A constrained $\ell 1$ minimization approach for estimating multiple Sparse Gaussian or Nonparanormal Graphical Models

Beilun Wang[1] Ritambhara Singh[1] Yanjun Qi[1]

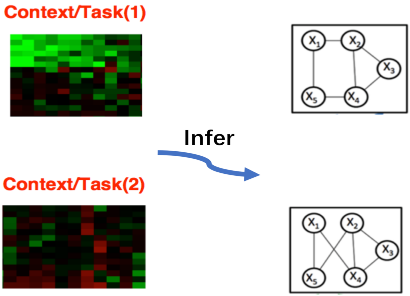
[1]University of Virginia
http://jointggm.org/

Published @ Machine Learing;

Outline

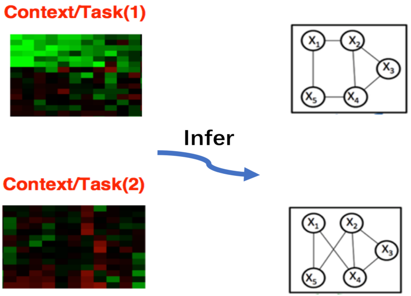
Motivation: Structure Learning from Heterogeneous Samples

- Learning relational graph structure among features/variables from an observed sample dataset is an important task in Machine Learning.



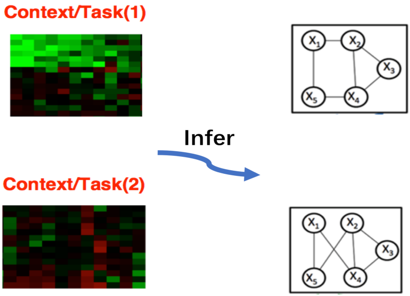
Motivation: Structure Learning from Heterogeneous Samples

- Learning relational graph structure among features/variables from an observed sample dataset is an important task in Machine Learning.
- This paper focuses on inferring graph structures from multiple related datasets (heterogeneous samples) about the same set of variables.



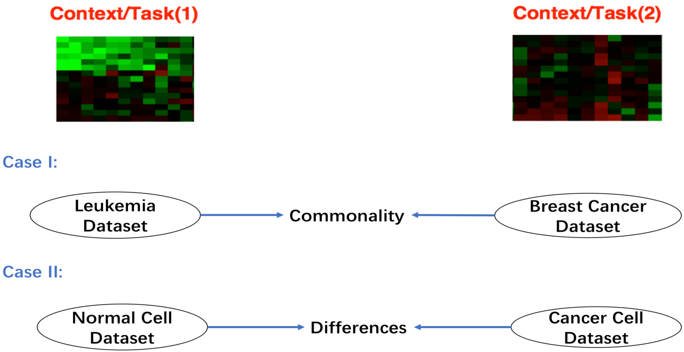
Motivation: Structure Learning from Heterogeneous Samples

- Learning relational graph structure among features/variables from an observed sample dataset is an important task in Machine Learning.
- This paper focuses on inferring graph structures from multiple related datasets (heterogeneous samples) about the same set of variables.
- We mainly focus on estimating conditional dependency graphs using the sparse Gaussian Graphical Model (sGGM).



When Working on Multiple Different but Related Datasets:

- Samples of many real applications take the form of multiple different but related data matrices.
  - Blood cancer samples vs. Breast cancer samples;
  - Normal patient samples vs. Cancel patient samples;

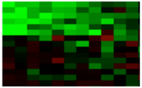


When Working on Multiple Different but Related Datasets:

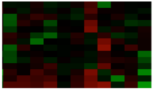
- Samples of many real applications take the form of multiple different but related data matrices.
  - Blood cancer samples vs. Breast cancer samples;
  - Normal patient samples vs. Cancel patient samples;

- A multi-task learning setting: to investigate the commonalities and differences among different datasets.

