Joint Gaussian Graphical Model Series – VII

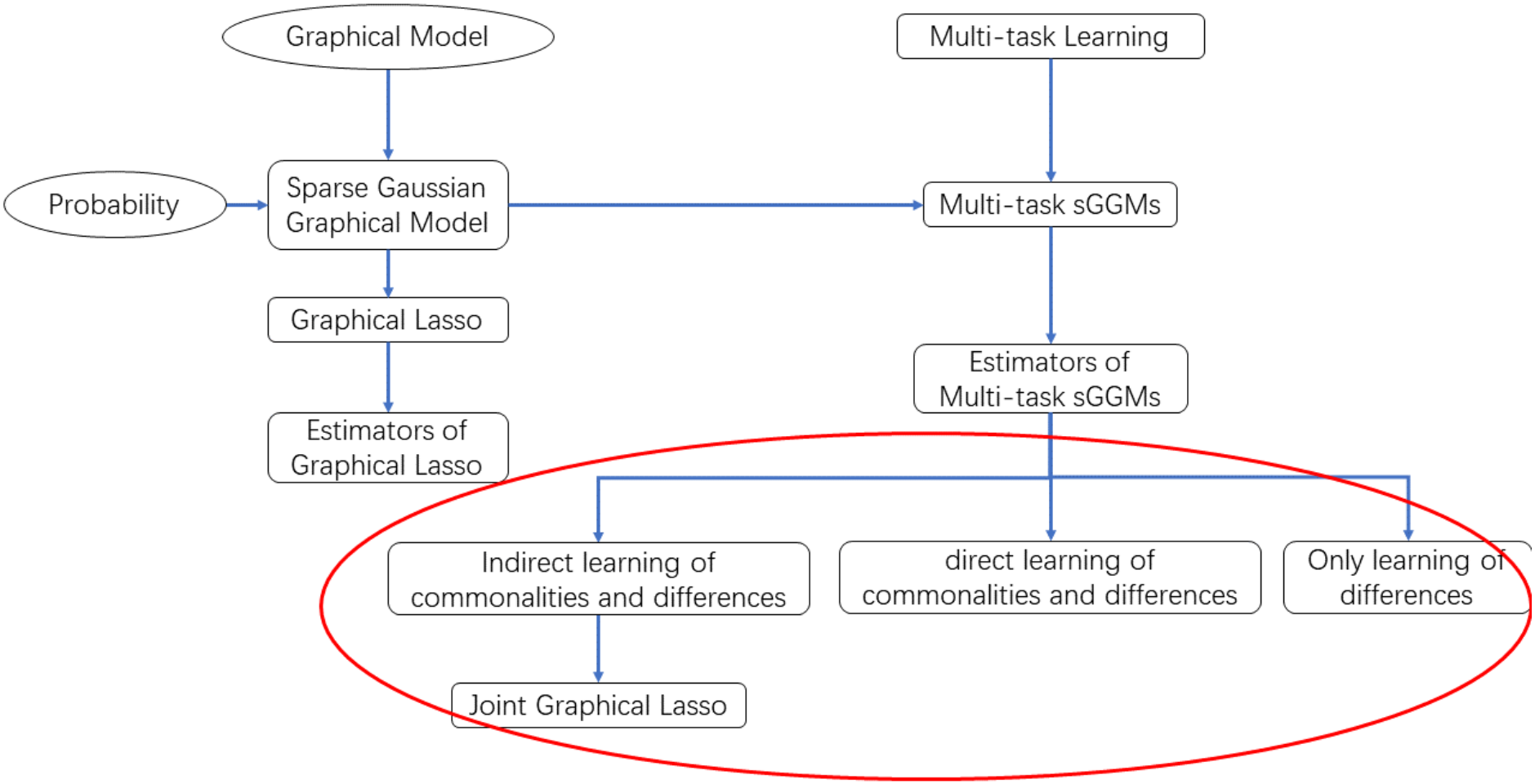
Multi-task sGGMs estimators

Beilun Wang

[1]Department of Computer Science, University of Virginia
http://jointggm.org/

August 18th, 2017

Road Map


Graphical Model
Multi-task Learning
Probability
Sparse Gaussian Graphical Model
Multi-task sGGMs
Graphical Lasso
Estimators of Multi-task sGGMs
Estimators of Graphical Lasso
Indirect learning of commonalities and differences
direct learning of commonalities and differences
Only learning of differences
Joint Graphical Lasso

Outline

1. Notation
2. Review
3. Multi-task Learning
4. Multi-task sGGMs
5. Multi-task sGGMs estimators
   - Joint Graphical Lasso
   - Directly learn the commonalities and differences among tasks
   - Directly learn the differences between case and control

Notation

Notation

$X^{(i)}$ The i-th data matrix

$\Sigma^{(i)}$ The i-th covariance matrix.

