Structure Learning in Infrastructure Networks

Rajasekhar Anguluri

Department of Computer Science and Electrical Engineering

University of Maryland, Baltimore County (UMBC)

rajangul@umbc.edu

Summer Workshop, July 15 – 18, 2024

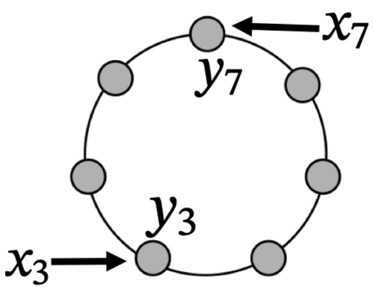
IIT, Bombay

slides template inspiration: Joel Tropp, Caltech

https://rajanguluri.github.io

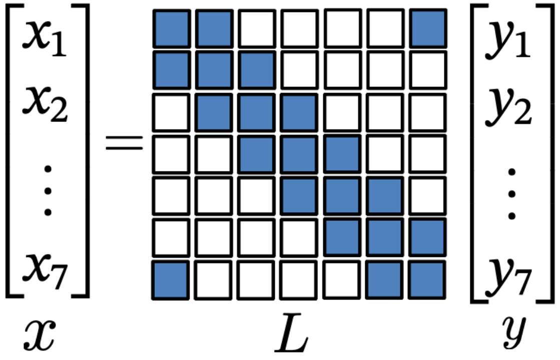
Structure Learning: Recap

Network Structure = Laplacian's Sparsity Pattern



infrastructure network

sparsity (zero & non-zero) of L captures network connections



nodal injections

network Laplacian

node potentials

- ***measurables:*** p-dim vectors x and y
- ***full coverage:*** access x or/and y
- ***partial coverage:*** sub-vectors of x or/and y

- ***linear model:***

$$X = LY + E$$ matrix-valued

$$\text{Vec}(X) = H(Y)\,\text{Ve}(L) + \text{Vec}(E)$$ vector-valued

- ***covariance models:***

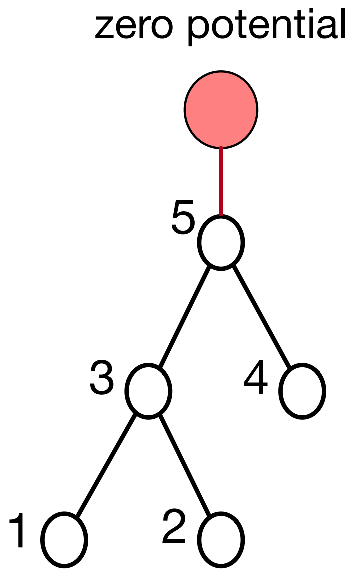
Today's lecture

Learning Sparse Infrastructure Networks

- **Goals:** learn structure in sparsely connected infrastructure networks using covariance models
  - **Goal 1:** structure learning from true full covariance matrix
  - **Goal 2:** structure learning from true partial covariance matrix
  - **Goal 3:** structure learning from sample covariance matrix

Structure Learning: True (full) Covariance

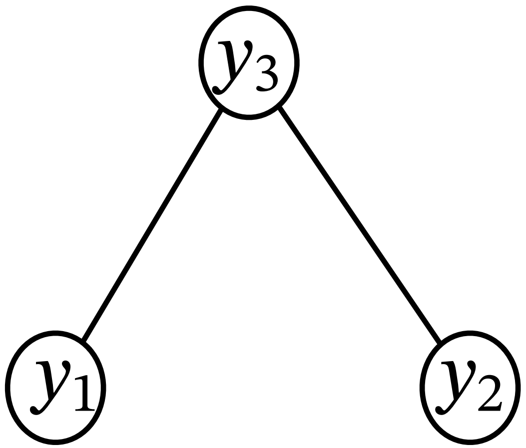
- *assumption 1*: $p \times p$-dimensional reduced-order Laplacian
- *equilibrium equation:* $y = L^{-1}x$
- *injections:* $x \sim \mathcal{N}(0, \Sigma_x)$
- *potentials:* $y \sim \mathcal{N}(0, \Sigma)$, where $\Sigma = L^{-1}\Sigma_x L^{-1}$
- *assumption 2*: Σ_x is diagonal but is UNKNOWN
- *learning problem:* estimate the sparsity (zero/non-zero pattern) of L from the covariance matrix Σ



