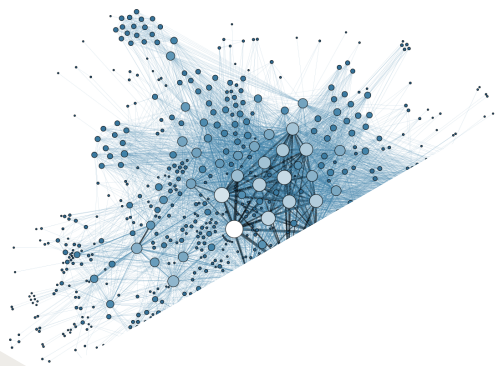


Graph Learning with ℓ_0-norm and SLOPE Penalties

Baozhuo SU

Télécom SudParis/CNAM

27/03/2026

Outline

1. Background

2. Distance based graph learning

3. Gaussian graphical model

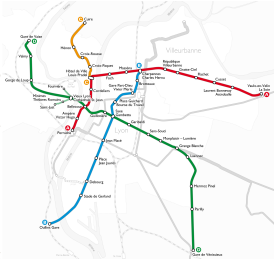
4. Glasso,Gslope ,Gl0

5. Datasets & Experiments

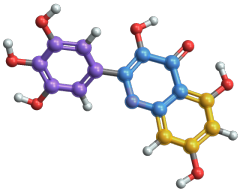
6. Conclusion

Graph everywhere

"Graphs are ubiquitous structures"



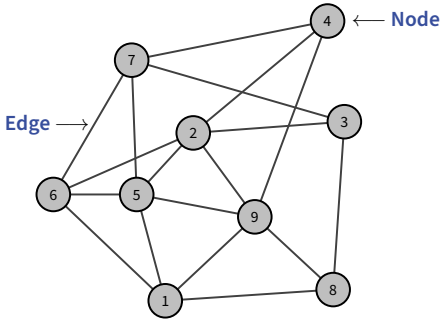




Graphs

Graph: $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \omega\}$

- **Nodes** (or **verteces**) $\mathcal{V} = \{1, \ldots, p\}$
- **Edges** $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$
- **Edge weights** $\omega : \mathcal{E} \to \mathbb{R}$
  - $w_{q,\ell} = \omega((q, \ell))$
  - $\mathbf{w} \in \mathbb{R}^{p(p-1)/2}$ stores the $w_{q,\ell}$'s



Matrix representations

- **Adjacency matrix** $\mathbf{A}$, with $[\mathbf{A}]_{q,\ell} = \begin{cases} w_{q,\ell} & \text{if } (q,\ell) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$
- **Laplacian matrix** $\mathbf{L} = \mathbf{D} - \mathbf{A}$, with **degree matrix** $\mathbf{D}$

