1, 2\}, \quad \mathcal{N}(4) = \{1\}$$

AND rule gives $E = \{(1,3), (2,3), (1,4)\}$. OR rule gives the same graph.

Remarks

- Neighbourhood selection is fast since
- many efficient solvers for lasso are available
- the p Lasso problems for each vertex can be solved in parallel

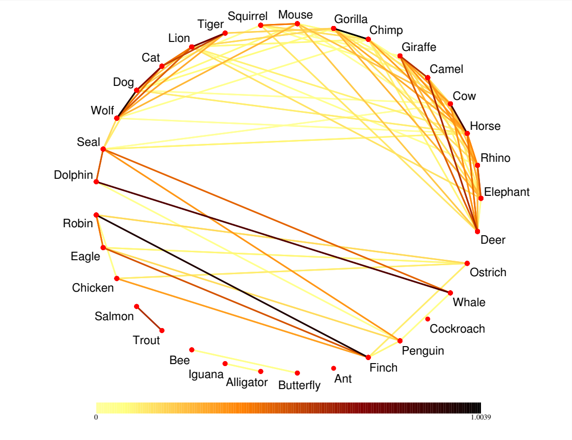
Applications

Animals data [2]

- $p = 33$ animals
- $n = 102$ questions: "has lungs?", "is warm-blooded?" ...
- Each entry is a true-false answer to the question
- Analyze the relation among animals

	Bee	Cat	Lion	... $p = 33$
has lungs?	0	1		
is warm-blooded?	0	1		
live in groups?	1	0		
.				
.				
.				
$n = 102$				

Animals similarity



Dolphin -- Whale

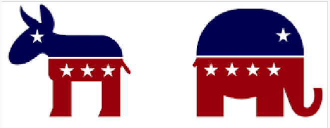
Robin -- Finch

Similar animals are connected with edges of large weights

Politician network [3]

- $p = 50$ politicians (Democratic / Republican senators)

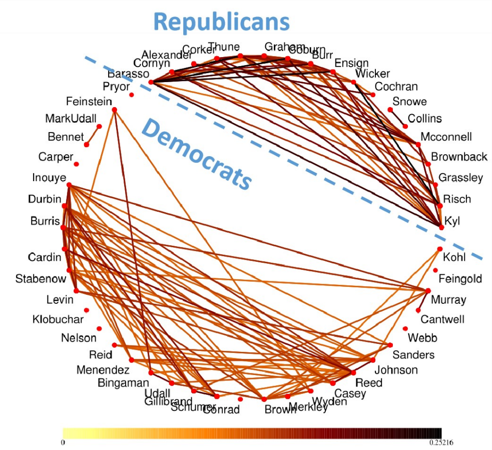
Democrats vs Republicans



- $n = 293$ voting records for the 111th United States Congress (2009-2011)
- Each entry is a yes-no vote
- Analyze the networks from politician's behaviors

$$
\begin{array}{cc}
 & \text{Alexander} \quad \text{Bennet} \ \ldots \ p = 50 \\
n = 293 & \begin{array}{|cc|} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ & \\ & \end{array}
\end{array}
$$

Politician network



A clear divide between Democrats (bottom left nodes) and Republicans (top right nodes)

The senators are clustered into two parties

J. Friedman, T. Hastie, and R. Tibshirani.
Sparse inverse covariance estimation with the graphical lasso.
Biostatistics, 9(3):432–441, 2008.

C. Kemp and J. B. Tenenbaum.
The discovery of structural form.
Proceedings of the National Academy of Sciences, 105(31):10687–10692, 2008.

B. M. Lake, N. D. Lawrence, and J. B. Tenenbaum.
The emergence of organizing structure in conceptual representation.
Cognitive science, 42:809–832, 2018.

S. L. Lauritzen.
Graphical models, volume 17.
Clarendon Press, 1996.

Z. Lu.
Smooth optimization approach for sparse covariance selection.
SIAM Journal on Optimization, 19(4):1807–1827, 2009.