Graphical Model Interpretation of $\Omega \triangleq \Sigma^{-1}$

- *Gaussian vector*: $y \sim \mathcal{N}(0, \Sigma)$ // parameterized by covariance
- *graphical model:* $y \sim \mathcal{N}(0, \Omega^{-1})$ // parameterized by inverse covariance
- example: suppose that $\epsilon_i \sim \mathcal{N}(0,1)$ and let

$$
\begin{aligned}
y_1 &= y_3 + \varepsilon_1 \\
y_2 &= y_3 + \varepsilon_2 \\
y_3 &= \varepsilon_3
\end{aligned}
$$



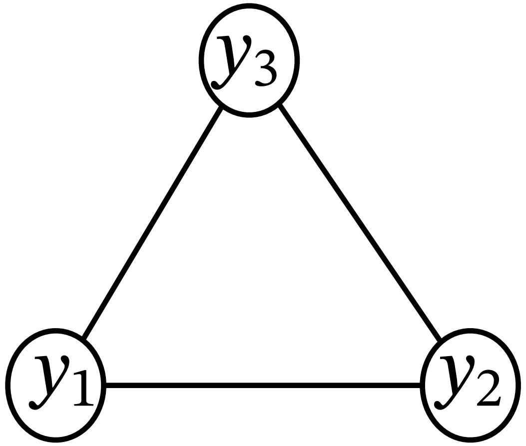
$$
\Omega = \Sigma^{-1} = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 3 \end{bmatrix}
$$

partial covariance or conditional independence or graphical model

Graphical Model Interpretation of $\Omega \triangleq \Sigma^{-1}$

- *Gaussian vector*: $y \sim \mathcal{N}(0, \Sigma)$ // parameterized by covariance
- *graphical model:* $y \sim \mathcal{N}(0, \Omega^{-1})$ // parameterized by inverse covariance
- example: suppose that $\epsilon_i \sim \mathcal{N}(0,1)$ and let

$$
\begin{aligned}
y_1 &= y_3 + \varepsilon_1 \\
y_2 &= y_3 + \varepsilon_2 \\
y_3 &= \varepsilon_3
\end{aligned}
$$

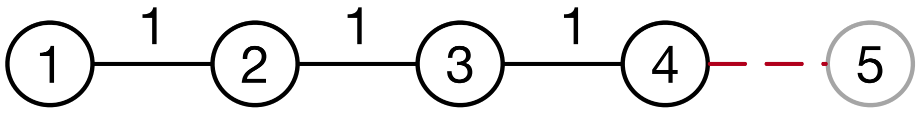


covariance or independence graph

$$
\Sigma = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}
$$

Infrastructure Network → Graphical Model

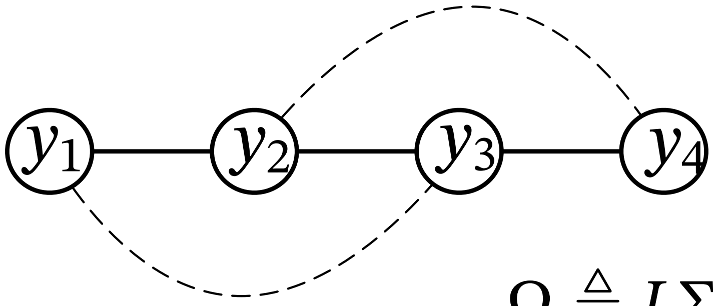
infrastructure network



$$L_{\text{org}} = \left[\begin{array}{cccc|c} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ \hline 0 & 0 & 0 & -1 & 1 \end{array}\right]$$

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}$$

(two-hop) graphical model



$$\Omega \triangleq L\Sigma_x^{-1}L$$

$$\Omega = L^2 = \begin{bmatrix} 2 & -3 & 1 & 0 \\ -3 & 6 & -4 & 1 \\ 1 & -4 & 6 & -4 \\ 0 & 1 & -4 & 5 \end{bmatrix}$$

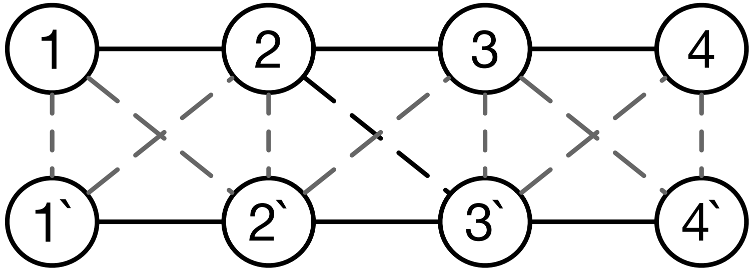
❧ +/-ve sign pattern is crucial for reconstruction

Exercises: Prove the following

- **Theorem 1** (real-case [1]) Graphical model for node potentials in an infrastructure network includes edges between potentials at neighbors and two-hop neighbors.