Outline

1. Background

2. Distance based graph learning

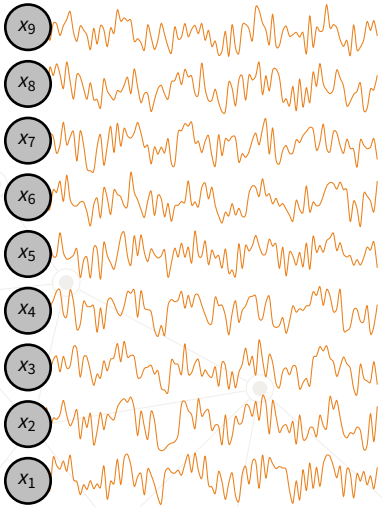
3. Gaussian graphical model

4. Glasso,Gslope ,Gl0

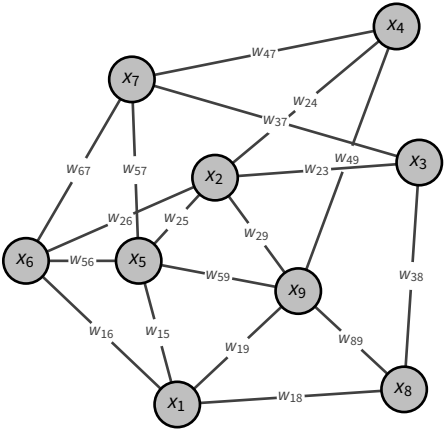
5. Datasets & Experiments

6. Conclusion

Graph learning



reconstruct



Distance based methods

Usually built from **Euclidean distance** **between samples**

- **ϵ-neighborhood**

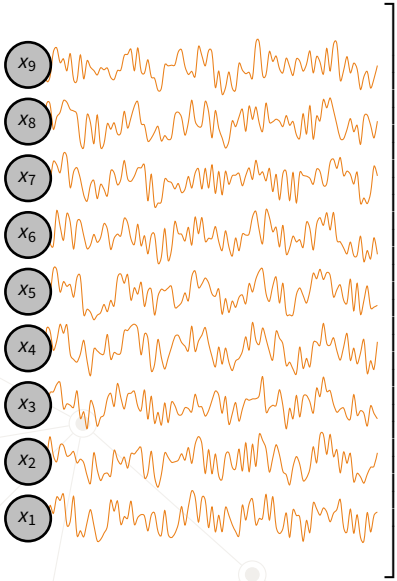
$$\mathbf{W}_{ij} = \begin{cases} 1 & \text{if } ||\mathbf{x}_i - \mathbf{x}_j||^2 \leq \epsilon \\ 0 & \textit{otherwise} \end{cases}$$

- k nearest neighbors, or **k-NNG** (symmetric, directed, mutual)

- **Weighted similarity graphs** built from **kernels**

$$\mathbf{W}_{ij} = \exp\left(-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{2\sigma^2}\right)$$

What happens if the signal is zero-mean?



$$= [\mathbf{x_1} \dots \mathbf{x_n}],\ \mathbf{x_i} \sim \mathcal{N}(\mathbf{0}, \Sigma)$$

Transition

k-NNG and **similarity graphs** built the graph from **hard distances** between entities

Can be **limiting**, *e.g.* for **zero-means signals** $\rightarrow$ **k-NNG** will degrade to random

Require **new paradigms** to handle zero-mean case.

A natural idea is to use a more **smooth criterion**:

- **Probabilistic based methods**

Outline

1. Background

2. Distance based graph learning

3. Gaussian graphical model

4. Glasso,Gslope ,Gl0

5. Datasets & Experiments

6. Conclusion

Precision matrix

In probabilistic setting, an **edge between two nodes** $\rightarrow$ a **conditional correlation relationship**.

In this case, we introduce the **precision matrix $\boldsymbol{\Theta}$**, which is defined as the inverse of covariance matrix

$$\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$$

It directly encodes the conditional dependence structure, since

$$\operatorname{corr}\left(x_i, x_j \mid x_{-\{i,j\}}\right) = -\frac{\Theta_{ij}}{\sqrt{\Theta_{ii}\Theta_{jj}}}$$

If $\Theta_{ij} = 0$, then nodes i and j are **conditionally independent**.

In practice we construct the precision matrix with Laplacian matrix $\boldsymbol{L}$.

$$\boldsymbol{\Theta} = \boldsymbol{L} + \kappa \boldsymbol{I}$$

From precision matrix to Gaussian graphical model

Gaussian graphical model(GGM) is a powerful model to reconstruct topology structure. With precision matrix $\boldsymbol{\Theta}$, it solves a **penalized log-likelihood maximization** problem

$$\underset{\boldsymbol{\Theta} \succ 0}{\text{maximize}} \quad \log\det(\boldsymbol{\Theta}) - \text{Tr}(\boldsymbol{S}\boldsymbol{\Theta}) - \lambda h(\boldsymbol{\Theta})$$

likelihood — promotes sparsity

$\lambda h(\boldsymbol{\Theta})$ is the penalty item to keep the sparsity.

The choice of the penalty has a **direct impact** on the reconstructed topology structure.

$\rightarrow$ essential to **test different penalty term**.

Outline

1. Background

2. Distance based graph learning

3. Gaussian graphical model

4. Glasso,Gslope ,Gl0

5. Datasets & Experiments

6. Conclusion

Glasso

ℓ_1 is a widely used to promote sparsity, it is also called lasso(Least Absolute Shrinkage and Selection Operator).

Let the penalty term be the lasso penalty

$$h(\boldsymbol{\Theta}) = \|\boldsymbol{\Theta}\|_{1,\text{off}}$$

then the GGM solves

$$\underset{\boldsymbol{\Theta} \succ 0}{\text{maximize}} \ \log\det(\boldsymbol{\Theta}) - \text{Tr}(\boldsymbol{S}\boldsymbol{\Theta}) - \lambda\|\boldsymbol{\Theta}\|_{1,\text{off}}$$

$\lambda > 0$ is the tuning parameter. This model is called **Glasso**.

Gslope & Gl0

Similar as Glasso, we can use different penalties to define Gslope and Gl0 models.