$\Omega^{(i)}$ The i-th precision matrix.

p The number of features.

n_i The number of samples in the i-th data matrix.

K The number of tasks.

Review

Review from last talk

- We introduce multi-task learning sparse Gaussian Graphical Models (sGGMs).
- We introduce the optimization chanllenges in the multi-task sGGMs.
- We introduce the ADMM method and the solution of Joint Graphical Lasso.

Review of Gaussian Graphical Model

Suppose the precision matrix $\Omega = \Sigma^{-1}$.

The log-likelihood of Ω equals to $\ln \det(\Omega) - \text{tr}\left(\Omega \widehat{S}\right)$

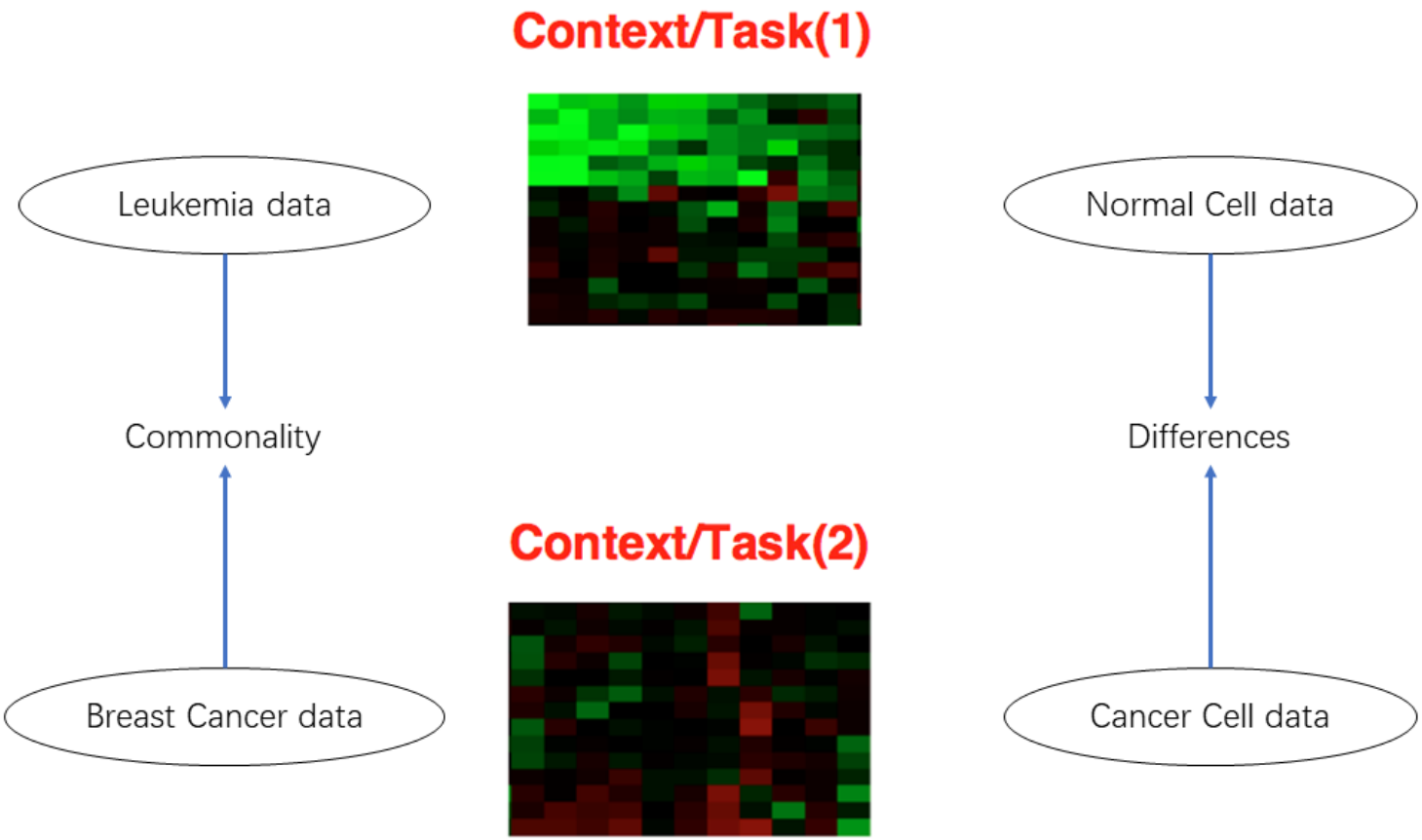
Multi-task Learning

Multi-task Learning

Multi-task Learning

Multi-task learning (MTL) is a subfield of machine learning in which multiple learning tasks are solved at the same time, while exploiting commonalities and differences across tasks.
This can result in improved learning efficiency and prediction accuracy for the task-specific models, when compared to training the models separately.