- **Theorem 2** (complex-lifted-case [1]) Graphical model for real and imaginary potentials in infrastructure network includes edges between real and imaginary potentials (i) at the self nodes; (ii) neighbors; and (iii) two-hop neighbors

$$\begin{bmatrix} y_R(k) \\ y_I(k) \end{bmatrix} = \underbrace{\begin{bmatrix} L_{RR} & L_{RI} \\ L_{IR} & L_{II} \end{bmatrix}}_{L_{\text{real-imag}}}{}^{-1} \begin{bmatrix} x_R(k) \\ x_I(k) \end{bmatrix}$$

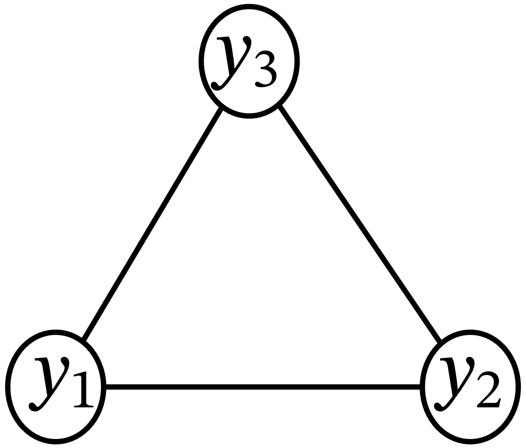


- edge-connectivity by $L_{\text{real}-\text{imag}}$: for every node associate an imaginary node (e.g., 1' for 1) – this is not a graphical model
- draw the connections based on the sparsity pattern of $L_{\text{real}-\text{imag}}$ (try this for line graph)

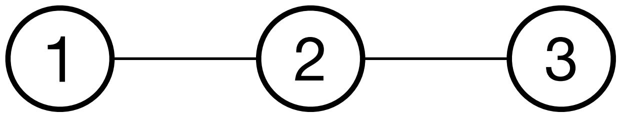
[1] D. Deka et. al. (2023) Graphical models in meshed distribution grids: topology estimation, change detection, and limitations, IEEE Trans on Smart Grid, 11 (5)

Graphical Model → Infrastructure Network

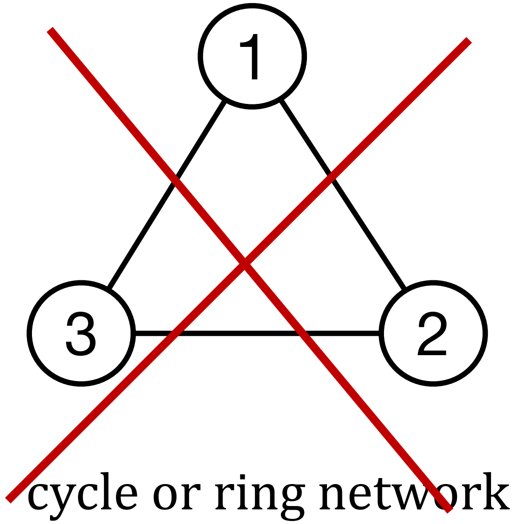
- *inverse problem:* from graphical model $\Omega = L\Omega_x L$ learn sparsity of L



graphical model Ω



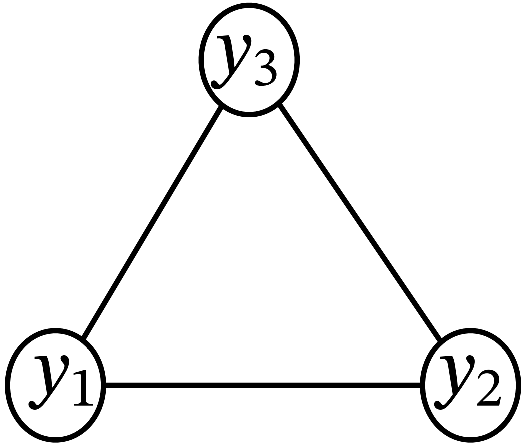
line network L



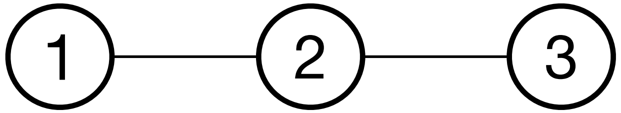
cycle or ring network L

Graphical Model → Infrastructure Network

- *inverse problem:* from graphical model $\Omega = L\Omega_x L$ learn sparsity of L



