Joint Learning of Multiple Related Gaussian Graphical Models from Heterogeneous Samples: Tasks, Estimators and Variations

Yanjun Qi[1]

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

@ UCLA Computational Genomics Summer Institute: CGSI 2019

Outline

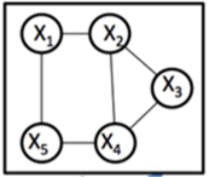
1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

This Year's Tutorial Talk: jointnets tools for Identifying Related Dependency Graphs from Heterogeneous Samples

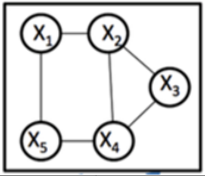
1. Graphical Models to reflect interactions among important variables



Xi	Xj
Protein	Protein
Gene	Gene
Protein	DNA/RNA
Neuron Region	Neuron Region
...	....

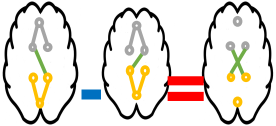
Summary: jointnets tools for Identifying Related Dependency Graphs from Heterogeneous Samples

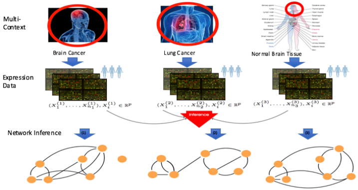
1. Graphical Models to reflect interactions among important variables



Xi	Xj
Protein	Protein
Gene	Gene
Protein	DNA/RNA
Neuron Region	Neuron Region
…	….

2. Consider Sample Heterogeneity to reflect network under many contexts

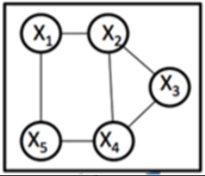




jointnets.org

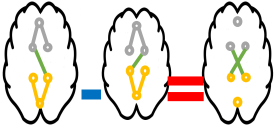
Summary: jointnets tools for Identifying Related Dependency Graphs from Heterogeneous Samples

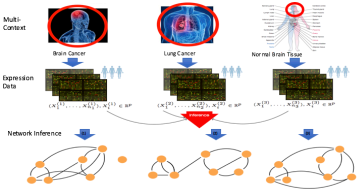
1. Graphical Models to reflect interactions among important variables



Xi	Xj
Protein	Protein
Gene	Gene
Protein	DNA/RNA
Neuron Region	Neuron Region
…	….

2. Consider Sample Heterogeneity to reflect network under many contexts



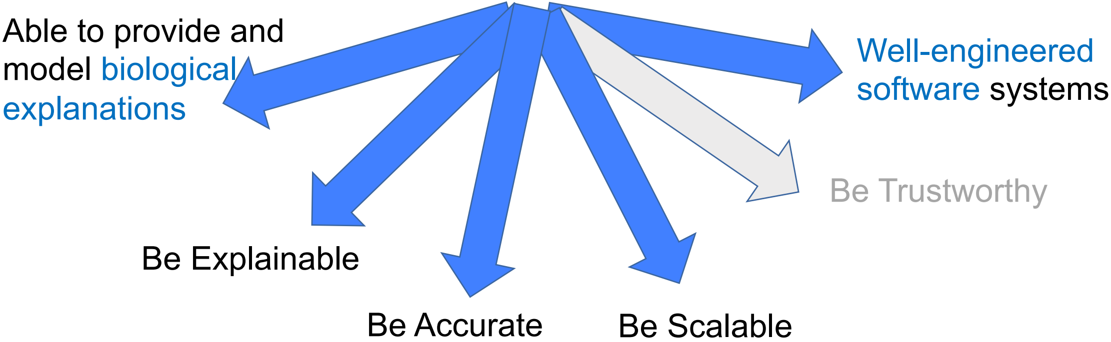


jointnets.org

- Joint graph discovery from heterogeneous samples
  - Fast and scalable graph estimators
  - Parallelizable method (GPU, multi-threading)
  - Sharp convergence rate (sharp error bounds)

Design Motivations: Our Research Philosophy in jointnets

Machine learning for Biomedicine
Our Research Philosophy:

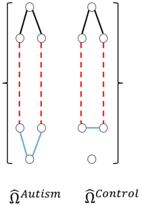


Time line of tools in jointnets.org

Timeline of JointNets

http://jointnets.org/

WSIMULE:

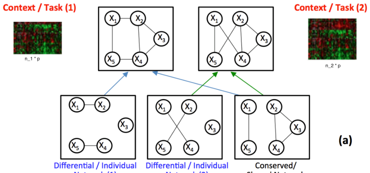


JEEK:

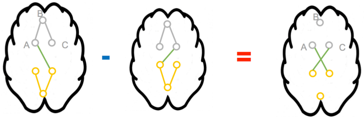
Solution of Vanilla MLE: Backward Mapping → (High-Dim.) → Proxy Backward mapping → (L1) → Elementary Estimator for sGGMs → Joint Estimation and Weighted L1 (Knowledge as Weights) → Proposed: JEEK Estimator

2015 | 2016 | 2017 | 2018 | 2019

SIMULE:



DIFFEE:



kDIFFNet:

Add Knowledge in DIFFEE

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

How to compare different estimators?

- Two major properties: [Accuracy] and [Speed]

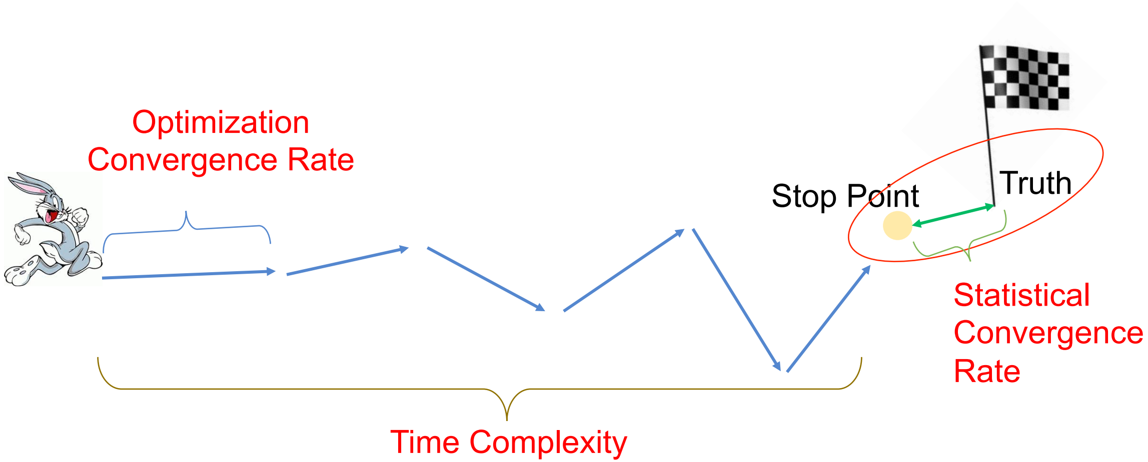
How to compare different estimators?

- Two major properties: [Accuracy] and [Speed]
- Accuracy:
  - Statistical Convergence rate / error bounds: corresponding to estimation error or approximation error / distance between your estimated parameter and the true parameter .

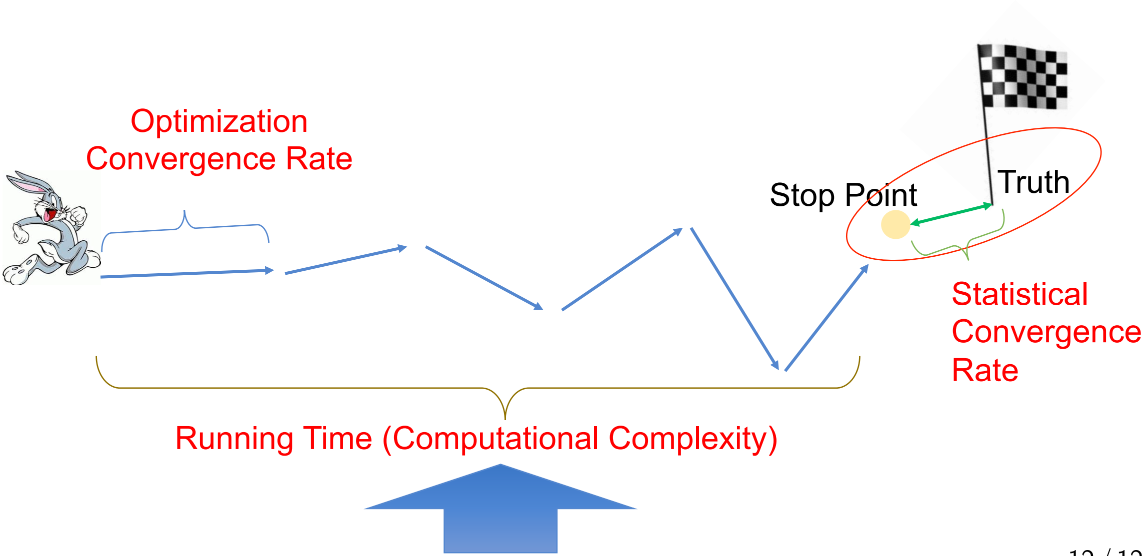
How to compare different estimators?

- Two major properties: [Accuracy] and [Speed]
- Accuracy:
  - Statistical Convergence rate / error bounds: corresponding to estimation error or approximation error / distance between your estimated parameter and the true parameter .
- Speed:
  - Computational complexity: How fast and efficient your algorithm is with respect to certain parameters, e.g., n and p.
  - Optimization convergence rate : How fast each optimization step moves the estimated parameter, such as linear or quadratic.

Overview Figure of the three major theoretical rates:



Overview Figure of the three rates: Computational Complexity



Computational Complexity: algorithmic cost

- The amount of required resources: e.g. running time, memory cost .
- Big O notation: asymptotically tight bound on the running cost.
- For machine learning tasks, mainly relate to n and p

Computational Complexity:

- Some well-known cases:
  - Matrix Multiplication: e.g., $w^T\mathbf{X}$ costs $O(np^2)$
  - Matrix inversion $O(p^3)$
  - SVD $O(p^3)$
  - soft-thresholding of matrix $O(p^2)$

Computational Complexity:

- Some well-known cases:
  - Matrix Multiplication: e.g., $w^T\mathbf{X}$ costs $O(np^2)$
  - Matrix inversion $O(p^3)$
  - SVD $O(p^3)$
  - soft-thresholding of matrix $O(p^2)$
- How to calculate if estimating parameter θ via iterative optimization?
  - Number of Iteration (depending on optimization convergence rate) $\times$ Computational complexity of each Iteration.
  - e.g., $O(Tp^3)$ if every iteration uses SVD.

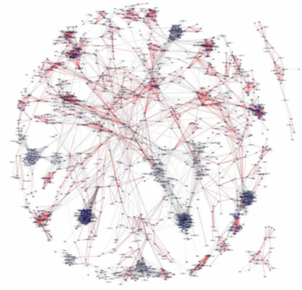
Some Notations

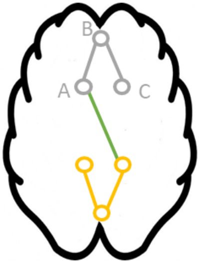
- X The sample matrix
- Σ The covariance matrix.
- Ω The precision matrix.
- p The number of features (input variables).
- n The number of samples in the data matrix.
- s The number of non-zero entries in the precision matrix.

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

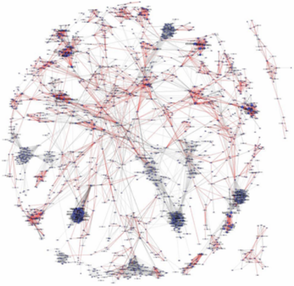
Background: Graph about p Variable

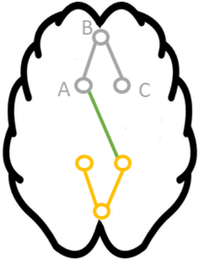




- Many applications need to know interactions among entities:
  - Brain functional connectivity
  - Gene Interactions, Transcription Factor co-bindings, ...

