Econometrics **May 26, 2023**

Topic 16: Graphical Network Inference

by Sai Zhang

Key points:

Disclaimer: *The note is built on Prof. Jinchi Lv's lectures of the course at USC, DSO 607, High-Dimensional Statistics and Big Data Problems.*

16.1 Motivation

Consider a classic question: For n observations of dimension p, how can we capture the statistical relationships between the variables of interest? Consider the example of the multivariate Gaussian distribution:

Example 16.1.1: Multivariate Gaussian Distribution

Suppose we have n observations of dimension p, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. let $\mathbf{S}$ be the empirical covariance matrix. Then the probability density

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

define the **inverse covariance matrix** or **precision matrix** as $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$, then we have

$$f_{\mu,\boldsymbol{\Omega}} = \exp\left\{\boldsymbol{\mu}'\boldsymbol{\Omega}\boldsymbol{x} - \left\langle \boldsymbol{\Omega}, \frac{1}{2}\mathbf{x}\mathbf{x}' \right\rangle - \frac{p}{2}\log(2\pi) + \frac{1}{2}\log\det(\boldsymbol{\Omega}) - \frac{1}{2}\boldsymbol{\mu}'\boldsymbol{\Omega}\boldsymbol{\mu}\right\}$$

where $\langle \mathbf{A}, \mathbf{B} \rangle = \operatorname{tr}(\mathbf{A}\mathbf{B})$.

In this example, we know that **every** multivariate Gaussian distribution can be represented by a pairwise **Gaussian Markov Random Field (GMRF)**, which an **undirected graph** $G = (V, E)$

- representing the collection of variables $\mathbf{x}$ by a vertex set $\mathcal{V} = \{1, \cdots, p\}$
- encoding correlations between variables by a set of edges $\mathcal{E} = \{(i,j) \in \mathcal{V} \mid i =\neq j, \Omega_{ij} \neq 0\}$

For simplicity, we normalize $\boldsymbol{\mu} = \mathbf{0}$. If we draw n i.i.d. samples $\mathbf{x}_1, \cdots, \mathbf{x}_n \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, then the log-likelihood is

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Omega}) &= \frac{1}{n}\sum_{i=1}^{n}\log f(\mathbf{x}_i) = \frac{1}{2}\log\det(\boldsymbol{\Omega}) - \frac{1}{2n}\sum_{i=1}^{n}\mathbf{x}_1'\boldsymbol{\Theta}\mathbf{x}_i \\ &= \frac{1}{2}\log\det(\boldsymbol{\Omega}) - \frac{1}{2}\left\langle \Omega, \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i'\mathbf{x}_i' \right\rangle\end{aligned}$$

What's the goal? We want to estimate a **sparse** graph structure given $n \ll p$ i.i.d. observations. But what does sparsity means in this context? A sparse graph is **equivalent** to a sparse precision matrix: the precision

matrix should have many 0s.

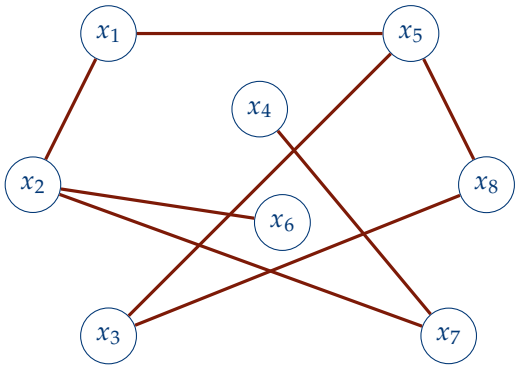
Sparse precision matrix for the Gaussian vector mentioned above $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, we have $\forall u, v$

$$x_u \perp x_v \mid \mathbf{x}_{\mathcal{V}\setminus\{u,v\}} \Leftrightarrow \Omega_{u,v} = 0$$