Gslope uses the Slope(Sorted L-One Penalty) penalty

$$\underset{\boldsymbol{\Theta}\succ 0}{\text{maximize}}\ \log\det(\boldsymbol{\Theta}) - \text{Tr}(\boldsymbol{S}\boldsymbol{\Theta}) - \sum_{i=1}^{m}\lambda_i\bar{\theta}_i$$

$$\bar{\boldsymbol{\theta}} = \text{sort}\circ|\ \text{vec}_{\text{up}}(\boldsymbol{\Theta})\ |,\ \lambda_1 \geq \cdots \geq \lambda_m \geq 0$$

Gl0 uses ℓ_0 penalty

$$\underset{\boldsymbol{\Theta}\succ 0}{\text{maximize}}\ \log\det(\boldsymbol{\Theta}) - \text{Tr}(\boldsymbol{S}\boldsymbol{\Theta}) - \lambda\|\boldsymbol{\Theta}\|_{0,\text{off}}$$

where $\|\boldsymbol{\Theta}\|_{0,\text{off}}$ denotes the number of nonzero item of $\boldsymbol{\Theta}$ of its off diagonal part.

Alternating direction method of multipliers

The **alternating direction method of multipliers**(ADMM) is an efficient solver for penalized problem. To apply ADMM, we introduce an auxiliary variable $\mathbf{Y}$

$$\min_{\boldsymbol{\Theta},\boldsymbol{Y}} \; -\log\det(\boldsymbol{\Theta}) + \mathrm{Tr}(\boldsymbol{\Theta}\boldsymbol{S}) + \mathbb{I}[\boldsymbol{\Theta} \succ 0] + h(\boldsymbol{Y})$$
$$\text{s. t. } \boldsymbol{\Theta} = \boldsymbol{Y}$$

the updating of ADMM is

Unified ADMM solver for Glasso, Gslope and Gl0

$$\boldsymbol{\Theta}^{(k+1)} = \arg\min_{\boldsymbol{\Theta}\in S_p^{++}} \left\{ -\log\det(\boldsymbol{\Theta}) + \mathrm{tr}(\boldsymbol{S}\boldsymbol{\Theta}) + \frac{\rho}{2}\|\boldsymbol{\Theta} - \boldsymbol{Y}^{(k)} + \boldsymbol{U}^{(k)}\|_F^2 \right\},$$
$$\boldsymbol{Y}^{(k+1)} = \arg\min_{\boldsymbol{Y}} \left\{ h(\boldsymbol{Y}) + \frac{\rho}{2}\|\boldsymbol{\Theta}^{(k+1)} - \boldsymbol{Y} + \boldsymbol{U}^{(k)}\|_F^2 \right\},$$
$$\boldsymbol{U}^{(k+1)} = \boldsymbol{U}^{(k)} + \boldsymbol{\Theta}^{(k+1)} - \boldsymbol{Y}^{(k+1)}.$$

Outline

1. Background

2. Distance based graph learning

3. Gaussian graphical model

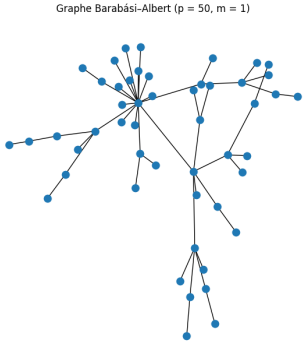
4. Glasso,Gslope ,Gl0

5. Datasets & Experiments

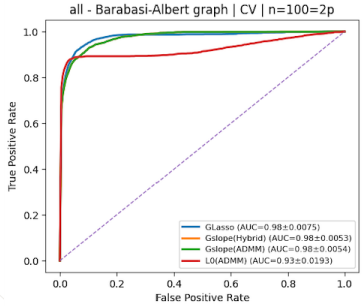
6. Conclusion

Experimental results on synthetic datasets

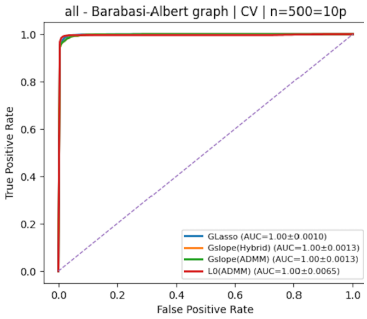
- Models are evaluated on Barabasi-Alerbt graphs with **50 Monte-Carlo**.
- Evaluated by **averaged receiver operating characteristic**(ROC) and AUC. AUC = 1 means all edges are correctly reconstructed.