graphical model Ω



line network L_1

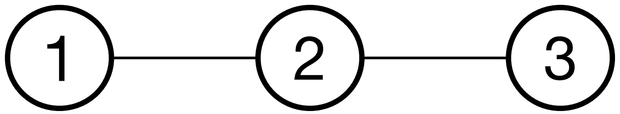
$$\Omega_1 = \begin{bmatrix} + & - & + \\ - & + & - \\ + & - & + \end{bmatrix}$$

line network L_2

$$\Omega_2 = \begin{bmatrix} + & + & - \\ + & + & - \\ - & - & + \end{bmatrix}$$

Graphical Model → Infrastructure Network

- *numerical exercise:* for L_1 and L_2 below and some positive diagonal Ω_x, verify the sparsity pattern of Ω_1 and Ω_2 shown on the right



line network L_1

$$L_1 = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \qquad \Omega_1 = \begin{bmatrix} + & - & + \\ - & + & - \\ + & - & + \end{bmatrix}$$

line network L_2

$$L_2 = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 3 \end{bmatrix} \qquad \Omega_2 = \begin{bmatrix} + & + & - \\ + & + & - \\ - & - & + \end{bmatrix}$$

Graphical Model → Infrastructure Network

- *sign-based algorithms:* consider the infrastructure network $\mathcal{G}$ with *minimum cycle length* greater than three. Then show that algorithms below work (exercise)

network reconstruction : real-case [1]

input: inverse covariance Ω and threshold $\tau > 0$

output: graph $\mathcal{G}$

1. for all nodes i and j do
2. if $\Omega(i,j) < -\tau$ then
3. insert edge (i,j) in $\mathcal{G}$
4. end if
5. end for

network reconstruction : complex-lift-case [1]

input: block matrix $\Omega_{(\mathrm{R,I})} = \begin{bmatrix} J_{RR} & J_{RI} \\ J_{IR} & J_{II} \end{bmatrix}$ and $\tau > 0$,

output: graph $\mathcal{G}$

1. for all nodes i and j do
2. if $\mathrm{J}_{RR}(i,j) + \mathrm{J}_{II}(i,j) < -\tau$ then
3. insert edge (i,j) in $\mathcal{G}$
4. end if
5. end for

[1] D. Deka et. al. (2023) Graphical models in meshed distribution grids: topology estimation, change detection, and limitations, IEEE Trans on Smart Grid, 11 (5)

Open problems

- **Problem 1** Use complex-valued Gaussian graphical model theory [1] to obtain above theorems using complex-valued covariance matrices (no lifting)

- **Problem 2** (favorite) Provide a linear-time algorithm to determine ”tree” network using the inverse covariance matrix $\Omega = L\Omega_x L$

  - The algorithms we described have second order complexity: $O(p\text{^}2)$

  - Ross and Harary (1959) considered problem 2 using adjacency matrices; they coined the terms square and square root of trees [2,3]

[1] J. K. Tugnait (2022) On sparse high-dimensional graphical model learning for dependent time series, Signal Processing, 197: 108539

[2] I. C. Ross and F. Harary (1959) The square of a tree, The Bell System Technical Journal, 39 (3): 641-647

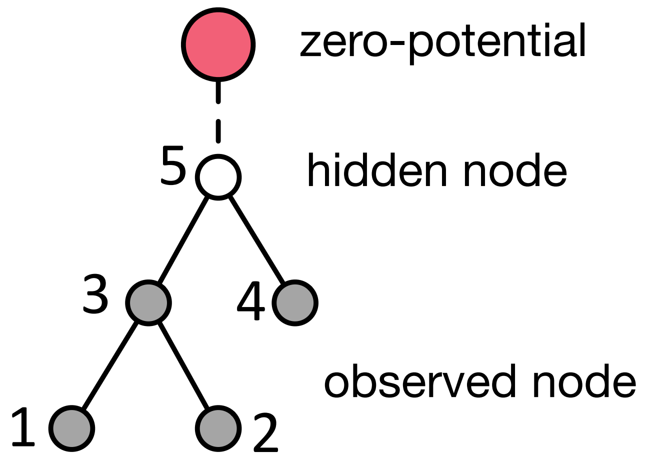
[3] Y. L. Lin and S. S. Skiena (1995) Algorithms for square roots of graphs, SIAM Journal on Discrete Mathematics 8(1): 99-118

Structure Learning: True (partial) Covariance

- *equilibrium equation (real-valued):* $y = L^{-1}x$, where $x, y \in \mathbb{R}^p$

- *partitioned equation: (p_O observed and p_H hidden nodes)*

$$\begin{bmatrix} y_O \\ y_H \end{bmatrix} = \begin{bmatrix} L_{OO} & L_{OH} \\ L_{HO} & L_{HH} \end{bmatrix}^{-1} \begin{bmatrix} x_O \\ x_H \end{bmatrix}$$



