Learning Loopy Graphical Models with Latent Variables: Efficient Methods and Guarantees

Anima Anandkumar

U.C. Irvine
Challenge: High-Dimensional Learning

Social Networks

Genetic Analysis

Financial Modeling

Neural Activity
Examples for Graphical Approaches

Modeling High-Dimensional Data
- Qualitative relationships: graph structure.
- Quantitative relationships: interaction strengths.

Topic Models
- Data: Word co-occurrences.
- Graph: Topic-word structure.

Financial Models
- Data: Stock returns.
- Graph: Company Classification.

Phylogenetics, Social Interactions, Computer Vision, ...
High-Dimensional Analysis

Steps Involved

- Estimate graph structure and strength of interactions.
- Employ the model to predict future behavior.

Focus on High-Dimensional Graph Estimation

- Graphical model on $p$ (labeled) nodes
- $n$ observations at the nodes

Challenges for High-Dimensional Estimation

- **Computational Complexity**: large $p$
- **Sample Complexity**: No. of samples $n$ for consistency ($p \gg n$)
- Presence of Hidden or Latent Variables

Goals

Tractable regimes, Novel methods, Provable guarantees
Walk-up: Learning Tree Models

Data processing inequality for Markov chains

\[ I(X_1; X_3) \leq I(X_1; X_2), I(X_2; X_3). \]

Tree Structure Estimation (Chow and Liu ‘68)

- **MLE**: Max-weight tree with estimated mutual information weights
Data processing inequality for Markov chains

\[ I(X_1; X_3) \leq I(X_1; X_2), I(X_2; X_3). \]

Tree Structure Estimation (Chow and Liu ‘68)

- **MLE**: Max-weight tree with estimated mutual information weights
- **Pairwise** statistics suffice
Data processing inequality for Markov chains

\[ I(X_1; X_3) \leq I(X_1; X_2), I(X_2; X_3). \]

Tree Structure Estimation (Chow and Liu ‘68)

- **MLE**: Max-weight tree with estimated mutual information weights
- **Pairwise** statistics suffice
- \( n \) samples and \( p \) nodes

Sample complexity: \( \frac{\log p}{n} = O(1). \)
Walk-up: Learning Tree Models

Data processing inequality for Markov chains

\[ I(X_1; X_3) \leq I(X_1; X_2), I(X_2; X_3). \]

Tree Structure Estimation (Chow and Liu ‘68)

- **MLE**: Max-weight tree with estimated mutual information weights
- **Pairwise** statistics suffice
- \( n \) samples and \( p \) nodes

Sample complexity: \( \frac{\log p}{n} = O(1). \)

What other classes of graphical models are tractable for learning?
Beyond Tree Models: Motivation

Topic Models

- Common words in different topics.
- Presence of latent or hidden variables.

Efficient Methods for High-dimensional Graph Estimation.
State of Art Approaches

Approaches Employed
Combinatorial approaches, Convex relaxation.

Algorithms for Structure Estimation

- Chow and Liu (68): Tree estimation
- Meinshausen and Bühlmann (06): Convex relaxation
- Ravikumar, Wainwright, Lafferty (10): Convex relaxation
- Bresler, Mossel and Sly (09): Bounded-degree graphs

Learning with Hidden Variables

- Erdös, et. al. (99): Latent trees
- Daskalakis, Mossel and Roch (06): Latent trees
- Chandrasekaran, Parrilo and Willsky (11): Latent Gaussian models
Summary of Results

Structure Estimation in Latent Variable Models

- **Number** of hidden variables and location unknown
- Estimate graph over all variables
Summary of Results

Structure Estimation in Latent Variable Models

- **Number** of hidden variables and location unknown
- Estimate graph over all variables
Summary of Results

Structure Estimation in Latent Variable Models

- **Number** of hidden variables and location unknown
- Estimate graph over all variables
Summary of Results

Structure Estimation in Latent Variable Models

- **Number** of hidden variables and location unknown
- Estimate graph over all variables

**Contributions**

- **Trees** and girth-constrained graphs.
- Algorithms based on pairwise statistics.
  - Local tests to recover global structure.
- Low sample and computational requirements
- Applicable in topic, financial and social domains

Graph Estimation in Loopy Models with Latent Variables
1. Introduction

2. Structure Estimation in Latent Graphical Models
   - Latent Tree Models
   - Loopy Latent Models

3. Experiments

4. Conclusion and Extensions
Learning Latent Graphical Models

- **Number and location** of hidden variables unknown
- Estimate graph over all variables
- **Trees and girth-constrained graphs**
Learning Latent Graphical Models

- Number and location of hidden variables unknown
- Estimate graph over all variables
- Trees and girth-constrained graphs

Information Distances $[d_{i,j}]$ for Tree Models

Gaussian: $d_{ij} := -\log |\rho_{ij}|$. Discrete: $d_{ij} := -\log |\text{Det}(P_{i,j})|$. 
Learning Latent Graphical Models

- Number and location of hidden variables unknown
- Estimate graph over all variables
- Trees and girth-constrained graphs

Information Distances $[d_{i,j}]$ for Tree Models

Gaussian: $d_{i,j} := -\log |\rho_{ij}|$. Discrete: $d_{i,j} := -\log |\text{Det}(P_{i,j})|$.