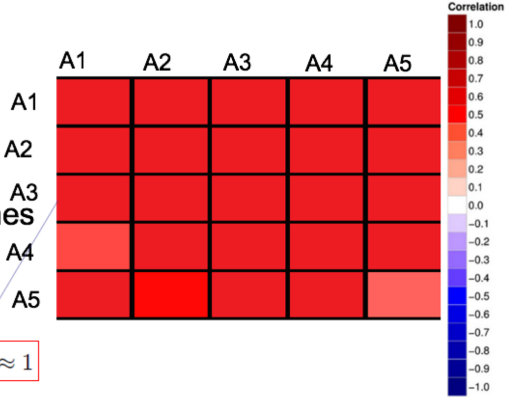
Background: Graph about p Variable





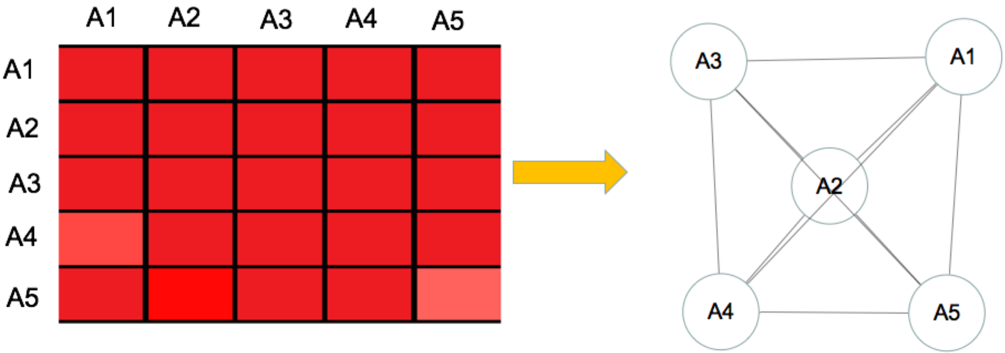
- Many applications need to know interactions among entities:
  - Brain functional connectivity
  - Gene Interactions, Transcription Factor co-bindings, ...
- Why to study the variable graphs?
  - Understanding
  - Diagnosis, e.g., marker
  - Treatment, e.g., drug development.

A1: Children swim
A2: Weather is hot
A3: High sale of ice cream
A4: Wear less amount of clothes
A5: High Electricity Consumption

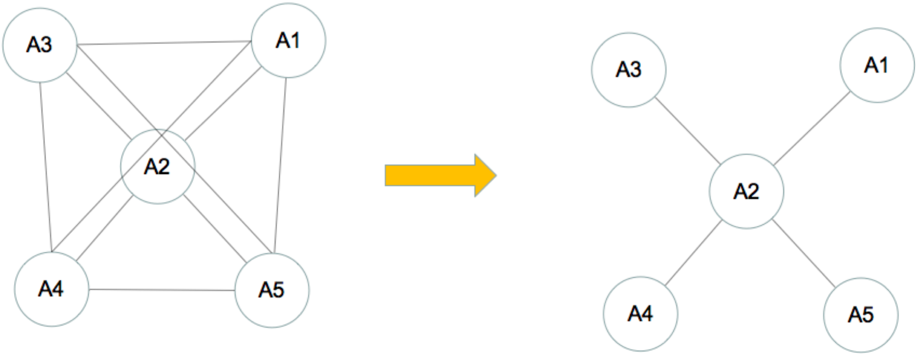


$\boxed{Cor(A_1, A_3) \approx 1}$

Background: What Type of Edges? Correlation to Conditional dependency

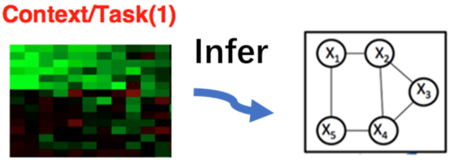


Background: What Type of Edges? Correlation to Conditional dependency

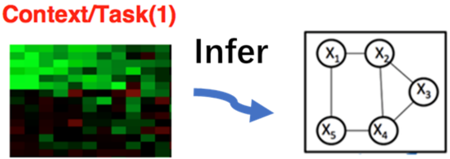


How to Infer Conditional dependency Graph? Data-driven approach

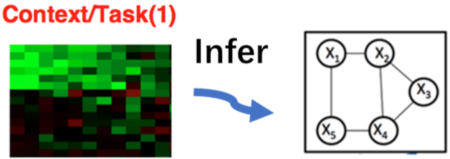
- Observed samples $\Longrightarrow$ Variable Graph



How to Infer Conditional dependency Graph? Data-driven approach



- Observed samples $\Longrightarrow$ Variable Graph
- n observed data samples
  - Each sample is a snapshot of all the entities (variables).
  - Each sample has measurements of p features/entities /variables.



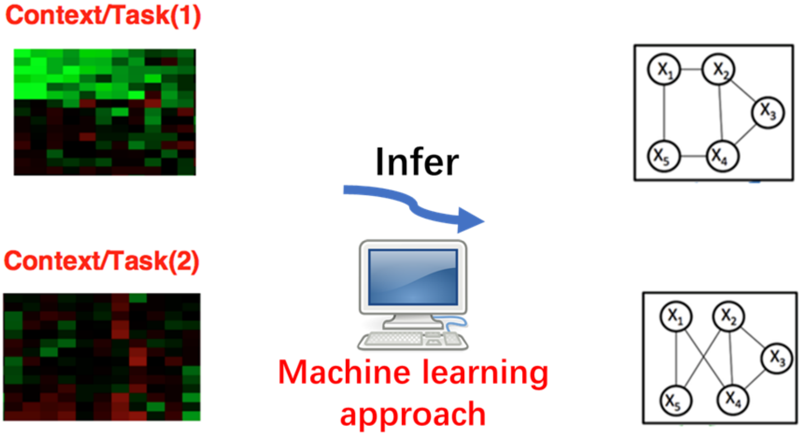
- Observed samples $\Longrightarrow$ Variable Graph
- n observed data samples
  - Each sample is a snapshot of all the entities (variables).
  - Each sample has measurements of p features/entities /variables.
- when $n >> p$ (low-dimensional, n data samples enough $\rightarrow$ a well estimated conditional dependency graph about p nodes).
- When $p > n$ (high-dimensional), need novel and theoretically sound approaches

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

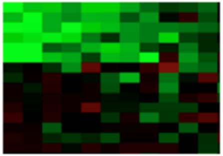
Background: Variable graphs from Heterogeneous Samples

- Most applications include heterogeneous samples.
- For example:
  - Totally n_{tot} data samples
  - From K different but related contexts, each having n_i data samples, $n_{tot} = \sum n_i$

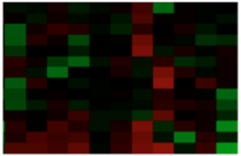


Background: Variable graphs from Heterogeneous Data

Context/Task(1)



Context/Task(2)



Case I:

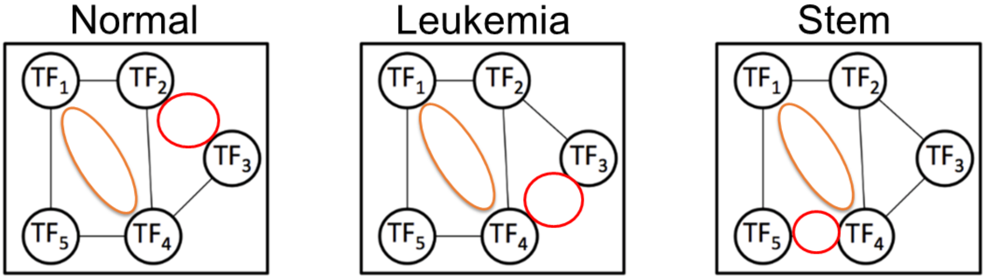
Leukemia Dataset → **Commonality** ← Breast Cancer Dataset

Case II:

Normal Cell Dataset → **Differences** ← Cancer Cell Dataset

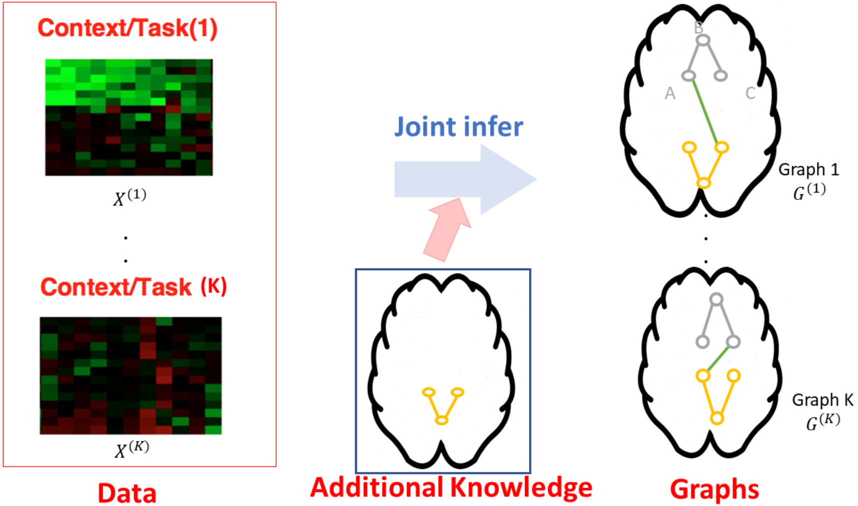
Task I: Learning multiple related graphs

- Learning multiple related graphs
- E.g., TF-TF interactions
    - Three graphs are similar

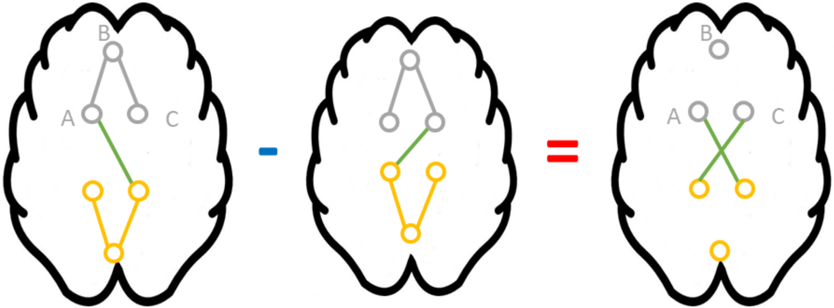


Task II: Integrating additional knowledge

- Integrating known knowledge in Learning multiple related graphs
  - E.g., known knowledge of Brain Connection E.g., known gene pathway knowledge



Task III: Learning sparse changes between two graphs



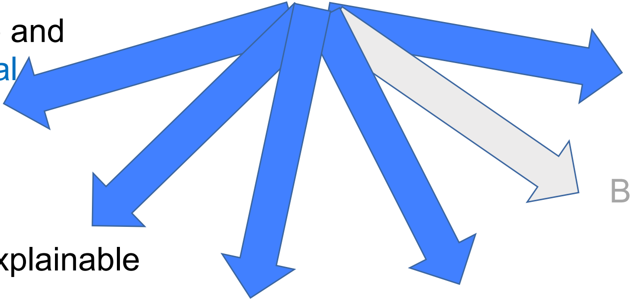
- A very interesting task:
  - Find differences in the brains of people with diseases, e.g. Autism, Alzheimer's
  - Use for understanding
  - Use for diagnosis

Notations

$X^{(i)}$	i-th Data matrix.
$\Sigma^{(i)}$	i-th Covariance matrix.
$\Omega^{(i)}$	i-th Inverse of covariance matrix (precision matrix).
p	The total number of feature variables.
n_{tot}	The total number of samples.
X^{tot}	the concatenation of all Data matrices.
Σ^{tot}	the concatenation of all Covariance matrices.
Ω^{tot}	the concatenation of all Inverse of covariance matrices (precision matrices).
W_I^{tot}	$(W_I^{(1)}, W_I^{(2)}, \ldots, W_I^{(K)})$
W_S^{tot}	$(W_S, W_S, \ldots, W_S)$
K	The total number of contexts.