$L_{OO} \in \mathbb{R}^{p_o \times p_o}$ encodes structure of observed-observed nodes

$L_{OH} \in \mathbb{R}^{p_o \times p_H}$ encodes structure of observed-hidden nodes

$L_{HH} \in \mathbb{R}^{p_H \times p_H}$ encodes structure of hidden-hidden nodes

- *inverse problem:* learn the sparsity pattern of L or equivalently the blocks (L_{OO}, L_{OH}, L_{HH}) from the covariance matrix (call it Σ_{OO}) of $y_O \in \mathbb{R}^{p_O}$

Partitioned Inverse Covariance Matrix

- *partitioned equation:* $\begin{bmatrix} y_O \\ y_H \end{bmatrix} = \begin{bmatrix} L_{OO} & L_{OH} \\ L_{HO} & L_{HH} \end{bmatrix}^{-1} \begin{bmatrix} x_O \\ x_H \end{bmatrix}$

- *full inverse covariance:*

$$\Omega = \begin{bmatrix} \Sigma_{OO} & \Sigma_{OH} \\ \Sigma_{HO} & \Sigma_{HH} \end{bmatrix}^{-1} = \begin{bmatrix} L_{OO} & L_{OH} \\ L_{HO} & L_{HH} \end{bmatrix} \begin{bmatrix} \Omega_{x,O} & 0 \\ 0 & \Omega_{x,H} \end{bmatrix} \begin{bmatrix} L_{OO} & L_{OH} \\ L_{HO} & L_{HH} \end{bmatrix}$$

$\Sigma_{OO} \in \mathbb{R}^{p_O \times p_O}$ is the covariance matrix of the vector $y_O \in \mathbb{R}^{p_O}$

$\Sigma_{HH} \in \mathbb{R}^{p_H \times p_H}$ is the covariance matrix of the vector $y_H \in \mathbb{R}^{p_H}$

$\Sigma_{OH} \in \mathbb{R}^{p_O \times p_H}$ is the cross-covariance matrix of y_O and y_H

- Ω contains info of L (via the two-hop graphical model); what about Ω_{OO}?

A formula for Observed Inverse Covariance

- *observed inverse covariance:*

$$\Omega_{OO} \triangleq \Sigma_{OO}^{-1} = K_{OO} - K_{OH}K_{HH}^{-1}K_{HO}$$

// Schur's complement (exercise)

where Ω_{OO} is the Schur's complement of the full inverse covariance matrix Ω

$$\Omega = \begin{bmatrix} K_{OO} & K_{OH} \\ K_{HO} & K_{HH} \end{bmatrix} \triangleq \begin{bmatrix} \Sigma_{OO} & \Sigma_{OH} \\ \Sigma_{HO} & \Sigma_{HH} \end{bmatrix}^{-1}$$

- Ω is a graphical model for all node potentials; Ω_{OO} is such a model for observed nodes
- K_{OO} is the conditional graphical model (write conditional density for the observed to see this)
- $K_{OH}K_{HH}^{-1}K_{HO}$ is called the marginal effects (no nice graphical model interpretation)

Summary of Graphical Models

- *full inverse covariance:* (right-hand side expression of inverse covariance in slide 15)

$$\begin{bmatrix} K_{OO} & K_{OH} \\ & K_{HH} \end{bmatrix} = \begin{bmatrix} L_{OO}\Omega_{x_O,x_O}L_{OO} + L_{OH}\Omega_{x_H,x_H}L_{HO} & L_{OO}\Omega_{x_O,x_O}L_{OH} + L_{OH}\Omega_{x_H,x_H}L_{HH} \\ & L_{HO}\Omega_{x_O,x_O}L_{OH} + L_{HH}\Omega_{x_H,x_H}L_{HH} \end{bmatrix}$$

- *observed inverse covariance:*

$$\Omega_{OO}$$ *// observed graphical model*

$$=$$

$$K_{OO} - K_{OH}K_{HH}^{-1}K_{HO}$$ *// conditional graphical model - marginal effects*

$$=$$

$$\underbrace{L_{OO}\Omega_{x_O,x_O}L_{OO}}_{\text{obsv-obsv}} + \underbrace{L_{OH}\Omega_{x_H,x_H}L_{HH}}_{\text{obsv-hid}} - K_{OH}K_{HH}^{-1}K_{HO}$$ *// conditional (using the formula on top) - marginal*

Exercises: Conditional Graphical Model

❧ **Theorem 3** (real-case [2]) the conditional graphical model for observed nodes in an infrastructure network includes edges between observed nodes at (ii) neighbors; and (iii) two-hop neighbors

❧ **Theorem 4** (complex-lifted-case [2]) the conditional graphical model for real and imaginary observed potentials in infrastructure network includes edges between real and imaginary potentials at (i) the self nodes; (ii) neighbors; and (iii) two-hop neighbors

- theorems give edge connections of conditional graphical model (K_{OO}) associated with the observed nodes.