$[d_{i,j}]$ is an additive tree metric:

$$d_{k,l} = \sum_{(i,j) \in \text{Path}(k,l;E)} d_{i,j}.$$
Learning Latent Graphical Models

- Number and location of hidden variables unknown
- Estimate graph over all variables
- Trees and girth-constrained graphs

Information Distances $[d_{i,j}]$ for Tree Models

Gaussian: $d_{ij} := -\log \left| \rho_{ij} \right|$. Discrete: $d_{ij} := -\log \left| \text{Det}(P_{i,j}) \right|$.

$[d_{i,j}]$ is an additive tree metric: $d_{k,l} = \sum_{(i,j) \in \text{Path}(k,l;E)} d_{i,j}$.

- Extensions for multivariate linear models (A. et. al. ‘11)
Learning Latent Graphical Models

- **Number and location** of hidden variables unknown
- Estimate graph over all variables
- **Trees and girth-constrained graphs**

---

**Information Distances** \([d_{i,j}]\) for Tree Models

**Gaussian:** \(d_{ij} := -\log |\rho_{ij}|\).  
**Discrete:** \(d_{ij} := -\log |\text{Det}(P_{i,j})|\).

\([d_{i,j}]\) is an additive tree metric:  
\[
d_{k,l} = \sum_{(i,j) \in \text{Path}(k,l;E)} d_{i,j}.
\]

- Extensions for multivariate linear models (A. et. al. ‘11)

Learning latent tree using \([\hat{d}_{i,j}]\)
Siblings Test Based on Information Distances

Exact Statistics: Distances \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk}' < d_{i,j}\ \forall\ k, k' \neq i, j\), \(\iff\) \(i, j\) leaves with common parent
- \(\Phi_{ijk} = d_{i,j}, \ \forall\ k \neq i, j\), \(\iff\) \(i\) is a leaf and \(j\) is its parent.

Sample Statistics: ML Estimates \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships.
Siblings Test Based on Information Distances

Exact Statistics: Distances \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk'} < d_{i,j} \forall k, k' \neq i, j\), \(\iff\) \(i, j\) leaves with common parent

- \(\Phi_{ijk} = d_{i,j}, \forall k \neq i, j\), \(\iff\) \(i\) is a leaf and \(j\) is its parent.

Sample Statistics: ML Estimates \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships
Siblings Test Based on Information Distances

Exact Statistics: Distances \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk'} < d_{i,j} \ \forall \ k, k' \neq i, j, \iff i, j \) leaves with common parent
- \(\Phi_{ijk} = d_{i,j}, \forall k \neq i, j, \iff i \) is a leaf and \(j \) is its parent.

Sample Statistics: ML Estimates \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships
**Siblings Test Based on Information Distances**

**Exact Statistics: Distances** \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk'} < d_{i,j} \ \forall \ k, k' \neq i, j\), \(\iff\) \(i, j\) leaves with common parent
- \(\Phi_{ijk} = d_{i,j}, \ \forall \ k \neq i, j\), \(\iff\) \(i\) is a leaf and \(j\) is its parent.

**Sample Statistics: ML Estimates** \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships
Siblings Test Based on Information Distances

Exact Statistics: Distances \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk}' < d_{i,j} \ \forall \ k, k' \neq i, j\), \(\iff\) \(i, j\) leaves with common parent
- \(\Phi_{ijk} = d_{i,j}, \ \forall \ k \neq i, j\), \(\iff\) \(i\) is a leaf and \(j\) is its parent.

Sample Statistics: ML Estimates \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships
Siblings Test Based on Information Distances

Exact Statistics: Distances \([d_{i,j}]\)

Let \(\Phi_{ijk} := d_{i,k} - d_{j,k}\).

