Econometrics **May 29, 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 $\langle \mathbf{A}, \mathbf{B} \rangle = \operatorname{tr}(\mathbf{AB})$.

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.

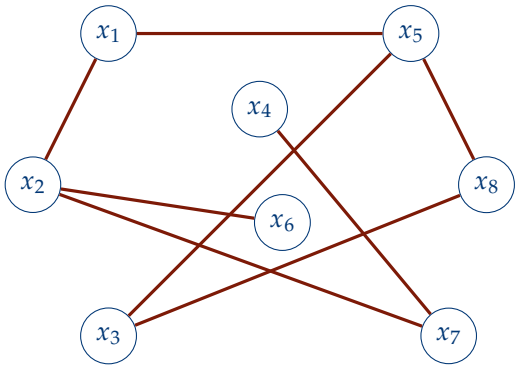
Sparse precision matrix for the Gaussian vector mentioned above $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, we have $\forall u, v$

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

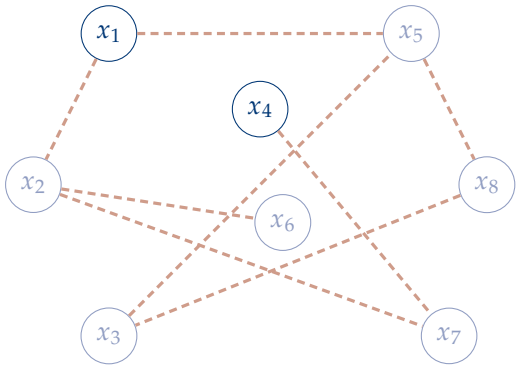
that is, sparsity of the precision matrix is equivalent to **conditional independence**[1]. Consider a graph, where x_1 and x_4 are only connected through other nodes, that is x_1 and x_4 are conditional independent, then we can have the precision matrix be something like:

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

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

$$\mathbf{0} \in \boldsymbol{\Omega}^{-1} - \mathbf{S} - \lambda \boldsymbol{\Gamma}$$

where $\boldsymbol{\Gamma}$ is a matrix of component-wise signs of $\boldsymbol{\Omega}$

$$\boldsymbol{\Gamma} = \partial \|\boldsymbol{\Omega}\|_1 \Rightarrow \gamma_{jk} \begin{cases} = \text{sign}(\omega_{jk}), & \omega_{jk} \neq 0 \\ \in [-1, 1], & \omega_{jk} = 0 \end{cases}$$

since in a graph, we always have that, following the global stationary conditions, $\omega_{jj} > 0$, which implies that

$$w_{ii} = s_{ii} + \lambda \qquad\qquad i = 1, \cdots, p \tag{16.2}$$

where we denote a working version of $\boldsymbol{\Omega}^{-1}$ as $\mathbf{W}$.

The idea is to repeatedly cycle through all columns-rows and in each step optimize only a single column-row. Consider the following partition where all matrices are partitioned into one column/row versus the rest

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{\Omega}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{12}' & \omega_{22} \end{pmatrix} \qquad \mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{s}_{12} \\ \mathbf{s}_{12}' & s_{22} \end{pmatrix} \qquad \mathbf{W} = \begin{pmatrix} \mathbf{W}_{11} & \mathbf{w}_{12} \\ \mathbf{w}_{12}' & w_{22} \end{pmatrix} \qquad \boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\Gamma}_{11} & \boldsymbol{\gamma}_{12} \\ \boldsymbol{\gamma}_{12}' & \gamma_{22} \end{pmatrix}$$

apply this partition to the optimality condition, get

$$\boldsymbol{\Omega}^{-1} = \mathbf{S} - \lambda \boldsymbol{\Gamma}$$
$$\begin{pmatrix} \mathbf{W}_{11} & \mathbf{w}_{12} \\ \mathbf{w}_{12}' & w_{22} \end{pmatrix} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{s}_{12} \\ \mathbf{s}_{12}' & s_{22} \end{pmatrix} + \lambda \begin{pmatrix} \boldsymbol{\Gamma}_{11} & \boldsymbol{\gamma}_{12} \\ \boldsymbol{\gamma}_{12}' & \gamma_{22} \end{pmatrix}$$

where $\boldsymbol{\Omega}_{11}$ is $(p-1) \times (p-1)$, $\boldsymbol{\omega}_{12}$ is $(p-1) \times 1$, ω_{22} is a scalar.

Consider a **blockwise** step: suppose we fix all but the last row/column, then using properties of inverses of block-partitioned matrices, we have

$$\begin{aligned} \begin{pmatrix} \mathbf{W}_{11} & \mathbf{w}_{12} \\ \mathbf{w}_{12}' & w_{22} \end{pmatrix} &= \begin{pmatrix} \left(\boldsymbol{\Omega}_{11} - \frac{\boldsymbol{\omega}_{12}\boldsymbol{\omega}_{12}'}{\omega_{22}}\right)^{-1} & -\mathbf{W}_{11}\frac{\boldsymbol{\omega}_{12}}{\omega_{22}} \\ & \frac{1}{\omega_{22}} - \frac{\boldsymbol{\omega}_{12}'\mathbf{W}_{11}\boldsymbol{\omega}_{12}}{\omega_{22}^2} \end{pmatrix} \\ &= \begin{pmatrix} \boldsymbol{\Omega}_{11}^{-1} + \frac{\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}\boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}}{\omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}} & -\frac{\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}}{\omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}} \\ & \frac{1}{\omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}} \end{pmatrix} \end{aligned}$$