Context/Task(1)



Context/Task(2)

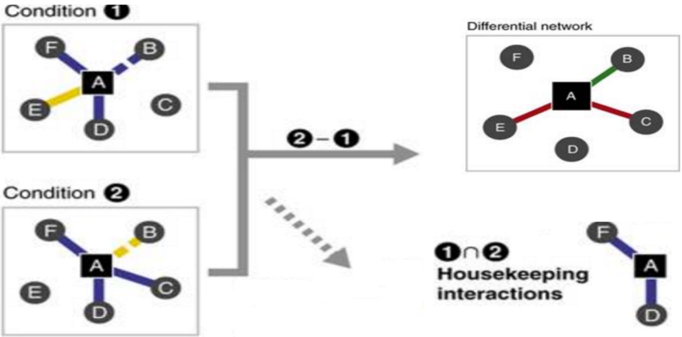


Case I:

Case II:

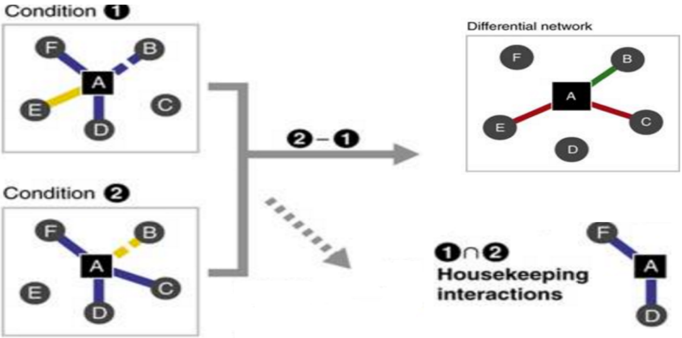
Our Aim: Shared and Task-specific Graph Structures

- We aim to obtain shared and task-specific graph structures from heterogeneous samples.



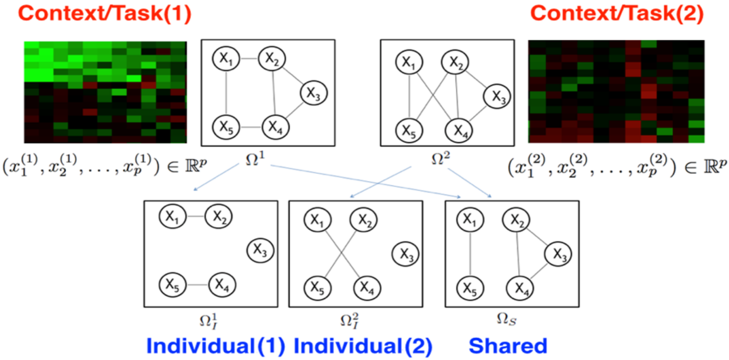
Our Aim: Shared and Task-specific Graph Structures

- We aim to obtain shared and task-specific graph structures from heterogeneous samples.
- For example, in computational biology [Ideker and Krogan(2012)] urges to estimate housekeeping interactions and differential network among genes or proteins.



Our Aim: To Learn Shared and Task-specific Graph Structures from Multiple Related Datasets

- Main Task: How to estimate / learn shared (Ω_S) and task-specific ($\Omega_I^{(i)}$) graph structures among feature variables from multiple different but related datasets about the same set of features.



Notations

X	Data matrix.
Σ	Covariance matrix.
Ω	Inverse of covariance matrix (precision matrix).
$X^{(i)}$	The i-th data matrix.
$\Sigma^{(i)}$	The i-th covariance matrix.
$\Omega^{(i)}$	The i-th precision matrix.
p	The total number of feature variables.
n_i	The number of samples in the i-th data matrix.
K	The total number of tasks.

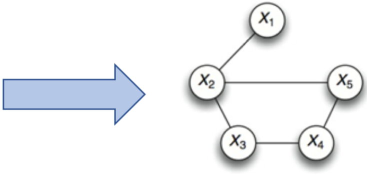
Outline

Background: Sparse Gaussian Graphical Model (sGGM)

- $X \sim N(\mu, \Sigma)$.