- \(-d_{i,j} < \Phi_{ijk} = \Phi_{ijk'} < d_{i,j} \forall k, k' \neq i, j, \iff i, j \text{ leaves with common parent}\)
- \(\Phi_{ijk} = d_{i,j}, \forall k \neq i, j, \iff i \text{ is a leaf and } j \text{ is its parent.}\)

Sample Statistics: ML Estimates \([\hat{d}_{i,j}]\)

Use only short distances: \(d_{i,k}, d_{j,k} < \tau\), Relax equality relationships
Recursive Grouping

Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Recursive Grouping

Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Recursive Grouping

Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Recursive Grouping

Recursive Grouping Algorithm (Choi, Tan, A., Willsky)

- Sibling test and remove leaves
- Build tree from bottom up
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$. 

[Diagram of trees]
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $\hat{d}_{i,j}$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Chow-Liu Based Grouping Algorithm

Efficient Initial Tree on Observed Nodes (MST)

Minimum spanning tree using edge weights $[\hat{d}_{i,j}]$.

Chow-Liu Based Grouping (Choi, Tan, A., Willsky ‘11)
Proof Ideas

Relating Chow-Liu Tree with Latent Tree

- Surrogate $Sg(i)$ for node $i$: observed node with strongest correlation
  \[
  Sg(i) := \arg\min_{j \in V} d_{i,j}
  \]

- Neighborhood preservation
  \[(i, j) \in T \Rightarrow (Sg(i), Sg(j)) \in T_{ML}.
  \]

Chow-Liu grouping reverses edge contractions
Proof by induction
Motivation: Topic Models

- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs

- Pairwise statistics not related to trees in general.
Motivation: Topic Models

- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs

- Pairwise statistics not related to trees in general
Loopy Graphical Models with Latent Nodes

Motivation: Topic Models
- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs
- Pairwise statistics not related to trees in general.
Loopy Graphical Models with Latent Nodes

Motivation: Topic Models
- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs
- Pairwise statistics not related to trees in general
- Under weak interactions (absence of long range correlations), local statistics converge to a tree limit.
Motivation: Topic Models
- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs
- Pairwise statistics not related to trees in general
- Under weak interactions (absence of long range correlations), local statistics converge to a tree limit.
Loopy Graphical Models with Latent Nodes

Motivation: Topic Models
- Common words among topics.
- Latent or hidden nodes.
- Typically long cycles: Locally tree-like.

Latent Models on Large Girth Graphs
- Pairwise statistics not related to trees in general
- Under weak interactions (absence of long range correlations), local statistics converge to a tree limit.

Local additivity \( d_{k,l} \approx \sum_{(i,j) \in \text{Path}(k,l;E)} d_{i,j} \).
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- **Merge** the MSTs to obtain a loopy graph
- Run **latent tree routine** on different local neighborhoods
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building **local MST**
- **Merge** the MSTs to obtain a loopy graph
- Run **latent tree routine** on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building local MST
- Merge the MSTs to obtain a loopy graph
- Run latent tree routine on different local neighborhoods

Original Graph

Local CL Grouping
Overview of Proposed Method

- Consider local neighborhoods for building **local MST**
- **Merge** the MSTs to obtain a loopy graph
- Run **latent tree routine** on different local neighborhoods

Original Graph

Local CL Grouping
Guarantees for Latent Structure Learning

- Depth $\delta$: worst-case distance between hidden and observed nodes.
- Parameter $\beta$: depends on min. and max. node and edge potentials
  - $\beta = 1$ for homogeneous models.

---

**Theorem (A., Valluvan ‘12)**

Proposed method correctly recovers graph structure w.h.p. on $p$ observed nodes and $n$ samples when

$$\frac{J_{\min}^{-2\delta\beta(\beta+1)-2} \log p}{n} = O(1).$$
Guarantees for Latent Structure Learning

- Depth $\delta$: worst-case distance between hidden and observed nodes.
- Parameter $\beta$: depends on min. and max. node and edge potentials
  - $\beta = 1$ for homogeneous models.

Theorem (A. Valluvan ‘12)
Proposed method correctly recovers graph structure w.h.p. on $p$ observed nodes and $n$ samples when

$$n \geq \Omega(J_{\min}^{-2\delta(\beta+1)-2}\log p) = O(1).$$

- Fully observed case $\delta = 0$: $n = \Omega(J_{\min}^{-2}\log p)$.

Latent Models on Large Girth Graphs Akin to Latent Trees
Insights and Implications

Tradeoff between depth $\delta$ and girth $g$

Roughly require: $\delta < g/4$.

Tradeoff between max. edge strength $J_{\text{max}}$ and degree $\Delta$

Require $J_{\text{max}} < \tanh(\Delta^{-1})$. 
Insights and Implications

Tradeoff between depth $\delta$ and girth $g$

Roughly require: $\delta < g/4$.