Design Motivations: Our Research Philosophy in jointnets

Machine learning for Biomedicine
Our Research Philosophy:



Able to provide and model biological explanations

Well-engineered software systems

Be Trustworthy

Be Explainable

Be Accurate

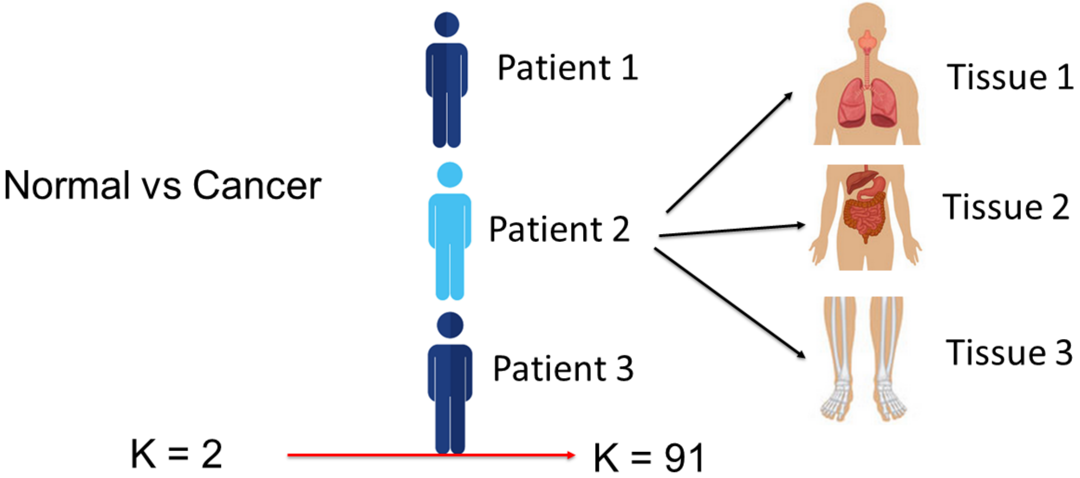
Be Scalable

Computational Challenges: More Num of features (p) to consider

- Yeast gene: 6K
  $\downarrow$
  Human gene: 30K

- Words interaction, millions of words ($p > 1,000,000$)

Computational Challenges: More num of tasks (K) to consider



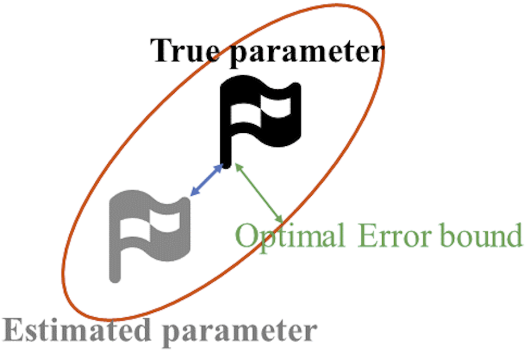
ENCODE Project Consortium et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57–74, 2012.

Why do we care computational complexity?

Estimators	JGL	WSIMULE
Computational complexity	$O(Kp^3)$ / iter	$O(K^4 p^5)$
Bottle neck	SVD	Linear programming

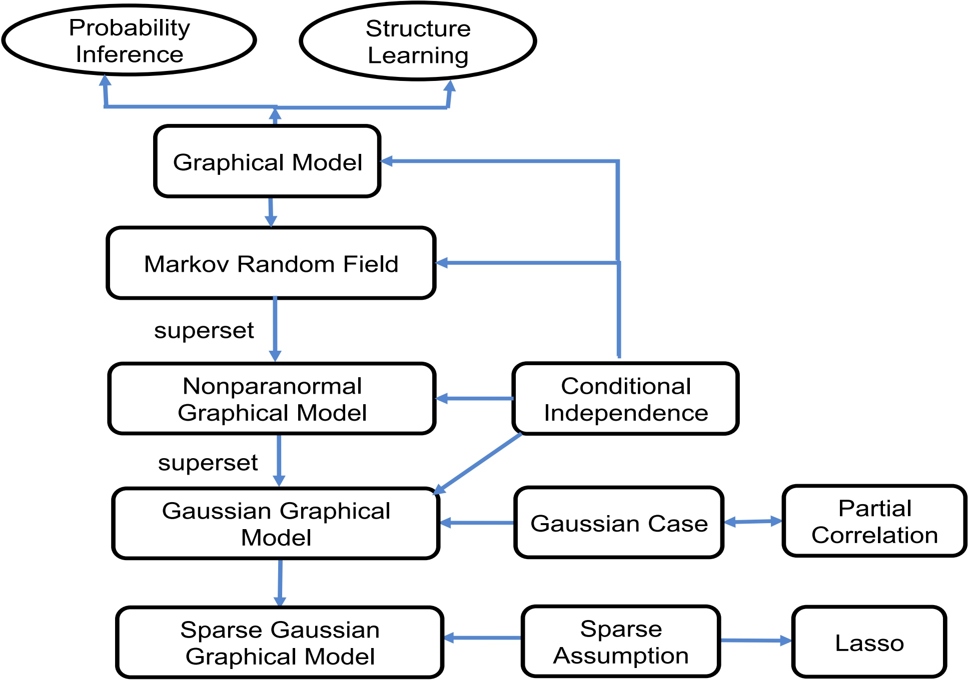
When $K = 91,\ p = 30\text{K}$	JGL	WSIMULE
Time	3.5 days / iter	years

- For large-scale cases, we need to design $O(p^2)$ methods, and consider parallelization computer architectures!!!
- At the same time, no sacrifices of the accuracy, e.g., same level of $||\widehat{\theta} - \theta^*||$;



Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

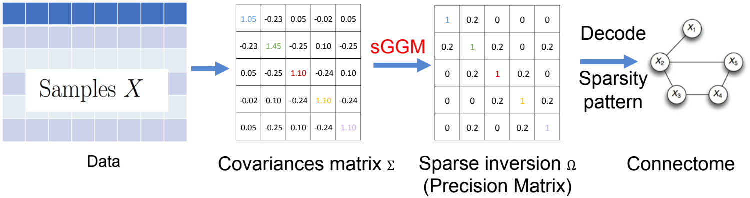

Probability Inference
Structure Learning
Graphical Model
Markov Random Field
superset
Nonparanormal Graphical Model
Conditional Independence
superset
Gaussian Graphical Model
Gaussian Case
Partial Correlation
Sparse Gaussian Graphical Model
Sparse Assumption
Lasso

Basics: Gaussian Case

- In the Gaussian case, the conditional dependence and partial correlation structure are equivalent.
- This pairwise relationship can be naturally described via a graph $G = (V, E)$.
- Undirected Gaussian Graphical Model, Undirected nonparanormal Graphical model, Markov random field;

Two main tasks for Graphical Models:

- **Probability Inference:** estimate joint probability, marginal probability, and conditional probability.
- **Structure learning:** Give dataset $\mathbf{X}$, learn the Graph structure from $\mathbf{X}$ (i.e., learn the edge patterns between variables).

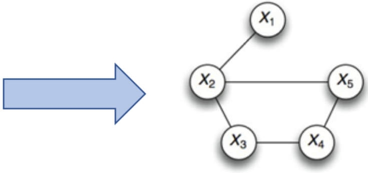


Background: Sparse Gaussian Graphical Model (sGGM)

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

Inverse Covariance Matrix

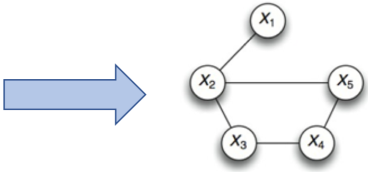
$$\begin{pmatrix} 1 & 0.2 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 & 0.2 \\ 0 & 0.2 & 1 & 0.2 & 0 \\ 0 & 0 & 0.2 & 1 & 0.2 \\ 0 & 0.2 & 0 & 0.2 & 1 \end{pmatrix}$$



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

Inverse Covariance Matrix

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

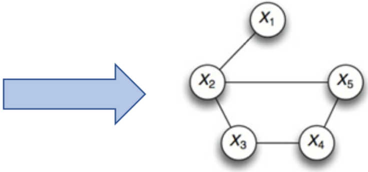


Background: Sparse Gaussian Graphical Model (sGGM)

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

Inverse Covariance Matrix

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

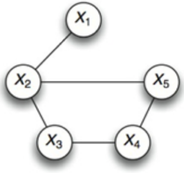


- $X \sim N(\mu, \Sigma)$.
- Covariance matrix Σ can be calculated from X
- Precision matrix Ω is the inverse of covariance matrix Σ

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

Inverse Covariance Matrix

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

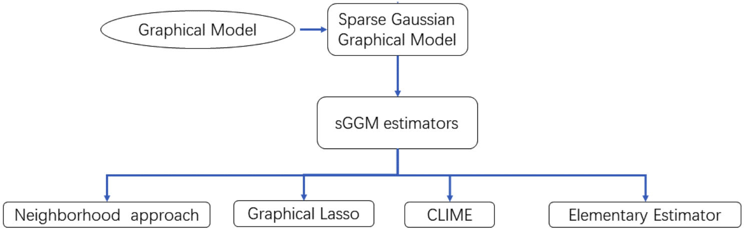


- Traditionally, we estimate sGGM from samples (of a single task) using an ℓ_1 penalized MLE formulation.

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

$$\operatorname*{argmin}_{\Omega} - \ln \det(\Omega) + \operatorname{tr}\left(\Omega\widehat{\Sigma}\right) + \lambda_n ||\Omega||_1 \tag{2.1}$$

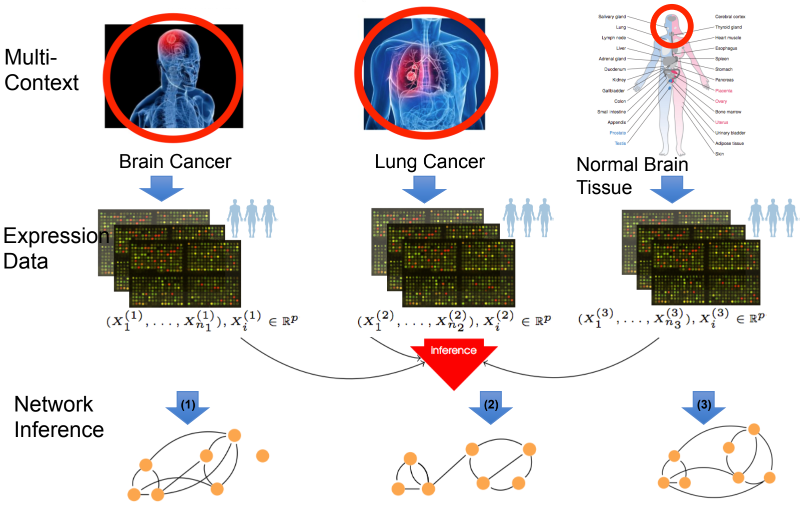
Four kinds of Estimators for Estimating sGGM from Data



Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Task I: Joint structure learning of Related Graph Structures from Multiple Related Datasets



JGL: 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]

$$\begin{aligned}\underset{\Omega^{(i)}}{\operatorname{argmin}} & -\sum_i (\ln\det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)}\widehat{\Sigma}^{(i)}\right)) \\ & + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)})\end{aligned} \tag{2.2}$$

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

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

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

$$\begin{aligned} \underset{\Omega^{(i)}}{\operatorname{argmin}} & -\sum_i (\ln\det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)}\widehat{\Sigma}^{(i)}\right)) \\ & + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) \end{aligned} \tag{2.2}$$

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

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

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

$$\begin{aligned} \underset{\Omega^{(i)}}{\operatorname{argmin}} & - \sum_i (\ln \det(\Omega^{(i)}) + \operatorname{tr}\left(\Omega^{(i)} \widehat{\Sigma}^{(i)}\right)) \\ & + \lambda_1 \sum_i ||\Omega^{(i)}||_1 + \lambda_2 P(\Omega^{(1)}, \Omega^{(2)}, \ldots, \Omega^{(K)}) \end{aligned} \tag{2.2}$$

Multi-task sGGMs estimators through JGL framework:

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

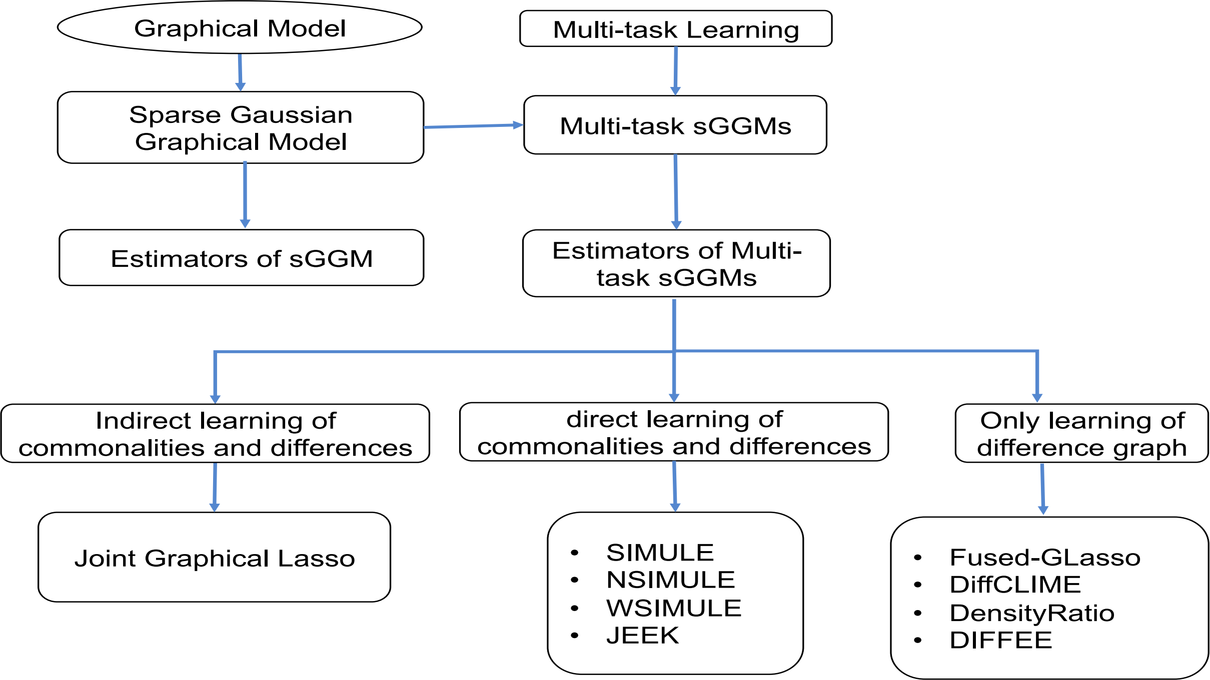
$P(\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)}) = ||\Omega^{(1)}, \Omega^{(2)}, \dots, \Omega^{(K)}||_{\mathcal{G},2}$.