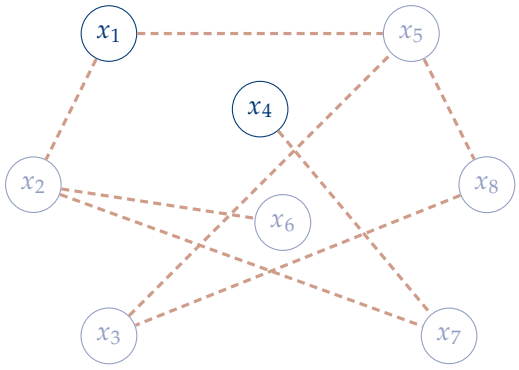
that is, sparsity of the precision matrix is equivalent to **conditional independence**[1]. Consider a graph, where x_1 and x_4 are only connected through other nodes, that is x_1 and x_4 are conditional independent, then we can have the precision matrix be something like:

$$\mathbf{\Theta} = \begin{bmatrix} * & * & 0 & 0 & * & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & * & * & 0 \\ 0 & 0 & * & 0 & * & 0 & 0 & * \\ 0 & 0 & 0 & * & 0 & 0 & * & 0 \\ * & 0 & * & 0 & * & 0 & 0 & * \\ 0 & * & 0 & 0 & 0 & * & 0 & 0 \\ 0 & * & 0 & * & 0 & 0 & * & 0 \\ 0 & 0 & * & 0 & * & 0 & 0 & * \end{bmatrix}$$

where 0 captures precisely the conditional independence.



x_1 **and** x_4 **are** ***connected***



x_1 **and** x_4 **are** ***NOT connected*****, conditionally**

Intuitively, a sparse graph is much simpler, which is why conditional independence is desired. So how to achieve sparsity? We can again use a L-1 regularization when maximizing the log-likelihood $\mathcal{L}(\mathbf{\Omega})$. Denote the sample covariance matrix as $\mathbf{S} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i'$, then the problem becomes the so-called **Graphical Lasso**

$$\max_{\mathbf{\Omega} \succeq \mathbf{0}} \log\det(\mathbf{\Omega}) - \operatorname{tr}(\mathbf{S\Omega}) - \rho \|\mathbf{\Omega}\|_1$$

which is equivalent to

$$\min_{\mathbf{\Omega} \succeq \mathbf{0}} -\log\det(\mathbf{\Omega}) + \operatorname{tr}(\mathbf{S\Omega}) + \rho \|\mathbf{\Omega}\|_1$$

16.2 Graphical Lasso

The graphical lasso method is developed by (Friedman et al., 2008). For the optimization problem

$$\min_{\mathbf{\Omega} \succeq \mathbf{0}} -\log\det(\mathbf{\Omega}) + \operatorname{tr}(\mathbf{S\Omega}) + \rho \|\mathbf{\Omega}\|_1 \tag{16.1}$$

[1] Meanwhile, for independence: $\Sigma_{u,v} = 0 \Leftrightarrow x_u \perp x_v$

The first-order optimality condition gives

$$\begin{aligned} &\mathbf{0} \in \boldsymbol{\Omega}^{-1} - \mathbf{S} - \lambda \partial \|\boldsymbol{\Omega}\|_1 \\ \Rightarrow &\boldsymbol{\Omega}^{-1}_{i,i} = \mathbf{S}_{i,i} + \lambda,\ 1 \le i \le p \qquad \text{in diagonal entries (self-loop), } 1 \in \partial\,|\boldsymbol{\Omega}_{i,i}| \end{aligned}$$

The idea is to repeatedly cycle through all columns-rows and in each step optimize only a single column-row. Denote a working version of $\boldsymbol{\Omega}^{-1}$ as $\mathbf{W}$, consider the following partition where all matrices are partitioned into one column/row versus the rest

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{\Omega}_{1,1} & \boldsymbol{\omega}_{1,2} \\ \boldsymbol{\omega}'_{1,2} & \omega_{2,2} \end{pmatrix} \qquad \mathbf{S} = \begin{pmatrix} \mathbf{S}_{1,1} & \mathbf{s}_{1,2} \\ \mathbf{s}'_{1,2} & s_{2,2} \end{pmatrix} \qquad = \begin{pmatrix} \mathbf{W}_{1,1} & \mathbf{w}_{1,2} \\ \mathbf{w}'_{1,2} & w_{2,2} \end{pmatrix}$$

where $\boldsymbol{\Omega}_{1,1}$ is $(p-1)\times(p-1)$, $\boldsymbol{\omega}_{1,2}$ is $(p-1)\times 1$, $\omega_{2,2}$ is scalar.

Consider a **blockwise** step: suppose we fix all but the last row/column, then

References

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.