Tradeoff between max. edge strength $J_{\text{max}}$ and degree $\Delta$

Require $J_{\text{max}} < \tanh(\Delta^{-1})$.

Sample complexity for uniform node sampling

Given $\rho$ fraction of nodes as observed nodes,

$$n = \Omega \left( \Delta^2 \rho^{-4} \log p^5 \right).$$

Necessary conditions for structure recovery

For any deterministic algorithm, the number of samples $n$ needs to be

$$n = \Omega \left( \frac{\Delta_{\text{min}}}{\rho} \log p \right)$$
Insights and Implications

Tradeoff between depth $\delta$ and girth $g$

Roughly require: $\delta < g/4$.

Tradeoff between max. edge strength $J_{max}$ and degree $\Delta$

Require $J_{max} < \text{atanh}(\Delta^{-1})$.

Sample complexity for uniform node sampling

Given $\rho$ fraction of nodes as observed nodes,

$$n = \Omega \left( \Delta^2 \rho^{-4} (\log p)^5 \right).$$

Necessary conditions for structure recovery

For any deterministic algorithm, the number of samples $n$ needs to be

$$n = \Omega \left( \frac{\Delta_{\text{min}} \log p}{\rho} \right).$$

Efficient Method for Learning Loopy Latent Models
Outline

1 Introduction

2 Structure Estimation in Latent Graphical Models
   - Latent Tree Models
   - Loopy Latent Models

3 Experiments

4 Conclusion and Extensions
Newsgroup Data
Stock Returns Data
Stock Returns Data
Outline

1. Introduction

2. Structure Estimation in Latent Graphical Models
   - Latent Tree Models
   - Loopy Latent Models

3. Experiments

4. Conclusion and Extensions
Summary and Outlook

Summary

- High-dimensional estimation via graphical approaches
- Model classes where learning is tractable
- Efficient methods for learning
- Guarantees on sample and computational complexities
Summary

- High-dimensional estimation via **graphical** approaches
- Model **classes** where learning is tractable
- Efficient **methods** for learning
- **Guarantees** on sample and computational complexities

Outlook

- Removing girth constraint on latent models
- Characterizing criterion for tractable learning
- Learning beyond regime of correlation decay
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators

- Naïve search for separators is exponential in maximum node degree
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators

- Naïve search for separators is exponential in maximum node degree
- Novel criterion: sparse local separation
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators

- Naïve search for separators is exponential in maximum node degree
- Novel criterion: sparse local separation
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators

- Naïve search for separators is exponential in maximum node degree
- Novel criterion: sparse local separation
- A large family of graphs have sparse local separators: e.g small world, Erdős-Rényi.
Extensions

Structure Estimation in Random Graph Models

- Fully observed models (no latent nodes)
- Random graph models such as Erdős-Rényi and small world

Graph Estimation Through Search for Vertex Separators

- Naïve search for separators is exponential in maximum node degree
- Novel criterion: sparse local separation
- A large family of graphs have sparse local separators: e.g. small world, Erdős-Rényi

Novel Criteria for High-Dimensional Estimation
Extensions and Connections

Topology Discovery With Few Participants (A. Hassidim, Kelner ‘11)

- End-to-end delays between participants in Erdős-Rényi random graph
- Edit distance guarantees with vanishing fraction of participants
Extensions and Connections

Topology Discovery With Few Participants (A. , Hassidim, Kelner ‘11)

- End-to-end delays between participants in Erdős-Rényi random graph
- Edit distance guarantees with vanishing fraction of participants

---

Covariance Decomposition

- Multiple graphs: combination of statistical relationships
- Markov and Independence Domains

\[
\begin{bmatrix}
\end{bmatrix} = \begin{bmatrix}
\end{bmatrix}^{-1} + \begin{bmatrix}
\end{bmatrix}
\]
Extensions and Connections

Topology Discovery With Few Participants (A., Hassidim, Kelner ‘11)

- End-to-end delays between participants in Erdős-Rényi random graph
- Edit distance guarantees with vanishing fraction of participants

Covariance Decomposition

- Multiple graphs: combination of statistical relationships
- Markov and Independence Domains

\[
\begin{bmatrix}
\end{bmatrix}
= \begin{bmatrix}
\end{bmatrix}^{-1} + \begin{bmatrix}
\end{bmatrix}
\]

Graphical Model Mixtures

- Multiple graphs: context specific dependencies
- Hidden context
- Learning guarantees
The Big Picture

- Method of moments
- Algorithms and complexity
- Statistical physics
- Spectral analysis
- High-dimensional estimation via graphical methods
- Information theory
- Random graph models

http://newport.eecs.uci.edu/anandkumar