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

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

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

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



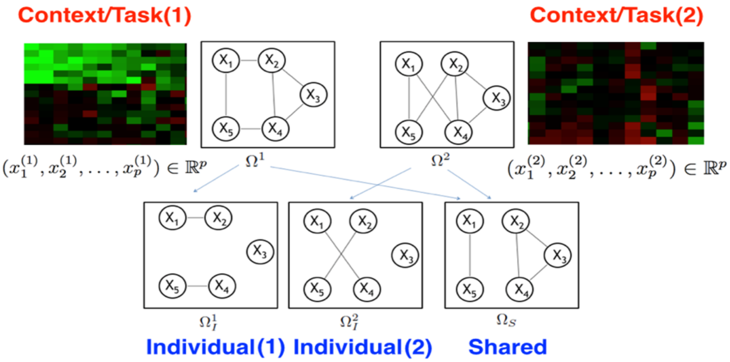
Graphical Model
Multi-task Learning
Sparse Gaussian Graphical Model
Multi-task sGGMs
Estimators of sGGM
Estimators of Multi-task sGGMs
Indirect learning of commonalities and differences
direct learning of commonalities and differences
Only learning of difference graph
Joint Graphical Lasso
SIMULE
NSIMULE
WSIMULE
JEEK
Fused-GLasso
DiffCLIME
DensityRatio
DIFFEE

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

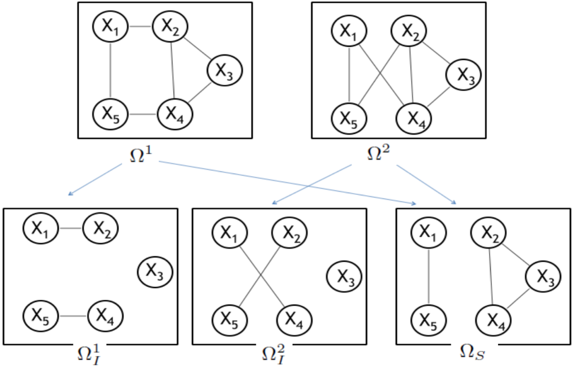
Explicit Estimation?

- 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.
- Get to know both: House keeping interactions and Context-specific networks

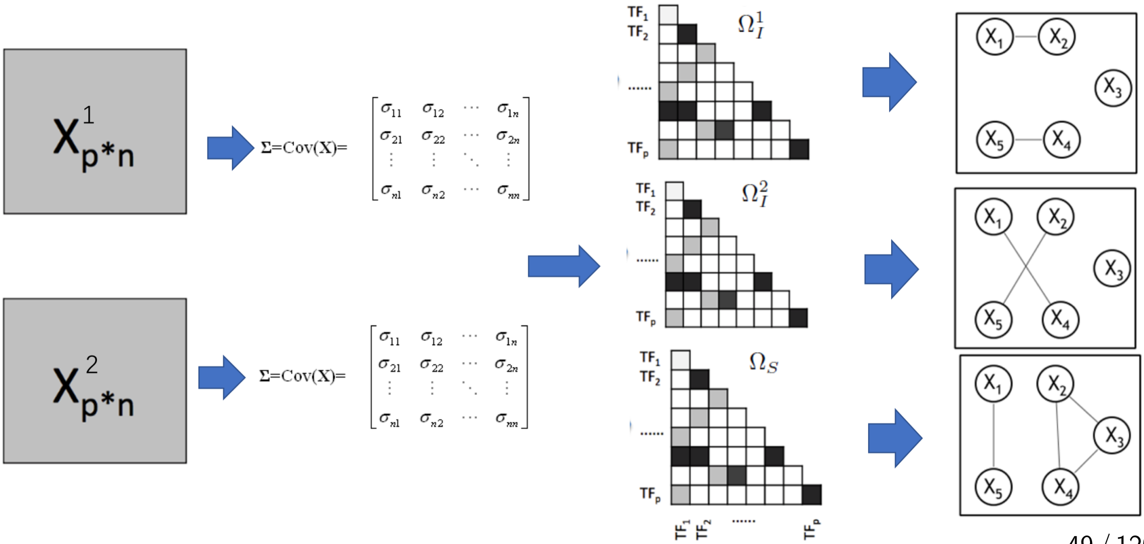


We model each task's precision matrix $\Omega^{(i)}$ as a sum of task-specific $\Omega_I^{(i)}$ and task-shared Ω_S:

$$\Omega^{(i)} = \Omega_I^{(i)} + \Omega_S \tag{2.3}$$



SIMULE method: Overview Figure



Goals

SIMULE 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.

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]

$$\begin{aligned} &\operatorname*{argmin}_{\Omega} ||\Omega||_1 \\ &\text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \leq \lambda_n \end{aligned} \tag{2.4}$$

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]

$$\operatorname*{argmin}_{\Omega} ||\Omega||_1$$
$$\text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \le \lambda_n \tag{2.4}$$

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]

$$\begin{gathered}\underset{\Omega}{\operatorname{argmin}} ||\Omega||_1 \\ \text{Subject to: } ||\widehat{\Sigma}\Omega - I||_\infty \leq \lambda_n\end{gathered} \tag{2.4}$$

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

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

SIMULE

$$\widehat{\Omega}_I^{(1)}, \widehat{\Omega}_I^{(2)}, \ldots, \widehat{\Omega}_I^{(K)}, \widehat{\Omega}_S = \operatorname*{argmin}_{\Omega_I^{(i)}, \Omega_S} \sum_i ||\Omega_I^{(i)}||_1 + \epsilon K ||\Omega_S||_1 \tag{2.5}$$

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

Theoretical Results: Statistical Convergence Rate

- 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: Statistical Convergence Rate

- 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: Statistical Convergence Rate

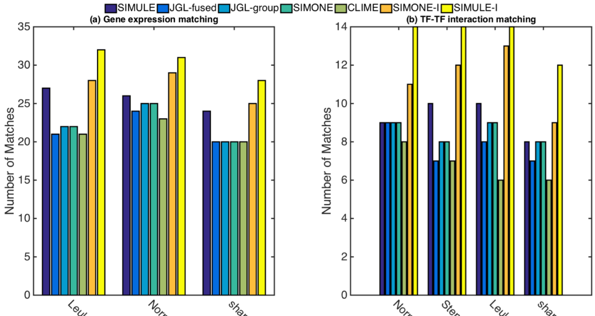
- Comparing SIMULE v. CLIME w.r.t the statistical convergence rate for estimating K graphs:

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!!!

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). figure
- Our methods obtain the most matches compared to the state-of-the-art baselines.



Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

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 2.5 into the kendal's tau correlation matrices $\widehat{\mathbf{S}}^{(i)}$.

Model Variation: NSIMULE for jointly estimating multiple nonparanormal Graphical Models

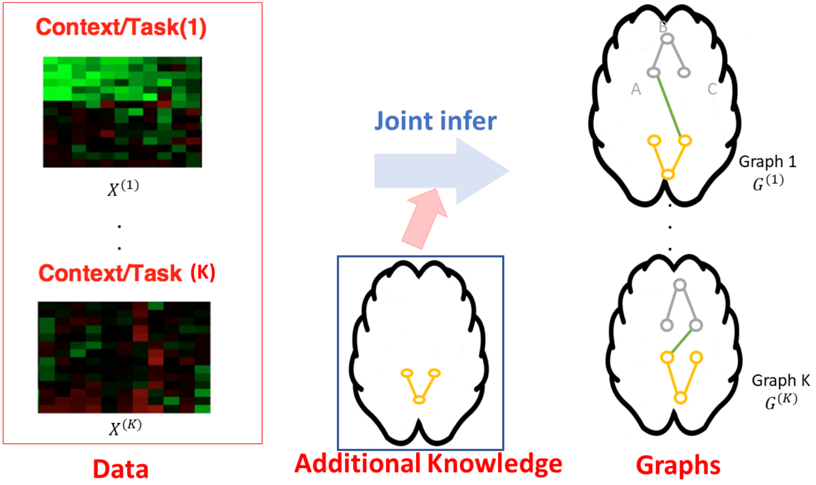
- The Gaussian assumption of our model can extend easily to a more general distribution family: nonparanormal.
- The only necessary change: by simply replacing the sample covariance matrices $\widehat{\Sigma}^{(i)}$ in Equation 2.5 into the kendal's tau correlation matrices $\widehat{\mathbf{S}}^{(i)}$.
- We denote this estimator as nonparanormal SIMULE (NSIMULE).

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Task II: Integrating additional knowledge

- Many additional knowledge exist beyond samples when Joint structure learning;
- E.g., known prior knowledge about Brain Connection



Solution: Using Knowledge as Weight in Regularization (KW-norm)

- Integrating additional knowledge through a novel regularization function $\mathcal{R}(\cdot)$

KW-norm

$$\mathcal{R}(\{\Omega^{(i)}\}) = \sum_{i=1}^{K} ||W_I^{(i)} \circ \Omega_I^{(i)}||_1 + \sum_{i=1}^{K} ||W_S \circ \Omega_S||_1 \tag{2.6}$$

- $\Omega^{(i)} = \Omega_I^{(i)} + \Omega_S$
- $\{W_I^{(i)}\}$: weights describing knowledge of each individual graph.
- W_S: weights describing knowledge of the shared graph.

Solution: Using Knowledge as Weight in Regularization (KW-norm)

- Use *tot* notation

KW-norm

$$\mathcal{R}(\Omega^{tot}) = ||W_I^{tot} \circ \Omega_I^{tot}||_1 + ||W_S^{tot} \circ \Omega_S^{tot}||_1 \quad (2.7)$$

- W_I^{tot}: weights describing knowledge of each individual graph.
- W_S^{tot}: weights describing knowledge of the shared graph.

Solution: Using Knowledge as Weight in Regularization (KW-norm)

- Use *tot* notation

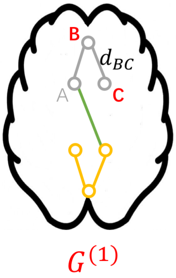
KW-norm

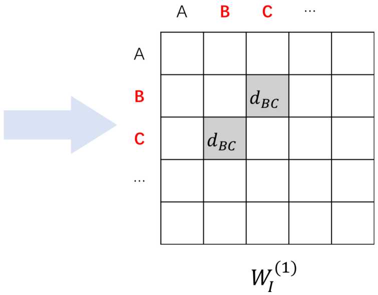
$$\mathcal{R}(\Omega^{tot}) = ||W_I^{tot} \circ \Omega_I^{tot}||_1 + ||W_S^{tot} \circ \Omega_S^{tot}||_1 \tag{2.7}$$

- W_I^{tot}: weights describing knowledge of each individual graph.
- W_S^{tot}: weights describing knowledge of the shared graph.
- No need to design knowledge-specific optimization
- KW-norm is flexible.