Multi-task Learning



Multi-task Learning–Linear Classifier Example

Linear Classifier

$$f(x) = \text{sgn}(w^T x + b) \tag{3.1}$$

Multi-task Linear Classifiers

For the i-th task,

$$f_i(x) = \text{sgn}((w_S^T + w_i^T)x + b) \tag{3.2}$$

Multi-task sGGMs

Multi-task sGGMs

Problem

- Input: $\{X^{(i)}\}$
- Output: $\{\Omega^{(i)}\}$
- Assumption I: Sparsity
- Assumption II: Commonalities and Differences

Multi-task sGGMs

Likelihood

$$\sum_i n_i(\ln\det(\Omega^{(i)}) - \mathrm{tr}\left(\Omega^{(i)}\widehat{S}^{(i)}\right)) \tag{4.1}$$

Likelihood with sparsity assumption

$$\underset{\Omega^{(i)}}{\mathrm{argmax}} \sum_i n_i(\ln\det(\Omega^{(i)}) - \mathrm{tr}\left(\Omega^{(i)}\widehat{S}^{(i)}\right)) \tag{4.2}$$

$$\text{Subject to: } ||\Omega^{(i)}||_1 \leq t \tag{4.3}$$

Multi-task sGGMs

Likelihood with multi-task setting

$$\underset{\Omega^{(i)}}{\operatorname{argmax}} \sum_i n_i (\ln \det(\Omega^{(i)}) - \operatorname{tr}\left(\Omega^{(i)} \widehat{S}^{(i)}\right)) \tag{4.4}$$

$$\text{Subject to: } ||\Omega^{(i)}||_1 \leq t \tag{4.5}$$

$$P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) \leq t_2 \tag{4.6}$$

Joint Graphical Lasso [Danaher et al.(2013)Danaher, Wang, and Witten]

$$\underset{\Omega^{(i)}}{\operatorname{argmin}} - \sum_i n_i (\ln \det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)} \widehat{S}^{(i)}\right)) + \lambda_1 ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \tag{4.7}$$

Multi-task sGGMs estimators

Multi-task sGGMs estimators

- Joint Graphical Lasso type estimators
- Directly learn the commonalities and differences among tasks
- Directly learn the differences between case and control

Joint Graphical Lasso estimators

Different Joint Graphical Lasso

In the end, different multi-task sGGMs estimators choose different $P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)})$.

Solutions

Most methods use ADMM as the solution of the estimators.

JGL:Problem

- Input: $\{X^{(i)}\}$
- Output: $\{\Omega^{(i)}\}$
- Assumption I: Sparsity
- Assumption II: Commonalities and Differences

Multi-task sGGMs estimators

Group Lasso[Danaher et al.(2013)Danaher, Wang, and Witten]

$$P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) = ||\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}||_{\mathcal{G},2}.$$

SIMONE[Chiquet et al.(2011)Chiquet, Grandvalet, and Ambroise]

$$P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) = \sum_{i \neq j}((\sum_{k=1}^{T}(\Omega_{ij}^{(k)})_+^2))^{\frac{1}{2}} + ((\sum_{k=1}^{K}(-\Omega_{ij}^{(k)})_+^2))^{\frac{1}{2}}.$$

Node
JGL[Mohan et al.(2013)Mohan, London, Fazel, Lee, and Witten]

$$P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) = \sum_{ij, i>j} RCON(\Omega^{(i)} - \Omega^{(j)}).$$

Node
JGL[Mohan et al.(2013)Mohan, London, Fazel, Lee, and Wit

Definition 1 *The row-column overlap norm (RCON) induced by a matrix norm $\|.\|$ is defined as*

$$\Omega(\Theta^1,\Theta^2,\ldots,\Theta^K) = \min_{V^1,V^2,\ldots,V^K} \left\| \begin{bmatrix} V^1 \\ V^2 \\ \vdots \\ V^K \end{bmatrix} \right\|$$
$$\text{subject to} \quad \Theta^k = V^k + (V^k)^T \quad \text{for } k = 1,\ldots,K.$$

Directly learn the commonalities and differences among tasks: Problem

- Input: $\{X^{(i)}\}$
- Output: $\{\Omega_I^{(i)}, \Omega_S\}$
- Assumption I: Sparsity
- Assumption II: Commonalities and Differences

Multi-task sGGMs estimators – Direct modeling

The second penalty function is still an indirect way to model the commonality and differences among tasks. Some works try to directly model this relationship.