then, by the partitioned optimality condition, we have[2]:

$$\mathbf{0} = -\mathbf{w}_{12} + \mathbf{s}_{12} + \lambda\boldsymbol{\gamma}_{12} = \mathbf{W}_{11}\frac{\boldsymbol{\omega}_{12}}{\omega_{22}} + \mathbf{s}_{12} + \lambda\boldsymbol{\gamma}_{12} \tag{16.3}$$

$$\mathbf{0} = \frac{\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}}{\omega_{22} - \boldsymbol{\omega}_{12}'\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12}} + \mathbf{s}_{12} + \lambda\boldsymbol{\gamma}_{12} = w_{22}\boldsymbol{\Omega}_{11}^{-1}\boldsymbol{\omega}_{12} + \mathbf{s}_{12} + \lambda\boldsymbol{\gamma}_{12} \tag{16.4}$$

The graphic Lasso algorithm them solves Eq.16.3 for $\boldsymbol{\beta} = \boldsymbol{\omega}_{12}/\omega_{12}$, that is

$$\mathbf{W}_{11}\boldsymbol{\beta} + \mathbf{s}_{12} + \lambda\boldsymbol{\gamma}_{12} = \mathbf{0}$$

[2]For Eq.16.4, by Eq.16.2, we know that $w_{22} = s_{22} + \lambda$, which is fixed.

where $\gamma_{12} \in \text{sign}(\boldsymbol{\beta})$ since $\omega_{22} > 0$, which is essentially solving:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{p-1}} \left\{ \frac{1}{2} \boldsymbol{\beta}' \mathbf{W}_{11} \boldsymbol{\beta} + \boldsymbol{\beta}' \mathbf{s}_{12} + \lambda \left\| \boldsymbol{\beta} \right\|_1 \right\}$$

and $\mathbf{W}_{11} > 0$ is assumed to be fixed.

This problem is analogous to a lasso regression problem of the **last variable** on **the rest**, but the cross-product matrix $\mathbf{S}_{11}$ is replaced by its **current estimation** $\mathbf{W}_{11}$. It is relatively easier to solve using elementwise coordinate descent, then

$$\begin{aligned}
&\mathbf{w}_{12} = -\mathbf{W}_{11} \frac{\boldsymbol{\omega}_{12}}{\omega_{22}} && \Rightarrow \hat{\mathbf{w}}_{12} = -\mathbf{W}_{11} \hat{\boldsymbol{\beta}} && \text{Step 1} \\
&w_{22} = \frac{1}{\omega_{22}} - \frac{\boldsymbol{\omega}_{12}' \mathbf{W}_{11} \boldsymbol{\omega}_{12}}{\omega_{22}^2} && \Rightarrow \frac{1}{\hat{\omega}_{22}} = w_{22} - \hat{\boldsymbol{\beta}}' \hat{\mathbf{w}}_{12} && \text{Step 2} \\
&\boldsymbol{\omega}_{12} = -\mathbf{W}_{11}^{-1} \mathbf{w}_{12} \omega_{22} && \Rightarrow \hat{\boldsymbol{\omega}}_{12} = -\mathbf{W}_{11}^{-1} \hat{\mathbf{w}}_{12} \hat{\omega}_{22} && \text{Step 3}
\end{aligned}$$

notice that after solving for $\boldsymbol{\beta}$ and updating $\mathbf{w}_{12}$ in Step 1, the graphic Lasso procedure can move onto the next block, that is, only Step 1 is used in the loop, Step 2 and 3 can be done at the end. The algorithm can be summarized as:

Algorithm 16.2.1: Graphical Lasso algorithm

1 Initialize $\mathbf{W} = \mathbf{S} + \lambda \mathbf{I}$

- Cycle around the columns repeatedly, performing the following steps till convergence:
  - a rearrange the rows/columns so that the target column is **the last** (implicitly)
  - b solve the lasso problem, starting the solution from the previous round for this column
  - c update the row/column (***off-diagonal***) of the covariance using $\hat{\mathbf{w}}_{12}$
  - d save $\hat{\boldsymbol{\beta}}$ for this column in the matrix $\mathbf{B}$

3 after convergence, for every row/column, compute the diagonal entries $\hat{\omega}_{jj}$, and covert the $\mathbf{B}$ matrix to $\boldsymbol{\Omega}$

Issues of GLasso method :

- $\boldsymbol{\theta}_{12}$ is entangled in $\mathbf{W}_{11}$, which is ***incorrectly*** treated as a constant
- after updating $\boldsymbol{\theta}_{12}$, the entire (working) covariance matrix $\mathbf{W}$ changes, but Glasso algorithm only updates $\mathbf{w}_{12}$ and $\mathbf{w}_{21}$

Together, the two issues lead to the non-monotonic behavior of Glasso in minimizing $f(\boldsymbol{\Omega})$. Next, we address these issues by introducing some modifications.

16.3 Graphical Lasso: Modifications

References

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