Inverse Covariance Matrix

$$\begin{pmatrix} 1 & 0.2 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 & 0.2 \\ 0 & 0.2 & 1 & 0.2 & 0 \\ 0 & 0 & 0.2 & 1 & 0.2 \\ 0 & 0.2 & 0 & 0.2 & 1 \end{pmatrix}$$

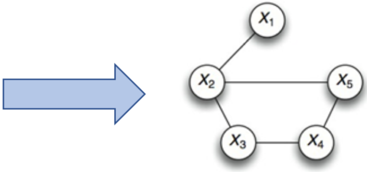


Background: Sparse Gaussian Graphical Model (sGGM)

- $X \sim N(\mu, \Sigma)$.
- Covariance matrix Σ can be calculated from X

Inverse Covariance Matrix

$$\begin{pmatrix} 1 & 0.2 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 & 0.2 \\ 0 & 0.2 & 1 & 0.2 & 0 \\ 0 & 0 & 0.2 & 1 & 0.2 \\ 0 & 0.2 & 0 & 0.2 & 1 \end{pmatrix}$$

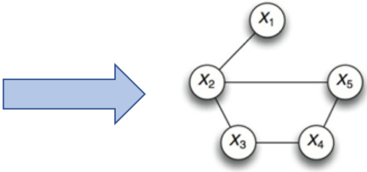


Background: Sparse Gaussian Graphical Model (sGGM)

- $X \sim N(\mu, \Sigma)$.
- Covariance matrix Σ can be calculated from X
- Precision matrix Ω is the inverse of covariance matrix Σ

Inverse Covariance Matrix

$$\begin{pmatrix} 1 & 0.2 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 & 0.2 \\ 0 & 0.2 & 1 & 0.2 & 0 \\ 0 & 0 & 0.2 & 1 & 0.2 \\ 0 & 0.2 & 0 & 0.2 & 1 \end{pmatrix}$$



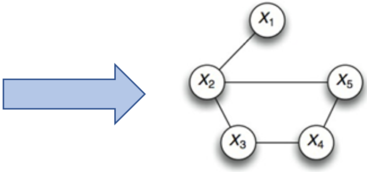
Background: Sparse Gaussian Graphical Model (sGGM)

- $X \sim N(\mu, \Sigma)$.
- Covariance matrix Σ can be calculated from X
- Precision matrix Ω is the inverse of covariance matrix Σ

- The sparsity pattern of Ω captures the conditional dependency pattern among variables.
- For example,

Inverse Covariance Matrix

$$\begin{pmatrix} 1 & 0.2 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 & 0.2 \\ 0 & 0.2 & 1 & 0.2 & 0 \\ 0 & 0 & 0.2 & 1 & 0.2 \\ 0 & 0.2 & 0 & 0.2 & 1 \end{pmatrix}$$



- Traditionally, we estimate sGGM from samples (of a single task) using an ℓ_1 penalized MLE formulation.

Graphical Lasso
[Friedman et al.(2008)Friedman, Hastie, and Tibshirani]

$$\underset{\Omega}{\operatorname{argmin}} - \ln \det(\Omega) + \operatorname{tr}\left(\Omega\widehat{\Sigma}\right) + \lambda_n ||\Omega||_1 \tag{1.1}$$

Previous Methods: Joint Graphical Lasso (JGL) for Jointly Estimating Multiple sGGMs

- Most previous studies add a second penalty function $P()$ into the penalized likelihood formulation.

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

$$\begin{aligned}\underset{\Omega^{(i)}}{\operatorname{argmin}} & -\sum_i n_i(\ln\det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)}\widehat{\Sigma}^{(i)}\right)) \\ & + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)})\end{aligned} \tag{1.2}$$

Previous Methods: Joint Graphical Lasso (JGL) for Jointly Estimating Multiple sGGMs

- Most previous studies add a second penalty function $P()$ into the penalized likelihood formulation.
- $P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)})$ captures a certain assumption about relationships between multiple graphs.

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

$$\begin{split}\underset{\Omega^{(i)}}{\operatorname{argmin}} - \sum_i n_i (\ln\det(\Omega^{(i)}) + \mathrm{tr}\left(\Omega^{(i)}\widehat{\Sigma}^{(i)}\right)) \\ + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)})\end{split} \tag{1.2}$$

Previous Methods: Joint Graphical Lasso (JGL) for Jointly Estimating Multiple sGGMs

- Most previous studies add a second penalty function $P()$ into the penalized likelihood formulation.
- $P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)})$ captures a certain assumption about relationships between multiple graphs.
- For example, fused norm to push graphs similar: $P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)}) = \sum\limits_{i>j} ||\Omega^{(i)} - \Omega^{(j)}||_1$.