- alert: conditional graphical model is not obtained by deleting edges between hidden-hidden and observed-hidden nodes in the full graphical model!

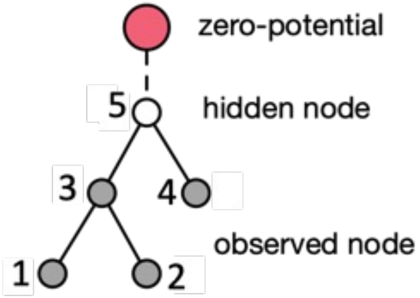
- similar theorems hold for the observed graphical model (Ω_{OO}) associated with the observed nodes (see [3])

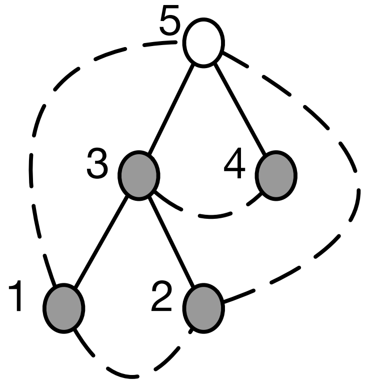
[2] R. Anguluri et. al. (2021) Grid topology identification with hidden nodes via structured norm minimization, IEEE Control Systems Letters, 6, 1244-1249

[3] H. Doddi et. al. (2020) Learning partially observed meshed distribution grids IEEE PMAPS, pp. 1-6

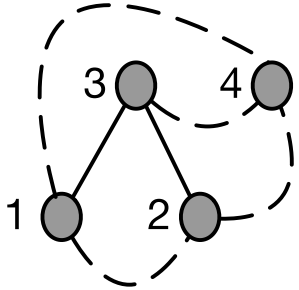
Infrastructure → Conditional Graphical Model

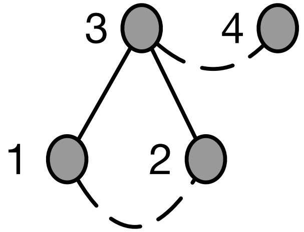
- *infrastructure → full graphical model* (*Theorem 1*)

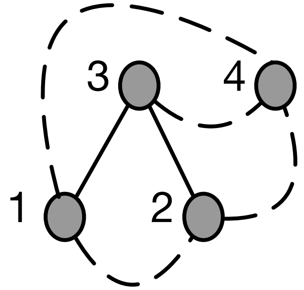




- *infrastructure → observed graphical model (see [3]; didn't discuss in our lecture)*



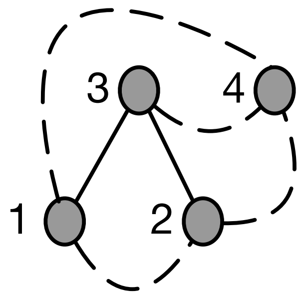


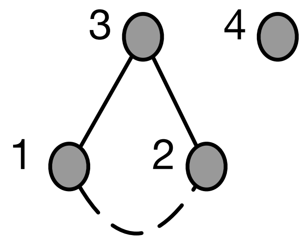


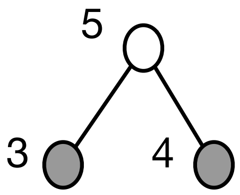
$$\Omega_{OO} = K_{OO} - K_{OH}K_{HH}^{-1}K_{HO}$$

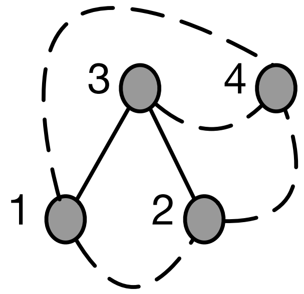
Infrastructure → Conditional Graphical Model

- *infrastructure → conditional model* $K_{OO} = S + M$ *(Theorem 3 on slide 18)*









Ω_{OO}

$$\overset{S}{\underbrace{L_{OO}\Omega_{x_O,x_O}L_{OO}}_{\text{obsv-obsv}}} + \overset{M}{\underbrace{L_{OH}\Omega_{x_H,x_H}L_{HH}}_{\text{obsv-hid}}} - K_{OH}K_{HH}^{-1}K_{HO}$$