Example I: KW-norm representing the edge-level knowledge

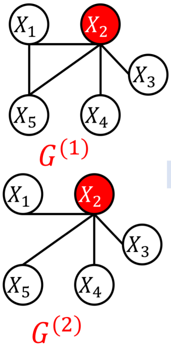
- e.g., Spatial distance among brain regions;





Example II: KW-norm describing the node-level knowledge

- e.g., X_2 is a known hub node;



	1	2	3	4	5
1		$1/\gamma$	1	1	1
2	$1/\gamma$		$1/\gamma$	$1/\gamma$	$1/\gamma$
3	1	$1/\gamma$		1	1
4	1	$1/\gamma$	1		1
5	1	$1/\gamma$	1	1	

W_s

WSIMULE: A weighted SIMULE estimator

SIMULE

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

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

- ADD $W_I^{(i)}, W_S$ $\Downarrow$

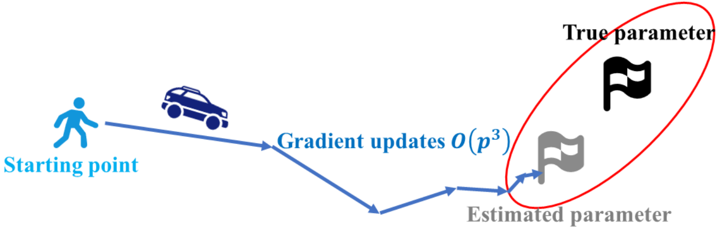
W-SIMULE

$$\widehat{\Omega}_I^{(1)}, ..., \widehat{\Omega}_I^{(K)}, \widehat{\Omega}_S = \sum_i \operatorname*{argmin}_{\Omega_I^{(i)}, \Omega_S} ||W_I^{(i)} \circ \Omega_I^{(i)}||_1 + K ||W_S \circ \Omega_S||_1 \tag{2.8}$$

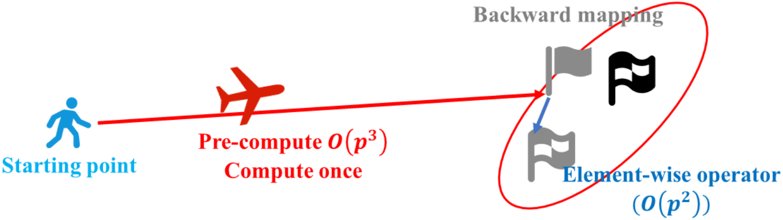
$$\text{Subject to: } ||\Sigma^{(i)}(\Omega_I^{(i)} + \Omega_S) - I||_\infty \le \lambda, i \in 1, ..., K$$

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:



- Previous studies:
- Elementary Estimator:



JEEK: Combine EE and KW-norm

Elementary Estimator

$$\underset{\theta}{\operatorname{argmin}} \mathcal{R}(\theta)$$
$$\text{Subject to: } \mathcal{R}^*(\theta - \mathcal{B}^*(\widehat{\phi})) \leq \lambda_n \tag{2.9}$$

$$+$$

KW-norm

$$\mathcal{R}(\Omega^{tot}) = ||W_I^{tot} \circ \Omega_I^{tot}||_1 + ||W_S^{tot} \circ \Omega_S^{tot}||_1 \tag{2.10}$$

JEEK Method: Joint Elementary Estimator incorporating additional Knowledge (JEEK)

EE	$\mathcal{R}(\cdot)$	θ	$\widehat{\theta}_n$	$\mathcal{R}^*(\cdot)$
EE-sGGM	$\|\|\cdot\|\|_1$	Ω	$[T_v(\widehat{\Sigma})]^{-1}$	$\|\|\cdot\|\|_\infty$
JEEK	kw-norm	Ω^{tot}	$inv[T_v(\widehat{\Sigma}^{tot})]$	kw-dual

JEEK

$$\underset{\Omega_I^{tot},\Omega_S^{tot}}{\operatorname{argmin}} ||W_I^{tot} \circ \Omega_I^{tot}||_1 + ||W_S^{tot} \circ \Omega_S^{tot}||$$

?? Subject to: $$||\frac{1}{W_I^{tot}} \circ (\Omega^{tot} - inv(T_v(\widehat{\Sigma}^{tot})))||_\infty \leq \lambda_n \qquad (2.11)$$

$$||\frac{1}{W_S^{tot}} \circ (\Omega^{tot} - inv(T_v(\widehat{\Sigma}^{tot})))||_\infty \leq \lambda_n$$

JEEK – Solution

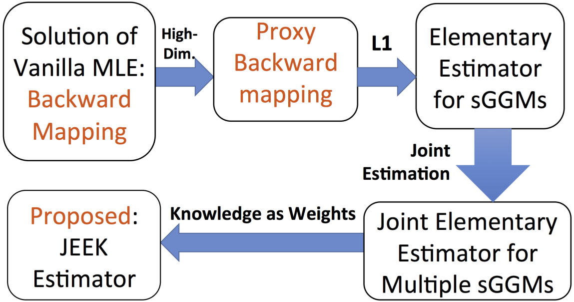
- Fast and Scalable solution[1] – p^2 small linear programming subproblems with only $K+1$ variables:

$$
\begin{aligned}
&\operatorname*{argmin}_{a_i, b} \sum_i |w_i a_i| + K|w_s b| \\
&\text{Subject to: } |a_i + b - c_i| \le \frac{\lambda_n}{\min(w_i, w_s)}, \\
&i = 1, \dots, K
\end{aligned}
\tag{2.12}
$$

[1] $a_i := \Omega_I^{(i)}{}_{j,k}$ (the $\{j,k\}$-th entry of $\Omega^{(i)}$)
$b := \Omega_{S_{j,k}}$
$c_i = [T_v(\widehat{\Sigma}^{(i)})]^{-1}_{j,k}$.
$W^{(i)}_{j,k} = w_i$ and $W^S_{j,k} = w_s$.

Why JEEK is better

- Rich and flexible for integrating additional knowledge
  - e.g., spatial, anatomy, hub, pathway, location, known edges;
- Parallelizable optimization with small sub-problems.
- Theoretical guaranteed



Theoretical Results: Sharp convergence rate

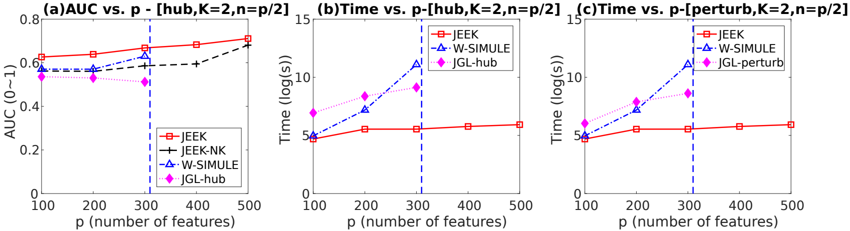
- Sharp convergence rate as the state-of-art

$$
\begin{aligned}
&||\widehat{\Omega}^{tot} - \Omega^{tot*}||_F \leq 4\sqrt{k_i + k_s}\lambda_n \\
&\max(||W_I^{tot} \circ (\widehat{\Omega}^{tot} - \Omega^{tot*})||_\infty, ||W_S^{tot} \circ (\widehat{\Omega}^{tot} - \Omega^{tot*}||_\infty) \leq 2\lambda_n \\
&||W_I^{tot} \circ (\widehat{\Omega}_I^{tot} - \Omega_I^{tot*})||_1 + ||W_S^{tot} \circ (\widehat{\Omega}_S^{tot} - \Omega_S^{tot*})||_1 \leq 8(k_i + k_s)\lambda_n
\end{aligned} \tag{2.13}
$$

Where a, c, κ_1 and κ_2 are constants

$$
\begin{aligned}
&||\widehat{\Omega}^{tot} - \Omega^{tot*}||_F \\
&\leq \frac{16\kappa_1 a \max_{j,k}(W_I^{tot}{}_{j,k}, W_S^{tot}{}_{j,k})}{\kappa_2} \sqrt{\frac{(k_i + k_s)\log(Kp)}{n_{tot}}}
\end{aligned} \tag{2.14}
$$

Empirical Results on Multiple Synthetic Datasets



- JEEK outperforms the speed of the state-of arts significantly ($\sim 5000\times$ faster);
- JEEK obtains better AUC as the state-of-the-art;
- JEEK obtains better AUC than JEEK-NK (no additional knowledge).

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Task III: To Learn Differential Network from two Datasets

- Focus: How to directly estimate / learn Differential Network (Δ) from Two datasets ($\mathbf{X}_c$, $\mathbf{X}_d$) about the same set of features in a large scale.

Sparsity Assumption:

Estimating the Difference by separately Learning Two Graphs from two datasets has Limitations

- If estimating two graphs separately, we need to enforce sparsity assumption on both graphs
- However, in some real-world applications, G_c, G_d are not sparse.

Direct modeling the differential networks I: Fused JointGLasso

Fused GLasso

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

$$\underset{\Omega_c, \Omega_d \succ 0, \Delta}{\operatorname{argmin}} \mathcal{L}(\Omega_c) + \mathcal{L}(\Omega_d) \lambda_n (||\Omega_c||_1 + ||\Omega_d||_1) + \lambda_2 ||\Delta||_1 \tag{2.15}$$

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

$$\begin{gathered}\underset{\Delta}{\operatorname{argmin}} ||\Delta||_1 \\ \text{Subject to: } ||\widehat{\Sigma}_c \Delta \widehat{\Sigma}_d - (\widehat{\Sigma}_c - \widehat{\Sigma}_d)||_\infty \leq \lambda_n\end{gathered} \tag{2.16}$$

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

$$\frac{p_c(x,\theta_c)}{p_d(x,\theta_d)} \propto \exp\left(\sum_t \Delta_t f_t(x)\right) \tag{2.17}$$

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

Density Ratio

$$r(x;\theta) = \frac{1}{N(\theta)} \exp\left(\sum_t \Delta_t f_t(x)\right) \tag{2.18}$$

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

$$\begin{aligned} \widehat{p}(x) &= p_d(x) r(x;\theta) \\ \mathrm{KL}[p_c || \widehat{p}] &= \text{Const.} - \int p_c(x) \log r(x;\theta) dx. \end{aligned} \tag{2.19}$$

DIFFEE: Large Scale Differential sGGM

- Two cases : d (disease) & c (control)

$$\underset{\theta}{\operatorname{argmin}} ||\theta||_1 \quad \text{Subject to: } ||\theta - \mathcal{B}^*(\widehat{\phi})||_\infty \le \lambda_n \tag{2.20}$$

$$\xrightarrow[]{\Delta = \Omega_d - \Omega_c} \Longrightarrow$$

$$\underset{\Delta}{\operatorname{argmin}} ||\Delta||_1 \quad \text{Subject to: } ||\Delta - \mathcal{B}^*(\widehat{\Sigma}_d, \widehat{\Sigma}_c)||_\infty \le \lambda_n \tag{2.21}$$

DIFFEE: Large Scale Differential sGGM via EE

Elementary Estimator (EE)

$$\underset{\theta}{\operatorname{argmin}} \mathcal{R}(\theta)$$
$$\text{Subject to: } \mathcal{R}^*(\theta - \mathcal{B}^*(\widehat{\phi})) \leq \lambda_n \tag{2.22}$$

EE	$\mathcal{R}(\cdot)$	θ	$\widehat{\theta}_n$	$\mathcal{R}^*(\cdot)$
EE-sGGM	$\|\|\cdot\|\|_1$	Ω	$[T_v(\widehat{\Sigma})]^{-1}$	$\|\|\cdot\|\|_\infty$
DIFFEE	$\|\|\cdot\|\|_1$	Δ	$\left([T_v(\widehat{\Sigma}_d)]^{-1} - [T_v(\widehat{\Sigma}_c)]^{-1}\right)$	$\|\|\cdot\|\|_\infty$

DIFFEE

$$\underset{\Delta}{\operatorname{argmin}} ||\Delta||_1$$
$$\text{Subject to: } ||\Delta - \left([T_v(\widehat{\Sigma}_d)]^{-1} - [T_v(\widehat{\Sigma}_c)]^{-1}\right)||_\infty \leq \lambda_n \tag{2.23}$$