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

$$
\begin{aligned}
\underset{\Omega^{(i)}}{\operatorname{argmin}} & - \sum_i n_i (\ln\det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)}\widehat{\Sigma}^{(i)}\right)) \\
& + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)})
\end{aligned} \tag{1.2}
$$

Previous Studies: Drawbacks

- Two possible ways to infer multiple sGGMs from heterogeneous samples:
  - (1) Estimating one by one using graphical lasso by assuming the graphs are not related.
  - (2) Using JGL: joint graphical lasso by designing the appropriate second penalty function $P()$.

Previous Studies: Drawbacks

- Two possible ways to infer multiple sGGMs from heterogeneous samples:
  - (1) Estimating one by one using graphical lasso by assuming the graphs are not related.
  - (2) Using JGL: joint graphical lasso by designing the appropriate second penalty function $P()$.
- **Drawbacks:**
  - **I:** Both of them can not directly output the shared structure among multiple graphs.
  - **II:** Need extra steps to decode and can not control estimating the shared and task-specific pattern among graphs.
  - **III:** No theoretical analysis in the previous JGL studies to prove why jointly learning graphs is helpful?

Outline

Goals

Our model aims to have the following properties:

- It estimates the shared and task-specific graph patterns explicitly and simultaneously.

Goals

Our model aims to have the following properties:

- It estimates the shared and task-specific graph patterns explicitly and simultaneously.
- It can control the estimation of shared versus the task-specific patterns.

Goals

Our model aims to have the following properties:

- It estimates the shared and task-specific graph patterns explicitly and simultaneously.
- It can control the estimation of shared versus the task-specific patterns.
- It provides a strong theoretical guarantee.

Goals

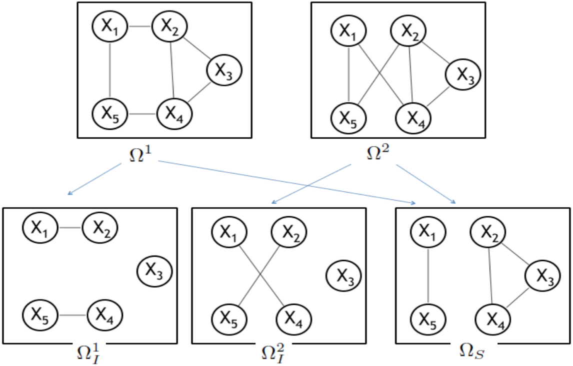
Our model aims to have the following properties:

- It estimates the shared and task-specific graph patterns explicitly and simultaneously.
- It can control the estimation of shared versus the task-specific patterns.
- It provides a strong theoretical guarantee.
- It achieves good empirical performance.

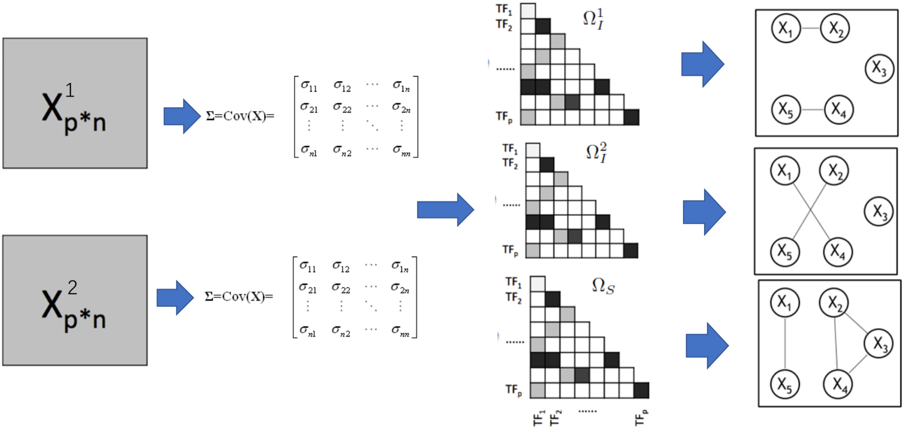
Proposed Method: Our "SIMULE" Formulation