(a) Generated graphs



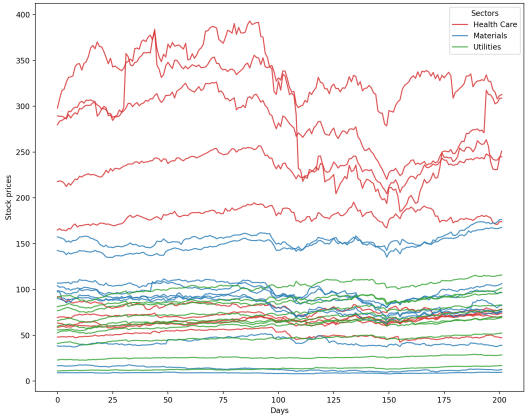
(b) ROC with $n = 2p$

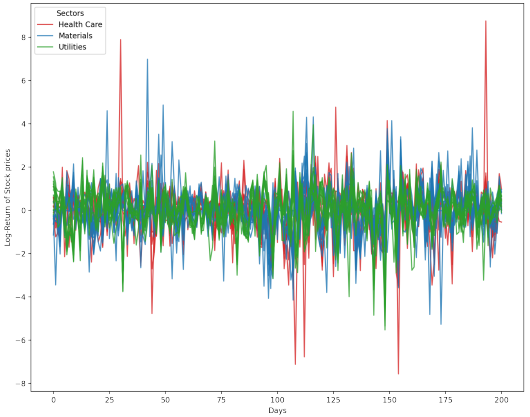


(c) ROC with $n = 10p$

S&P 500 Dataset

- $p = 30$ stocks from 3 sectors *Utilities*, *Materials*, and *Health Care*
- $n = 200$ daily observations
- Log-returns of stock prices: $\mathbf{X}_{i,j} = \log P_{i,j} - \log P_{i,j-1}$



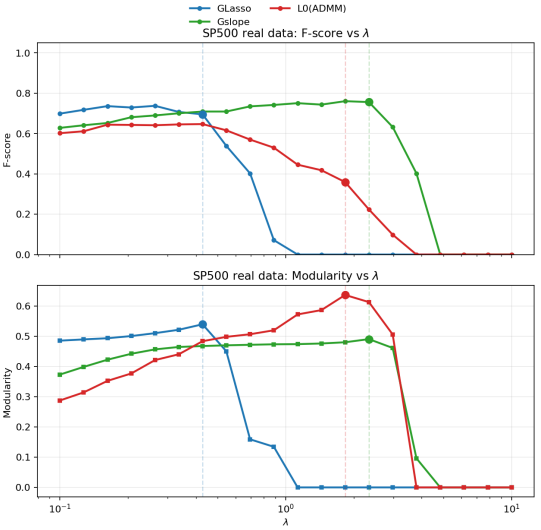


Experimental results on S&P500 dataset

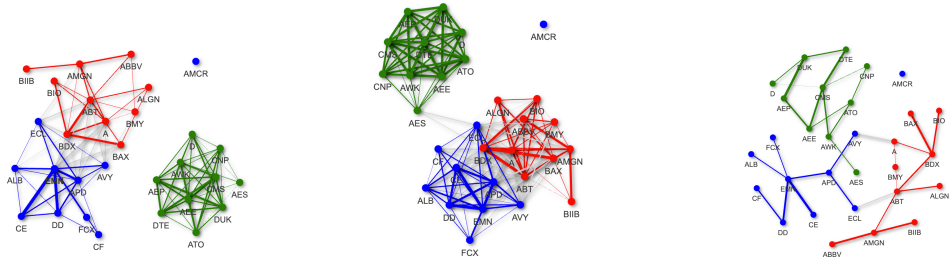
- **Parameter tuning**: 10 folds Cross-Validation
- **F-score**, defined as

$$\mathrm{FS} = \frac{2\,\mathrm{tp}}{2\,\mathrm{tp} + \mathrm{fp} + \mathrm{fn}}$$

  - tp: true positive
  - fp: false positive
  - fn: false negative
- **Modularity**: performance of graph clustering



Experimental results on S&P 500 dataset



(a) Glasso **(b)** Gslope **(c)** Gl0

Figure 2: Results of clustering. Green stands for Utilities, red for Health Care, and blue for Materials.

Outline

1. Background

2. Distance based graph learning

3. Gaussian graphical model

4. Glasso,Gslope ,Gl0

5. Datasets & Experiments

6. Conclusion

Conclusion

We generalize Gaussian graphical model by **leveraging different penalties** and define Glasso, Gslope Gl0 models.

- All three methods: Glasso, Gslope, Gl0 demonstrate a **powerful reconstruction ability.**
- Gslope is very **robust**, i.e., less sensitive to parameter tuning and demonstrates the **highest F-Score** .
- Gl0 demonstrates the **highest modularity**; encourages to **suppress the false positive ratio** and returns the **most sparse topology**.

Thanks for your attention !

Baozhuo Su