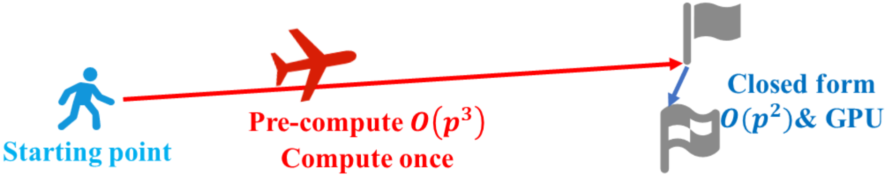
DIFFEE: Optimization Solution

- Close form

$$\widehat{\Delta} = S_{\lambda_n}([T_v(\widehat{\Sigma}_d)]^{-1} - [T_v(\widehat{\Sigma}_c)]^{-1}) \tag{2.24}$$

$$[S_\lambda(A)]_{ij} = \mathrm{sign}(A_{ij}) \max(|A_{ij}| - \lambda, 0) \tag{2.25}$$

- GPU-parallelizable



Computational Complexity of DIFFEE:

- It has closed-form solution.
- It is faster than the previous studies:

DIFFEE	FusedGLasso	Density Ratio	Diff-CLIME
$O(p^3)$	$O(T * p^3)$	$O((n_c + p^2)^3)$	$O(p^8)$

- $O(p^2)$ to tune different λ_n
- Theoretical guaranteed

Theoretical Results: Statistical Convergence Rate

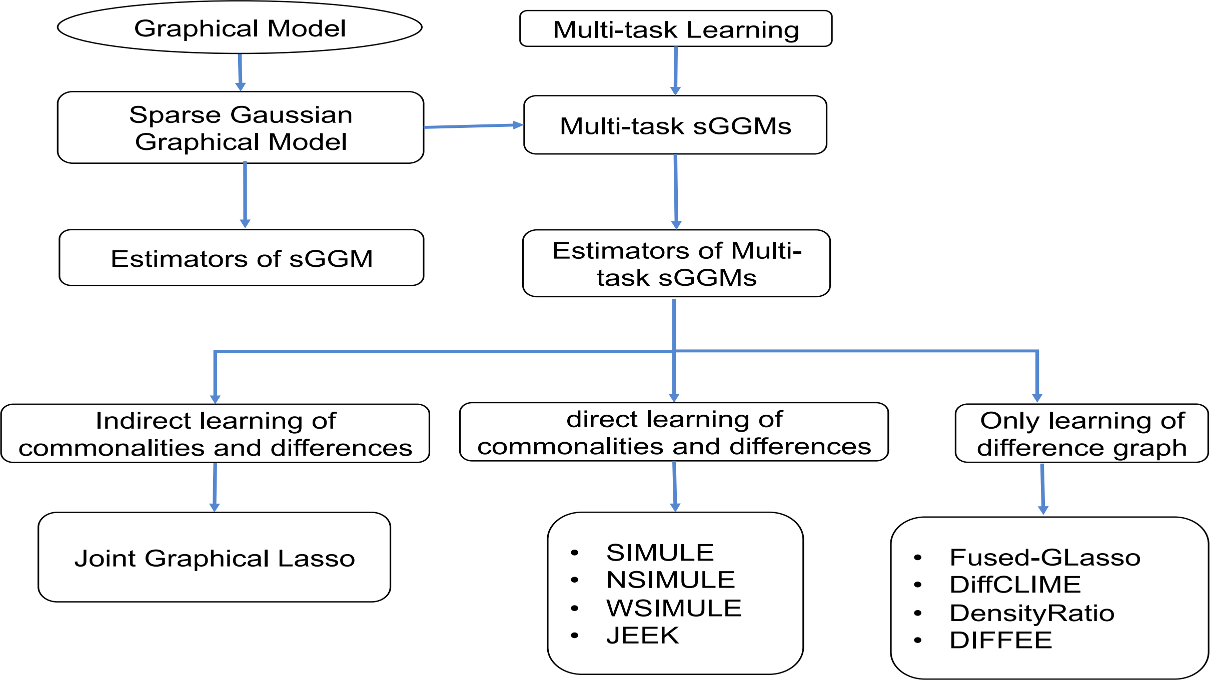
- error bound: $||\Delta^* - \widehat{\Delta}||$
- DIFFEE achieves similar error bound as the previous studies.

DIFFEE	FusedGLasso	Density Ratio	Diff-CLIME
$\frac{\log p}{\min(n_c, n_d)}$	N/A	$\frac{\log p}{\min(n_c, n_d)}$	$\frac{\log p}{\min(n_c, n_d)}$

Empirical Results on fMRI Datasets: the Classification Accuracy

- (1) ABIDE dataset
- (2) Train the differential network and use it as the parameter of a LDA classifier

Method	DIFFEE	FusedGLasso	Diff-CLIME
Accuracy (%)	**57.58%**	56.90%	53.79%

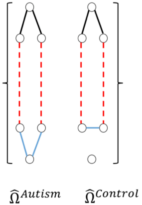

Graphical Model
Multi-task Learning
Sparse Gaussian Graphical Model
Multi-task sGGMs
Estimators of sGGM
Estimators of Multi-task sGGMs
Indirect learning of commonalities and differences
direct learning of commonalities and differences
Only learning of difference graph
Joint Graphical Lasso
• SIMULE
• NSIMULE
• WSIMULE
• JEEK
• Fused-GLasso
• DiffCLIME
• DensityRatio
• DIFFEE

Recap: Time line of tools jointnets.org

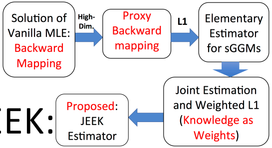
Timeline of JointNets

http://jointnets.org/

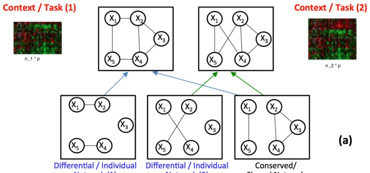
WSIMULE:



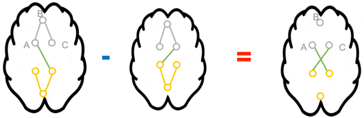
JEEK:



SIMULE:



DIFFEE:



kDIFFNet:

Add Knowledge in DIFFEE

Related Publications:

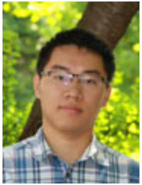
- JEEK
  - A Fast and Scalable Joint Estimator for Integrating Additional Knowledge in Learning Multiple Related Sparse Gaussian Graphical Models, B Wang, A Sekhon, Y Qi, ICML 2018
- DIFFEE
  - Fast and Scalable Learning of Sparse Changes in High-Dimensional Gaussian Graphical Model Structure, B Wang, A Sekhon, Y Qi, AISTATS 2018
- SIMULE, NSIMULE and W-SIMULE
  - A constrained L1 minimization approach for estimating multiple sparse Gaussian or nonparanormal graphical models, B Wang, R Singh, Y Qi, Machine Learning 106 (9-10), 1381-1417, 2016
  - A Constrained, Weighted-L1 Minimization Approach for Joint Discovery of Heterogeneous Neural Connectivity Graphs, C Singh, B Wang, Y Qi, Advances in Modeling and Learning Interactions from Complex Data, NeurIPS 2017 Workshop

R Package is Available !!!

- The project website: `http://jointnets.org/`
- R package "simule":
  - `install.packages("simule")`
  - `demo(simule)` !
- R package "diffee":
  - `install.packages("diffee")`
  - `demo(diffee)` !
- R package "jeek":
  - `install.packages("jeek")`
  - `demo(jeek)` !
- A complete package "jointNet" in CRAN.
  - `install.packages('JointNets', dependencies=TRUE)`
  - Including all above tools and more variations, plus network visualization, synthetic data simulation, graph evaluation and downstream classification;

Acknowledgments

Ritambhara Singh



Beilun Wang

Weilin Xu

Jack Lanchantin

Arshdeep Sekhon

Ji Gao

UVA Department of Biochemistry and Molecular Genetics: Dr. Mazhar Adli

National Science Foundation
WHERE DISCOVERIES BEGIN

UVA Computer Science Dept. Security Research Group: Prof. David Evans

Computer Science at the University of Virginia

Thank you

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.

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.

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

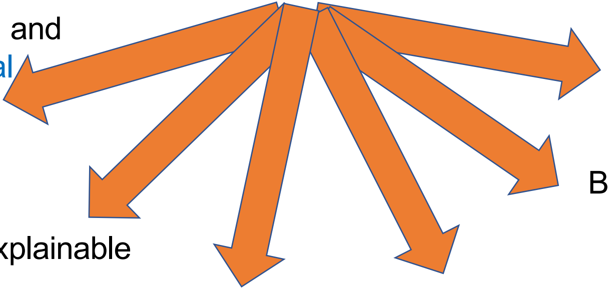
K. Mohan, P. London, M. Fazel, S.-I. Lee, and D. Witten.
Node-based learning of multiple gaussian graphical models

Backup Slides

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Machine learning for Biomedicine
Our Research Philosophy:



Able to provide and model biological explanations

Well-engineered software systems

Be Trustworthy

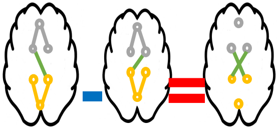
Be Explainable

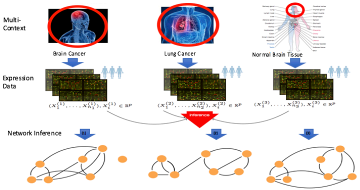
Be Accurate

Be Scalable

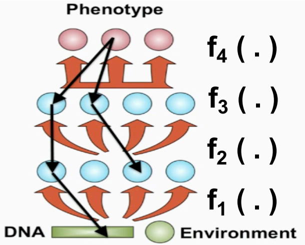
Overview of My Team's Three Research Topics

1. Fast and Scalable Learning Algorithms to Extract Related Graphs from Samples

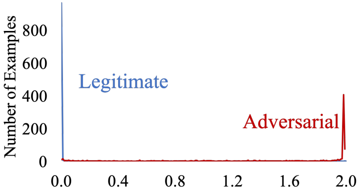




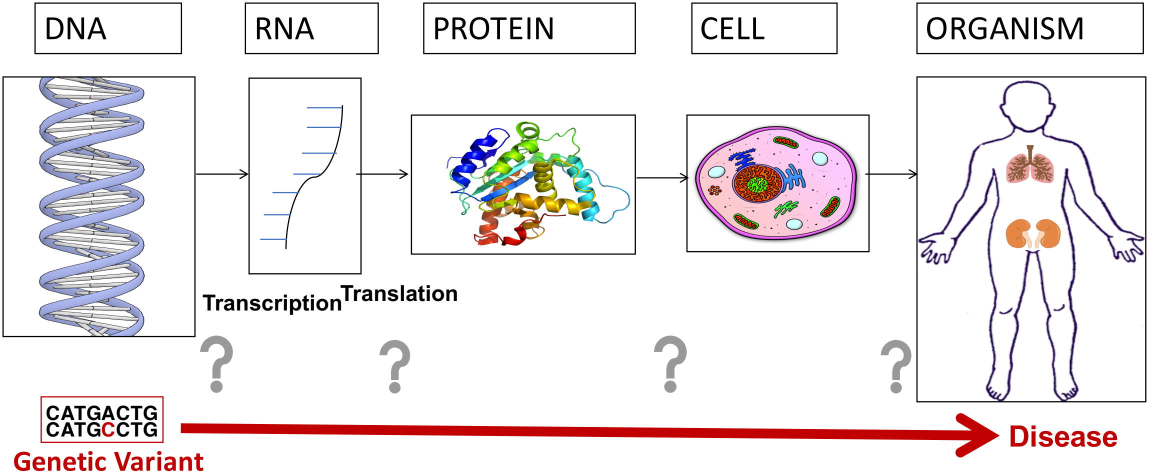
2. Making Explainable Deep Learning for Biomedicine



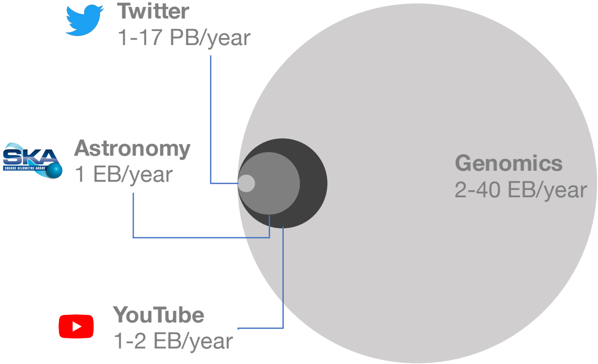
3. Making Deep Learning trustworthy

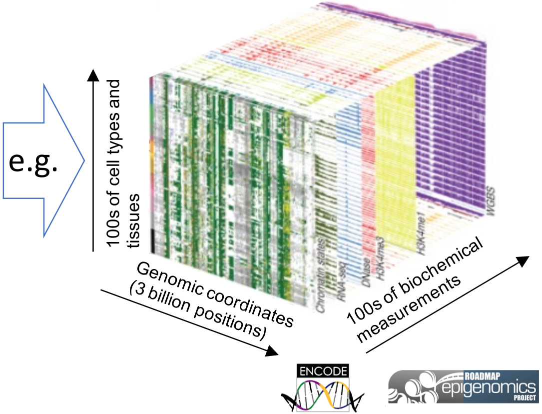


Biology in One Slide?