We model each task's precision matrix $\Omega^{(i)}$ as a sum of task-specific $\Omega_I^{(i)}$ and task-shared Ω_S:

$$\Omega^{(i)} = \Omega_I^{(i)} + \Omega_S \tag{2.1}$$



Proposed method: Overview Figure



Why JGL Estimators Can't Get "SIMULE"

- JGL estimators are mostly solved by ADMM based optimization.

CLIME estimator [Cai et al.(2011)Cai, Liu, and Luo]

$$\begin{aligned} &\operatorname*{argmin}_{\Omega} ||\Omega||_1 \\ &\text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \leq \lambda_n \end{aligned} \tag{2.2}$$

Why JGL Estimators Can't Get "SIMULE"

- JGL estimators are mostly solved by ADMM based optimization.
- With "SIMULE" formulation, difficult to separate the optimization into independent ADMM sub-procedures. Because,
  - The derivative of "SIMULE" in the JGL, i.e., gradient of $\ln\det(\Omega_I^{(i)} + \Omega_S)$ gets inverse of matrix summation.
  - Inverse of the summation of two matrices makes the optimization not separable.

CLIME estimator [Cai et al.(2011)Cai, Liu, and Luo]

$$\begin{aligned} &\underset{\Omega}{\operatorname{argmin}} ||\Omega||_1 \\ &\text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \leq \lambda_n \end{aligned} \tag{2.2}$$

Why JGL Estimators Can't Get "SIMULE"

- JGL estimators are mostly solved by ADMM based optimization.
- With "SIMULE" formulation, difficult to separate the optimization into independent ADMM sub-procedures. Because,
  - The derivative of "SIMULE" in the JGL, i.e., gradient of $\ln\det(\Omega_I^{(i)} + \Omega_S)$ gets inverse of matrix summation.
  - Inverse of the summation of two matrices makes the optimization not separable.
- Therefore, we use an alternative formulation for sGGM: A constrained ℓ_1 minimization formulation.

CLIME estimator [Cai et al.(2011)Cai, Liu, and Luo]

$$\begin{aligned} &\underset{\Omega}{\operatorname{argmin}} ||\Omega||_1 \\ &\text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \le \lambda_n \end{aligned} \tag{2.2}$$

SIMULE: to Infer Shared and Individual Parts of MULtiple sGGM Explicitly

- By using a constrained ℓ_1 minimization formulation, our estimator SIMULE can jointly learn multiple graphs from multiple different but related sample datasets (on the same set of feature variables).

SIMULE

$$\widehat{\Omega}_I^{(1)}, \widehat{\Omega}_I^{(2)}, \ldots, \widehat{\Omega}_I^{(K)}, \widehat{\Omega}_S = \underset{\Omega_I^{(i)}, \Omega_S}{\operatorname{argmin}} \sum_i ||\Omega_I^{(i)}||_1 + \epsilon K ||\Omega_S||_1 \tag{2.3}$$

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

Outline

Method: Optimization Solution

- Column-wise parallelizable.

Method: Optimization Solution

- Column-wise parallelizable.
- In detail, suppose $\beta^{(i)}, \beta^s$ are a column of $\Omega_I^{(i)}, \Omega_S$.

$$\underset{\beta^{(i)},\beta^s}{\operatorname{argmin}} \sum_i ||\beta^{(i)}||_1 + \epsilon K||\beta^s||_1 \tag{2.4}$$

$$\text{Subject to: } ||\widehat{\Sigma}^{(i)}(\beta^{(i)} + \beta^s) - e_j||_\infty \leq \lambda_n, i = 1, \ldots, K$$

Method: Optimization Solution

- Column-wise parallelizable.
- In detail, suppose $\beta^{(i)}, \beta^s$ are a column of $\Omega_I^{(i)}, \Omega_S$.

$$\underset{\beta^{(i)},\beta^s}{\operatorname{argmin}} \sum_i ||\beta^{(i)}||_1 + \epsilon K ||\beta^s||_1 \tag{2.4}$$

$$\text{Subject to: } ||\widehat{\Sigma}^{(i)}(\beta^{(i)} + \beta^s) - e_j||_\infty \leq \lambda_n, i = 1, \ldots, K$$

- Can be solved by any linear programming solver.