- if $L_{OH} = 0$ *(no hidden)*, then $\Omega = \Omega_{OO}$ (the graphical model coincide with the result in Thm 2)
- the conditional model $K_{OO} = S + M$ contains information about L_{OO} and L_{OH}

Conditional Graphical Model → Infrastructure

❧ *problem:* learn infrastructure from the conditional model

Ref [3] (see footnote on slide 18) proposed an algorithm using K_{OO}

Ref [2] (see footnote on slide 18) proposed an algorithm using S and M where $K_{OO} = S + M$

❧ assumption 1 we know matrices S and M (later on how to obtain these)

❧ assumption 2 hidden nodes in the infrastructure are not adjacent to each other (like a line) nor they are leaves (end nodes)

Conditional Graphical Model → Infrastructure

- *sign-based algorithms:* consider the infrastructure network $\mathcal{G}$ with *minimum cycle length* greater than three. Then show that algorithms below work (exercise)

network reconstruction : real-case [1]

input: matrices S and M and threshold $\tau > 0$

output: graph $\mathcal{G}$

1. for all nodes i and j do
2. if $S(i,j) < -\tau$ then
3. insert edge (i,j) in $\mathcal{G}$
4. end if
5. if $M(i,j) > \tau$ then
6. insert a node k in $\mathcal{G}$
7. insert path $i - k - j$ in $\mathcal{G}$
8. end if
9. end for

- for complex-lift-case, similar algorithm holds; details in [2]

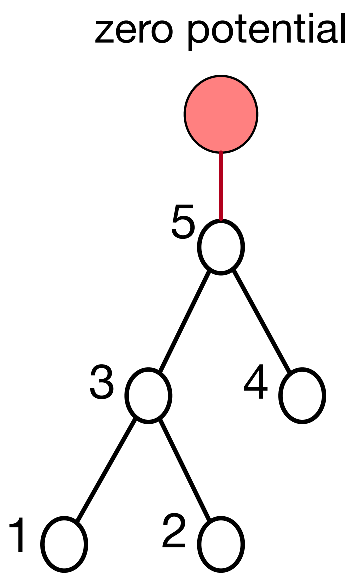
[2] R. Anguluri et. al. (2021) Grid topology identification with hidden nodes via structured norm minimization, IEEE Control Systems Letters, 6, 1244-1249

Structure Learning: Sample Covariance

- *assumption 1*: reduced-order Laplacian matrix
- *equilibrium equation:* $y = L^{-1}x$
- *injections:* $x \sim \mathcal{N}(0, \Sigma_x)$
- *potentials:* $y \sim \mathcal{N}(0, \Sigma)$, where $\Sigma = L^{-1}\Sigma_x L^{-1}$
- *assumption 2*: Σ_x is diagonal but UNKNOWN
- *learning problems*

(i) estimate the sparsity of L from finite i.i.d data $y_1, \dots, y_K$ \`

(ii) estimate the sparsity of L from finite observed i.i.d data $y_{O,1}, \dots, y_{O,K}$

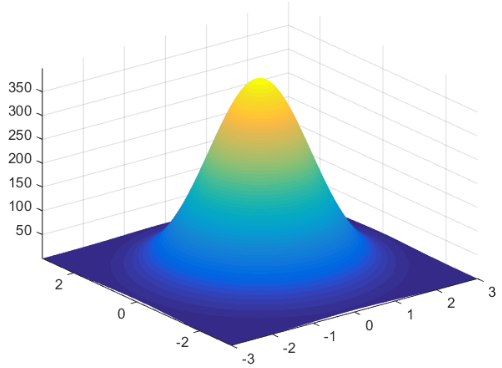


Maximum Likelihood Estimate of $\Omega = \Sigma^{-1}$

- let $y \sim \mathcal{N}(0, \Omega^{-1})$, where Ω is the $p \times p$ inverse covariance matrix

- probability density function:

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



- unconstrained MLE of Ω based on i.i.d $y_1, \ldots, y_K$

$$\max_{\Omega \succ 0} \underbrace{f(y_1) f(y_2) \ldots f(y_K)}_{\ell(S_K;\Omega)}$$

Exercise problems

- $(K \geq p)$ MLE: $\widehat{\Omega} = S_K^{-1}$
- $(K < p)$ MLE does not exist

Maximum Likelihood Estimate of $\Omega = \Sigma^{-1}$

- the rescaled log-likelihood is

$$\log \ell(\Omega; S_K) = \frac{1}{K}\sum_{k=1}^{K} \log f(y_i) = \log\det(\Omega) - \mathrm{Tr}(S_K \Omega_y)$$