Mixed Neighborhood Selection (MSN)[Monti et al.(2015)Monti, Anagnostopoulos, and Montana]

the neighborhood edges of a given node v in the i-task is modeled as $\beta^v + \widetilde{\boldsymbol{b}}^{(i),v}$. Here $\widetilde{\boldsymbol{b}}^{(i),v} \sim N(0, \Phi^v)$.

Consider the CLIME estimator, we can directly model the graphs as the sum of commonality and differences

SIMULE

$\Omega^{(i)} = \epsilon\Omega_S + \Omega_I^{(i)}$.

Direct modeling commonalities and differences – SIMULE

SIMULE

$$\widehat{\Omega}_I^{(1)}, \widehat{\Omega}_I^{(2)}, \ldots, \widehat{\Omega}_I^{(K)}, \widehat{\Omega}_S = \underset{\Omega_I^{(i)}, \Omega_S}{\operatorname{argmin}} \sum_i ||\Omega_I^{(i)}||_1 + \epsilon K ||\Omega_S||_1$$

Subject to: $||\Sigma^{(i)}(\Omega_I^{(i)} + \Omega_S) - I||_\infty \leq \lambda_n,\ i = 1, \ldots, K$

Multi-task sGGMs estimators – Direct modeling the differential networks: Problem

- Input: $\{X^{(i)}\}$
- Output: $\{\Delta\}$
- Assumption I: Sparse Differential networks

Multi-task sGGMs estimators – Direct modeling the differential networks I

Fused GLasso

By adding a regularization to enforce the sparsity of $\Delta = \Omega_c - \Omega_d$, we have the following formulation:

$$\underset{\Omega_c, \Omega_d \succ 0, \Delta}{\operatorname{argmin}} \ \mathcal{L}(\Omega_c) + \mathcal{L}(\Omega_d) \lambda_n (||\Omega_c||_1 + ||\Omega_d||_1) + \lambda_2 ||\Delta||_1 \qquad (5.1)$$

The Fused Lasso assumes $\Omega_{case}, \Omega_{control}, \Delta$. However, many real world applications, like brain imaging data, only assume the differential network Δ is sparse.

Direct modeling the differential networks II: Differential CLIME

A recent study proposes the following model, which only assume the sparsity of Δ.

Differential CLIME

$$\begin{aligned} &\underset{\Delta}{\operatorname{argmin}} \, ||\Delta||_1 \\ &\text{Subject to: } ||\widehat{\Sigma}_c \Delta \widehat{\Sigma}_d - (\widehat{\Sigma}_c - \widehat{\Sigma}_d)||_\infty \leq \lambda_n \end{aligned} \tag{5.2}$$

However, this method is solved by a linear programming. It has p^2 variables in this method. Therefore, the time complexity is at least $O(p^8)$. In practice, it takes more than 2 days to finish running the method when $p = 120$.

Direct modeling the differential networks III: Density Ratio

The above methods all make the Gaussian assumption. This method relaxes the model to the exponential family distribution.

Density Ratio

$$\frac{p_c(x, \theta_c)}{p_d(x, \theta_d)} \propto \exp(\sum_t \Delta_t f_t(x)) \tag{5.3}$$

Here Δ_t encodes the difference between two Networks for factor f_t.

Density Ratio

$$r(x; \theta) = \frac{1}{N(\theta)} \exp(\sum_t \Delta_t f_t(x)) \tag{5.4}$$

Here Δ_t encodes the difference between two Networks for factor f_t. $N(\theta)$ is a normalization term.

Direct modeling the differential networks III: Density Ratio

Density Ratio for Markov Random Field

$$\widehat{p}(x) = p_d(x) r(x;\theta)$$
$$\mathrm{KL}[p_c || \widehat{p}] = \text{Const.} - \int p_c(x) \log r(x;\theta) dx. \tag{5.5}$$

Summary

- We introduce the multi-task sGGMs estimators.
- We introduce the multi-task sGGMs estimators, which directly model the commonalities and differences.
- We introduce the multi-task sGGMs estimators, which directly model the differences.

References I


J. Chiquet, Y. Grandvalet, and C. Ambroise.
Inferring multiple graphical structures.
Statistics and Computing, 21(4):537–553, 2011.


P. Danaher, P. Wang, and D. M. Witten.
The joint graphical lasso for inverse covariance estimation across multiple classes.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 2013.


K. Mohan, P. London, M. Fazel, S.-I. Lee, and D. Witten.
Node-based learning of multiple gaussian graphical models.
arXiv preprint arXiv:1303.5145, 2013.

References II


R. P. Monti, C. Anagnostopoulos, and G. Montana.
Learning population and subject-specific brain connectivity networks via mixed neighborhood selection.
arXiv preprint arXiv:1512.01947, 2015.