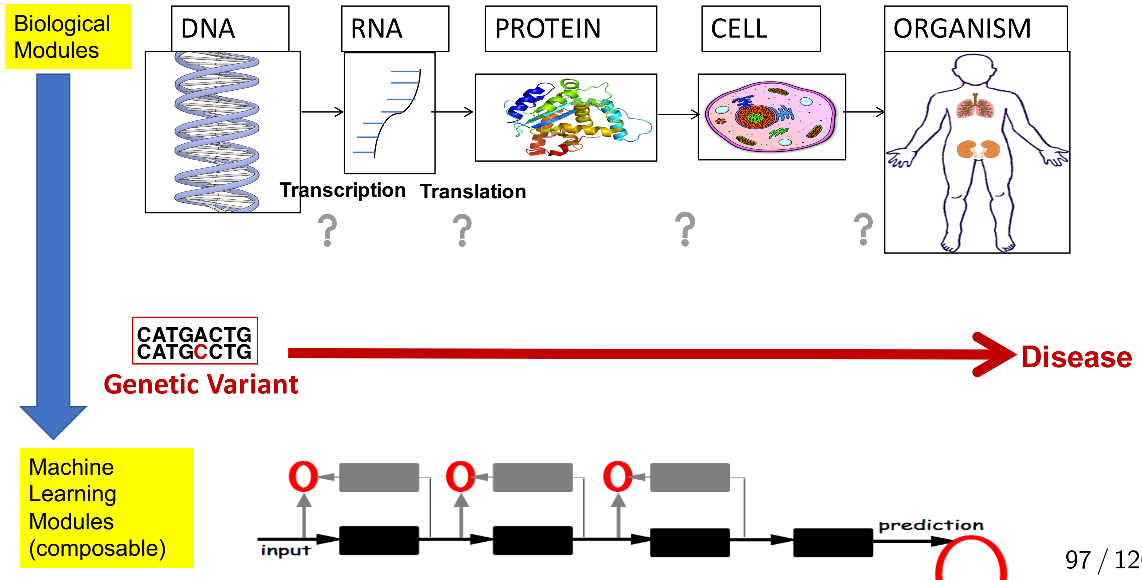
Big Data of Bio-Medicine





Adapted from Stephens ZD et al PLOS Biol 2015

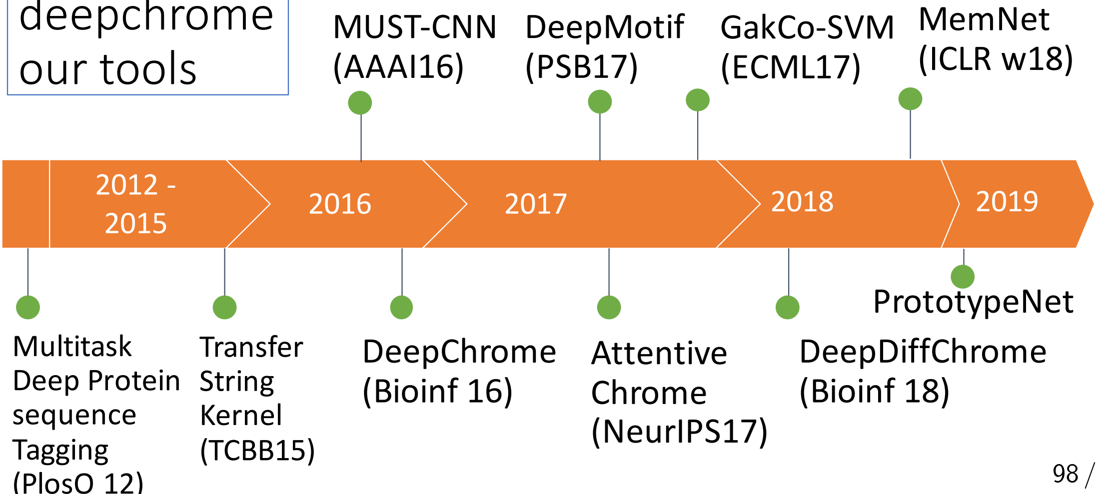
Last Year's Tutorial Talk Covered: deepChrome tools



Last Year's Tutorial Talk Covered: deepChrome tools

Timeline of deepchrome our tools

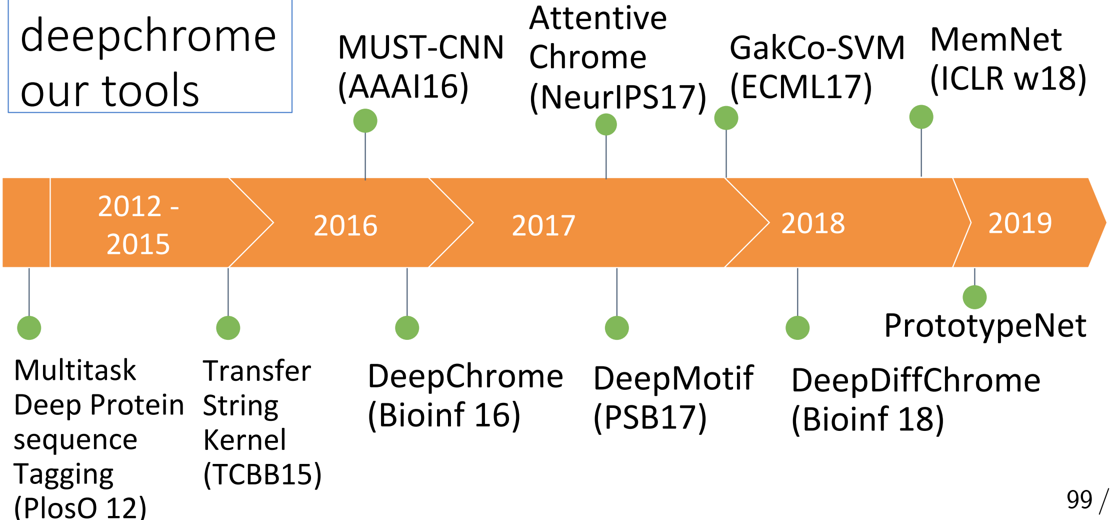
http://deepchrome.org/



Time line of our tools via deepchrome.org

Timeline of deepchrome our tools

http://deepchrome.org/



Deep Learning 2Read

A List of Deep Learning Papers We Read:

Deep Learning Readings Organized by Detailed Tags (2017 to Now)

https://qdata.github.io/deep2Read/

Besides using high-level categories, we also use the following detailed tags to label each read post we finished. Click on a tag to see relevant list of readings.

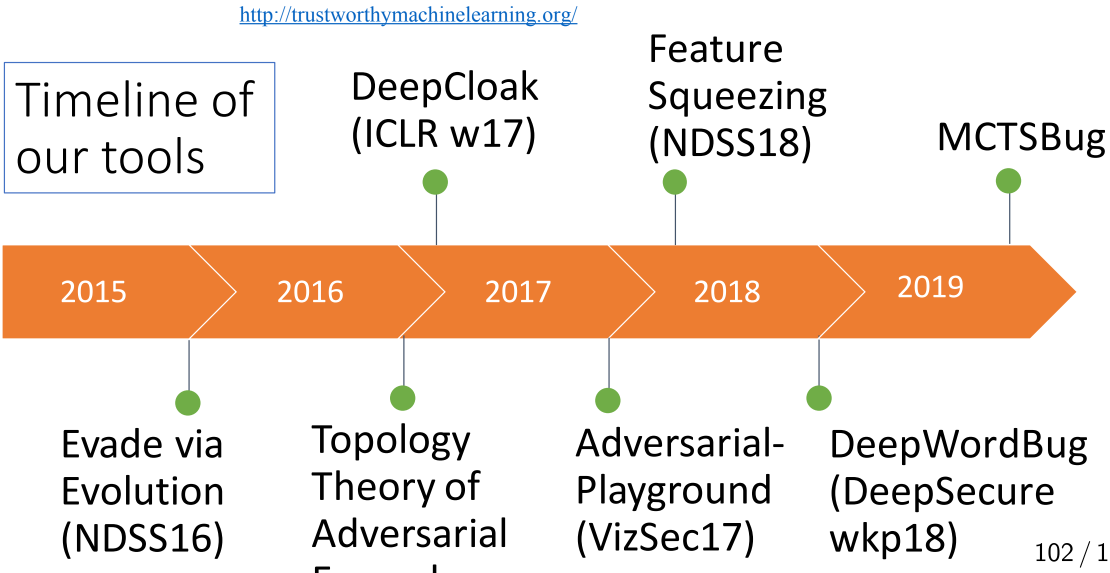
adversarial-examples | adversarial-loss | alphago | amortized | architecture-search | associative | attention | attribution | autoencoder | autoregressive | auxiliary | backprop | beam | bert | bias-variance | binary | black-box | blocking | brain | casual | certified-defense | composition | compression | crispr | cryptography | curriculum | denoising | dialog | difference-analysis | differentiation | dimension-reduction | discrete | distillation | distributed | dna | domain-adaptation | dynamic | ehr | em | embedding | expressive | few-shot | forcing | fuzzing | gan | generalization | generative | genomics | geometric | graph | graphical-model | hash | heterogeneous | hierarchical | high-dimensional | hyperparameter | imitation-learning | imputation | influence-functions | infomax | interpretable | invariant | knowledge-graph | learn2learn | low-rank | manifold | matching | matching-net | matrix-completion | memorization | memory | meta-learning | metamorphic | metric-learning | mimic | mobile | model-criticism | molecule | multi-label | multi-task | neural-programming | neuroscience | nlp | noise | nonparametric | ntm | optimization | parallel | parsimonious | planning | pointer | privacy | program | propagation | protein | pruning | qa | random | recommendation | relational | rl | rna | rnn | robustness | sample-quality | sampling | scalable | secure | semi-supervised | seq2seq | set | sketch | software-testing | sparsity | structured | stylometric | temporal-difference | text | transfer-learning | trees | understanding | value-networks | variational | verification | visualizing | white-box

[1]: adversarial-examples

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

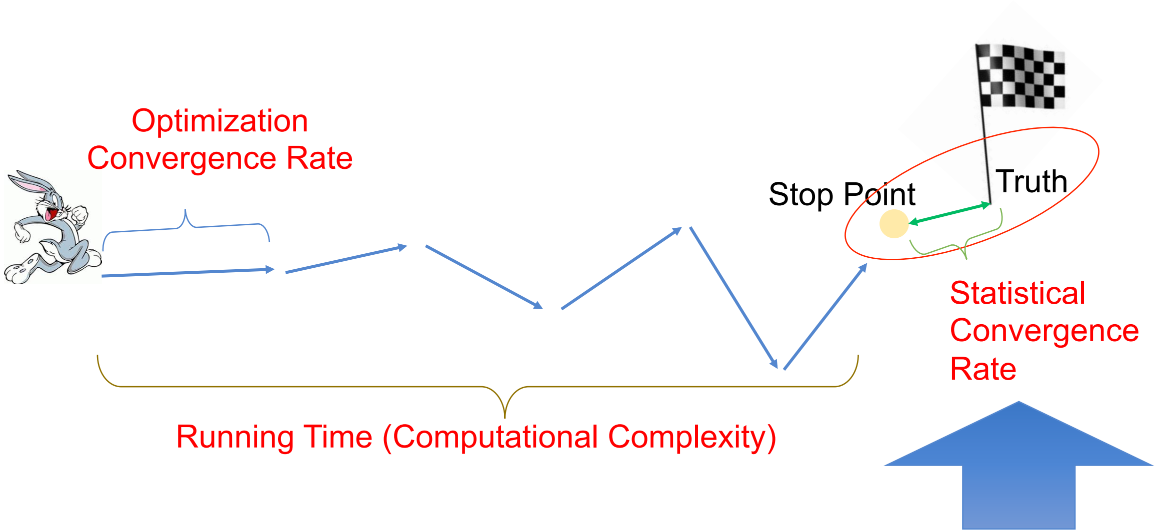
Time line of our tools via trustworthymachinelearning.org



Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

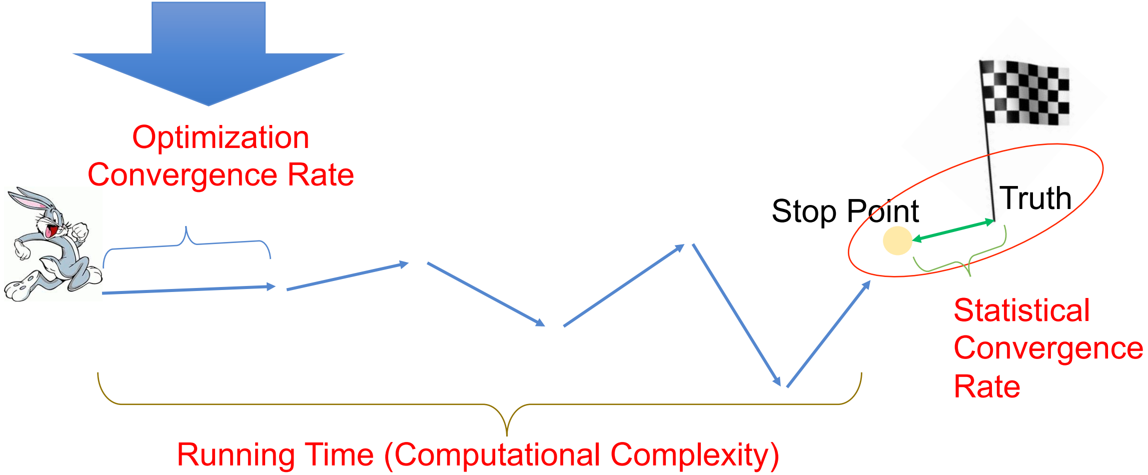
Overview Figure of the three rates: Statistical Convergence Rate



Statistical Convergence Rate: error bounds

- Suppose the model parameter you need to estimate is θ, the truth is θ^*
- $\| \theta - \theta^* \|$ or $\mathcal{R}(\theta - \theta^*)$. $\mathcal{R}]$ are mostly certain norm functions.
- When high-dimensional ($p > n$), many sparse estimators' error bounds relate to $\frac{\log p}{n}$.

Overview Figure of the three rates: Optimization Convergence Rate



Optimization Convergence Rate: optimization speed

- Linear, e.g. gradient descent, ADMM
- Higher order, e.g. quadratic
- Closed form solution, e.g. vanilla linear regression solution
- A rough comparison of speed: closed form $\geq$ Higher order $\geq$ linear;

Outline

1. Tasks in Joint Structure Learning from Heterogeneous Samples
   - http://jointnets.org
   - How to Measure Being Accurate and/or Scalable?
   - Correlation or Conditional Dependency?
   - From Heterogeneous Samples plus Knowledge beyond Samples
2. Joint Sparse GGMs: Methods and Variations
   - Basics: Sparse Gaussian Graphical Model (sGGM)
   - Method: Joint Graphical Lasso (JGL)
   - Method: SIMULE: Shared and Individual Parts of MULtiple sGGM Explicitly
   - Method Variation: NSIMULE: Gaussian to nonparanormal
   - Method Variation: WSIMULE: Adding Extra knowledge
   - Large Scale Variation of WSIMULE: JEEK
   - Large Scale Variation of Differential sGGM: DIFFEE
3. Backup Slides
   - Summary of Other Research: http://deepchrome.org
   - Summary of Other Research: http://trustworthymachinelearning.org
   - More about Convergence Rates:

Markov Random Field

Markov Random Field

Given an undirected graph $G = (V, E)$, a set of random variables $X = (X_v)_{v \in V}$ indexed by V form a Markov random field with respect to G if they satisfy the local Markov property:
A variable is conditionally independent of all other variables given its neighbors:
$X_v \perp\!\!\!\perp X_{V \setminus N(v)} | X_{N(v)}$

This property is stronger than the pairwise Markov property:

pairwise Markov property

Any two non-adjacent variables are conditionally independent given all other variables:
$X_u \perp\!\!\!\perp X_v \mid X_{V \setminus \{u,v\}} \quad \text{if } \{u, v\} \notin E$

Clique factorization

If this joint density can be factorized over the cliques of G:

$$p(X = x) = \prod_{C \in \mathrm{cl}(G)} \phi_C(x_C)$$

then X forms a Markov random field with respect to G. Here $\mathrm{cl}(G)$ is the set of cliques in G.

Log-linear Model

Any Markov random field can be written as log-linear model with feature functions f_k such that the full-joint distribution can be written as:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_k w_k^\top f_k(X)\right)$$

. Notice that the reverse doesn't hold.

Example I: Pairwise Model

Pairwise Model

$$P(X = x) = \frac{1}{Z(\Theta)} \exp\left(\sum_{s \in V} \theta_s^\top x_s^2 + \sum_{(s,t) \in E} \theta_{st}^\top x_s x_t\right)$$

.

Examples:

- Gaussian Graphical Model
- Ising Model

These two models have good estimators to infer the MRF. Generally, estimate Θ is difficult. Since it involves computing $Z(\Theta)$ or its derivatives.

Example I: Pairwise Model – Gaussian Case

Gaussian Case

$$f(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\mu)\right)}{\sqrt{(2\pi)^k|\Sigma|}}$$

.

Solution:

$$\ln \mathcal{L}(\bar{x}, \Omega) \propto \ln \det(\Omega) - \operatorname{tr}\left(\Omega \frac{1}{n}\sum_{i=1}^{n}(\bar{x}-\mu)(\bar{x}-\mu)^T\right) \tag{3.1}$$

$$= \ln \det(\Omega) - \operatorname{tr}\left(\Omega \widehat{S}\right) \tag{3.2}$$

where $\widehat{S}$ is the sample covariance matrix.

Ising Case

For the Ising model, we use generalized covariance matrix to avoid the normalization term.

Example II: Non-pairwise model – Nonparanormal Graphical Model

Are there any non-pairwise model which is easy to estimate?

Nonparanormal Graphical Model

$$P(X = x) = \frac{1}{Z} \exp\left(-\frac{1}{2}(f(x) - \mu)^T \Sigma^{-1} (f(x) - \mu)\right)$$

.

where $f(X) = (f_1(X_1), f_2(X_2), \ldots f_p(X_p))$ and each f_i is a univariate monotone function.
$f(X) \sim N(\mu, \Sigma)$.

Elementary Estimator (EE): Step I – Backward mapping

- Backward mapping $\mathcal{B}^*(\widehat{\phi})$ of the parameter (Solution of Vanilla Maximum Likelihood Estimator (MLE))
- Vanilla MLE: $\operatorname*{argmax}_{\theta} \mathcal{L}(\theta)$
  - Already close to true parameter
  - But without assumptions e.g., sparse
  - For instance, linear regression solution $(X^T X)^{-1} X^T Y$

Elementary Estimator: Step II – Optimization formulation

Elementary Estimator (EE)

$$\underset{\theta}{\operatorname{argmin}} \mathcal{R}(\theta) \quad \text{Subject to: } \mathcal{R}^*(\theta - \mathcal{B}^*(\widehat{\phi})) \leq \lambda_n \tag{3.3}$$

- Let $\mathcal{R}(\cdot) = \|\cdot\|_1$

$$\Downarrow$$

$$\underset{\theta}{\operatorname{argmin}} ||\theta||_1 \quad \text{Subject to: } ||\theta - \mathcal{B}^*(\widehat{\phi})||_\infty \leq \lambda_n \tag{3.4}$$

- Easy to prove the sharp convergence rate when $\mathcal{R}$ and $\mathcal{B}^*$ satisfy certain conditions.

EE-Benefit: Fast and scalable solution

- A soft-thresholding operator (closed form)
- Closed form & $O(p^2)$
- Easy to parallelize in GPU

$$\widehat{\theta} = S_{\lambda_n}(\mathcal{B}^*(\widehat{\phi}))$$

$$[S_\lambda(A)]_{ij} = \text{sign}(A_{ij}) \max(|A_{ij}| - \lambda, 0) \tag{3.5}$$

- Element-wise

$$\Sigma=\mathrm{Cov}(X)=\begin{bmatrix} \boxed{\sigma_{11}} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix} \quad \Sigma=\mathrm{Cov}(X)=\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \boxed{\sigma_{21}} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix} \quad \Sigma=\mathrm{Cov}(X)=\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \boxed{\sigma_{nn}} \end{bmatrix}$$

Apply same operator

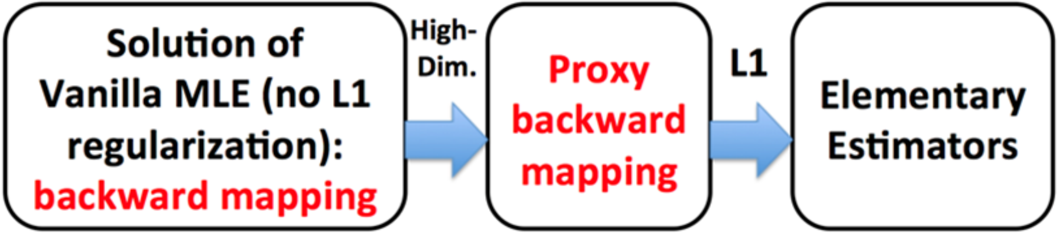
Independent calculation

EE-GM: Elementary Estimator for sGGM

- Vanilla MLE: $\underset{\Omega}{\operatorname{argmin}} - \log(\det(\Omega)) + < \Omega, \Sigma >$
- Backward mapping of Ω is Σ^{-1}
- Not invertible when $p \geq n$

EE-GM: Elementary Estimator for sGGM

- Vanilla MLE: $\underset{\Omega}{\operatorname{argmin}} - \log(\det(\Omega)) + < \Omega, \Sigma >$
- Backward mapping of Ω is Σ^{-1}
- Not invertible when $p \geq n$
- Need apporximated backward mapping
  - proxy backward mapping $\widehat{\theta}_n \approx \mathcal{B}^*(\widehat{\phi})$
  - In sGGM, $\widehat{\theta}_n = [T_v(\widehat{\Sigma})]^{-1}$



EE-GM: Elementary Estimator for sGGM

$$\underset{\theta}{\operatorname{argmin}} ||\theta||_1 \quad \text{Subject to: } ||\theta - \mathcal{B}^*(\widehat{\phi})||_\infty \leq \lambda_n \tag{3.6}$$

- $\widehat{\theta}_n = [T_v(\widehat{\Sigma})]^{-1}$

$\Downarrow$

EE-sGGM

$$\underset{\Omega}{\operatorname{argmin}} ||\Omega||_{1,,\text{off}} \quad \text{subject to:} ||\Omega - [T_v(\widehat{\Sigma})]^{-1}||_{\infty,\text{off}} \leq \lambda_n \tag{3.7}$$

EE	$\mathcal{R}(\cdot)$	θ	$\widehat{\theta}_n$	$\mathcal{R}^*$
EE-sGGM	$\|\|\cdot\|\|_1$	Ω	$[T_v(\widehat{\Sigma})]^{-1}$	$\|\|\cdot\|\|_\infty$

EE-Benefit: Easy to prove error bound

- Error bound:

$$||\widehat{\theta} - \theta^*||_\infty \le 2\lambda_n$$
$$||\widehat{\theta} - \theta^*||_F \le 4\sqrt{s}\lambda_n \tag{3.8}$$
$$||\widehat{\theta} - \theta^*||_1 \le 8s\lambda_n$$

- Condition:

$$\lambda_n \ge ||\widehat{\theta}_n - \theta^*||_\infty \tag{3.9}$$

- Constant: s is the num of non-zero entries.