- S_K is the sample covariance matrix; and

$$\log\det(\Omega) = \begin{cases} \sum_{j=1}^{p} \log\left(\lambda_j(\Omega)\right) & \text{if } \Omega \succ 0 \\ -\infty; & \text{otherwise} \end{cases}$$

- $\lambda_j(\Omega)$ is the j-th eigenvalue of Ω

Constrained Maximum Likelihood Estimator

- constrained MLE:

$$\arg\max_{\Omega \succ 0} \{\log\det(\Omega) - \mathrm{trace}(S\Omega)\}$$

$$\text{subject to } \mathcal{C}(\Omega) \leq t$$

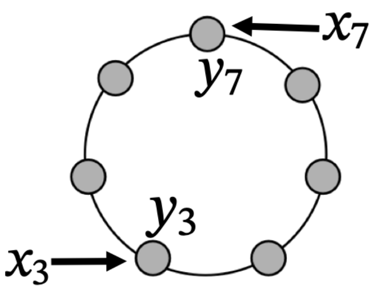
- regularized MLE:

$$\arg\max_{\Omega \succ 0} \{\log\det(\Omega) - \mathrm{trace}(S\Omega)\} + \lambda(t)\mathcal{R}(\Omega)$$

  - the constraints $C(\Omega)$ and regularization $R(\Omega)$ depends on what type of estimate we want to obtain
  - similar optimization problem could be written for observed inverse covariance matrix (next lecture)

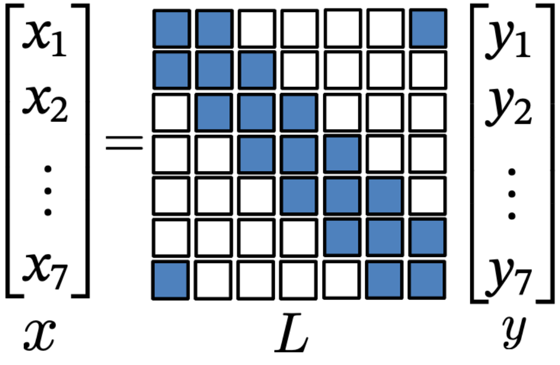
Structure Learning Problems: Models

Network Structure = Laplacian's Sparsity Pattern



infrastructure network

sparsity (zero & non-zero) of L captures network connections



nodal injections

network Laplacian

node potentials

- ***measurables:*** p-dim vectors x and y
- ***full coverage:*** access x or/and y
- ***partial coverage:*** sub-vectors of x or/and y

- ***linear model:***

$$X = LY + E \qquad \text{matrix-valued}$$

$$\mathrm{Vec}(X) = H(Y)\,\mathrm{Ve}(L) + \mathrm{Vec}(E) \qquad \text{vector-valued}$$

- ***covariance models:***

$$\Omega = L\Omega_x L \qquad \text{full coverage}$$

$$\Omega_{OO} = K_{OO} - K_{OH}K_{HH}^{-1}K_{HO} \qquad \text{partial coverage}$$

Structure Learning Problems: Wrap up

- full coverage models (linear or covariance) are easy to deal
- partial coverage models are difficult and needs more assumptions
- **covariance models (recipe)**
  - start with the infrastructure network and obtain graphical models (inv. cov)
  - make assumptions to recover network structure from graphical models
  - use optimization techniques to obtain graphical models from samples
- next lecture estimating inverse covariance matrices using MLE theory

To learn more…

Contact: rangulur@asu.edu https://rajanguluri.github.io (lecture notes: coming soon)

Selective Papers:

- Y. Yuan et. al. (2022) Inverse power flow problem, IEEE Trans on Control of Network Systems, 10(1)
- O. Ardakanian et. al. (2019) On Identification of Distribution Grids, IEEE Trans on Control of Network Systems, 6(3)
- *D. Deka et. al. (2023) Learning distribution grid topologies, IEEE Trans on Smart Grid, 15(1)
- *G. Cavraro et. al. (2021) Learning power grid topologies, in Advanced Data Analytics in Power Systems, Cambridge (link: click)
- F. Dorfler and F. Bullo (2012) Kron reduction of graphs with applications to electrical networks, IEEE Trans on Circuits on Systems, 60(1)
- S. Bolognani (2013) Identification of power distribution network topology via voltage correlation analysis, IEEE CDC, 2013
- R. Anguluri (2023) Grid topology identification with hidden nodes via structured norm minimization, IEEE CSS Letters, 6: 1244-1249

*tutorial papers