Method: Optimization Solution

- Column-wise parallelizable.
- In detail, suppose $\beta^{(i)}, \beta^s$ are a column of $\Omega_I^{(i)}, \Omega_S$.

$$\operatorname*{argmin}_{\beta^{(i)},\beta^s} \sum_i ||\beta^{(i)}||_1 + \epsilon K||\beta^s||_1 \tag{2.4}$$

$$\text{Subject to: } ||\widehat{\Sigma}^{(i)}(\beta^{(i)} + \beta^s) - e_j||_\infty \leq \lambda_n, i = 1, \ldots, K$$

- Can be solved by any linear programming solver.
- We have proved the "SIMULE" formulation guarantees a unique optimal solution.

Method: Optimization Solution

- Column-wise parallelizable.
- In detail, suppose $\beta^{(i)}, \beta^s$ are a column of $\Omega_I^{(i)}, \Omega_S$.

$$\underset{\beta^{(i)},\beta^s}{\operatorname{argmin}} \sum_i ||\beta^{(i)}||_1 + \epsilon K||\beta^s||_1 \tag{2.4}$$

$$\text{Subject to: } ||\widehat{\Sigma}^{(i)}(\beta^{(i)} + \beta^s) - e_j||_\infty \leq \lambda_n, i = 1, \dots, K$$

- Can be solved by any linear programming solver.
- We have proved the "SIMULE" formulation guarantees a unique optimal solution.
- We use ϵ to control the sparsity of shared versus task-specific graph patterns.

Model Variation: NSIMULE for jointly estimating multiple nonparanormal Graphical Models

- The Gaussian assumption of our model can extend easily to a more general distribution family: nonparanormal.

Model Variation: NSIMULE for jointly estimating multiple nonparanormal Graphical Models

- The Gaussian assumption of our model can extend easily to a more general distribution family: nonparanormal.
- The only necessary change: by simply replacing the sample covariance matrices $\widehat{\Sigma}^{(i)}$ in Equation 2.3 into the kendal's tau correlation matrices $\widehat{\mathbf{S}}^{(i)}$.

Model Variation: NSIMULE for jointly estimating multiple nonparanormal Graphical Models

- The Gaussian assumption of our model can extend easily to a more general distribution family: nonparanormal.
- The only necessary change: by simply replacing the sample covariance matrices $\widehat{\Sigma}^{(i)}$ in Equation 2.3 into the kendal's tau correlation matrices $\widehat{\mathbf{S}}^{(i)}$.
- We denote this estimator as nonparanormal SIMULE (NSIMULE).

Outline

Theoretical Results

- Comparing SIMULE v. CLIME w.r.t the statistical convergence rate for estimating K graphs:

Multi-task:	K Single-task:
$O(\frac{\log(Kp)}{n_{tot}})$	$\sum_i O(\frac{\log p}{n_i}))$

- By assuming $n_i = \frac{n_{tot}}{K}$:

Theoretical Results

- Comparing SIMULE v. CLIME w.r.t the statistical convergence rate for estimating K graphs:

Multi-task:	K Single-task:
$O(\frac{\log(Kp)}{n_{tot}})$	$\sum_i O(\frac{\log p}{n_i}))$

- By assuming $n_i = \frac{n_{tot}}{K}$:
- We can conclude that $\frac{\log(Kp)}{n_{tot}} < K\frac{\log p}{n_{tot}}$

Theoretical Results

- Comparing SIMULE v. CLIME w.r.t the statistical convergence rate for estimating K graphs:

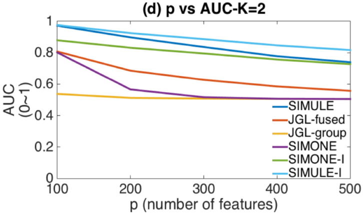
Multi-task:	K Single-task:
$O(\frac{\log(Kp)}{n_{tot}})$	$\sum_i O(\frac{\log p}{n_i}))$

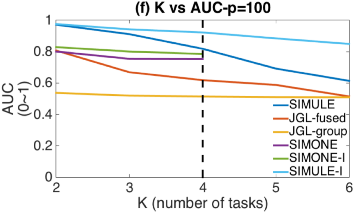
- By assuming $n_i = \frac{n_{tot}}{K}$:
- We can conclude that $\frac{\log(Kp)}{n_{tot}} < K\frac{\log p}{n_{tot}}$
- This indicates that the multi-task estimator is better!!!

Outline

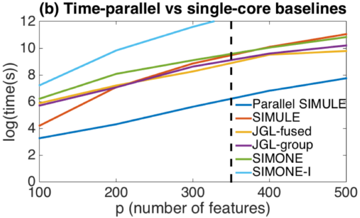
Results on Synthetic Datasets: Accuracy and Parallelization

- Accuracy (AUC with a varying p and a varying K):



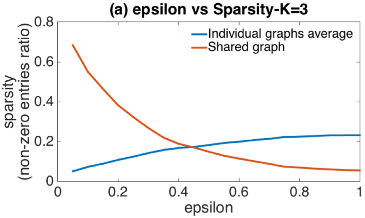


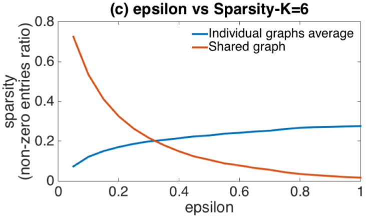
- Computation time cost with a varying p:



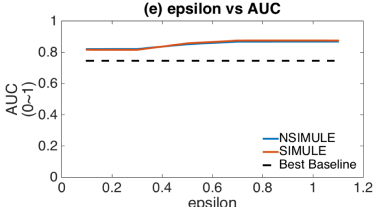
Results on Synthetic Datasets: Sensitivity of Hyperparameter ϵ

- The hyperpara ϵ controls the differences of sparsity among the shared graph and task-specific graphs.



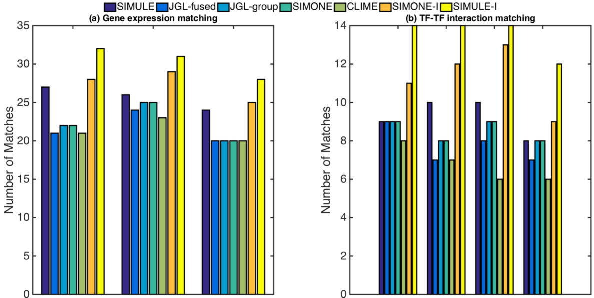


- The sensitivity of ϵ vs. accuracy.



Results on Two Real-World Datasets: Number of Matched Edges versus the Existing Domain Databases

- Two real world datasets:
  - (1) Gene expressions of samples in 2 different cell types
  - (2) Transcription Factors' ENCODE ChIP-seq measurements across 3 different cell lines
- Validation by counting the overlapped interactions according to the existing bio-databases (MInact).
- Our methods obtain the most matches compared to the state-of-the-art baselines.



R Package is Available !!!

- The project website: `http://jointggm.org/`
- R package "simule":
  - `install.packages("simule")`
  - `demo(simuleDemo)` !
  - `https://cran.r-project.org/web/packages/simule/index.html`

References

T. Cai, W. Liu, and X. Luo.
A constrained l1 minimization approach to sparse precision matrix estimation.
Journal of the American Statistical Association, 106(494):594–607, 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.

J. Friedman, T. Hastie, and R. Tibshirani.
Sparse inverse covariance estimation with the graphical lasso.
Biostatistics, 9(3):432–441, 2008.

T. Ideker and N. J. Krogan.
Differential network biology.
Molecular systems biology, 8(1):565, 2012.