# Large Scale Gaussian Copula Precision Estimation with Missing Values

Arindam Banerjee
Department of Computer Science & Engineering
University of Minnesota, Twin Cities

Collaborators:
Huahua Wang, Farideh Fazayeli, Soumyadeep Chatterjee (UMN)
Cho-Jui Hsieh, Pradeep Ravikumar, Inderjit Dhillon (UT Austin)

ICML 2014 Workshop
Covariance Selection and Graphical Model Structure Learning
June 26, 2014

# Overview

- Structure learning in graphical models
  - Conditional independence in multivariate distributions
- Multi-variate Gaussians and Gaussian copulas
  - Gaussian: Inverse of covariance matrix gives structure
  - Gaussian Copulas: Semi-parametric generalization of Gaussians
- Copula precision estimation with missing values: Statistics
  - Idea: Act as if there are no missing values
  - Finite sample guarantees on estimation error
- Copula precision estimation: Computation
  - Idea: Lifting, linear programming
  - Parallel inexact alternating direction method

# Structure Learning in Graphical Models

- Consider multivariate distribution over $X = (X_1, \ldots, X_p)$
- Graphical models assume a factorization structure:

$$p(X) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \phi_C(X_C)$$

  - $C$ is a clique, $\mathcal{C}$ is the set of all cliques
  - $X_C$ is a small set variables, e.g., $X_C = \{X_2, X_3, X_7\}$
  - Product over local factors $\phi_C(X_C)$
  - Each variable $X_j$ can be in multiple factors
- Examples: Most probabilistic models used in practice
  - Bayesian networks, Markov random fields
  - Mixture of Gaussians, hidden Markov models, naive-Bayes
- Can we 'learn' the factorization structure from samples?

# The CLIME Estimator: Gaussian Precision

- Given $n$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim N_p(0, \Sigma_0)$

# The CLIME Estimator: Gaussian Precision

- Given $n$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim N_p(0, \Sigma_0)$
- Estimate the sample covariance $\hat{S}_n = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{n} \hat{\bar{\mathbf{x}}} \hat{\bar{\mathbf{x}}}^T$

# The CLIME Estimator: Gaussian Precision

- Given $n$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim N_p(0, \Sigma_0)$
- Estimate the sample covariance $\hat{S}_n = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{n} \hat{\bar{\mathbf{x}}} \hat{\bar{\mathbf{x}}}^T$
- Estimate precision $\hat{\Theta}_n$ by solving [Cai et al., 2011, 2014]

$$\min \|\hat{\Theta}\|_1 \quad \text{s.t.} \quad \|\hat{S}_n \hat{\Theta} - \mathbb{I}\|_\infty \leq \lambda_n$$

# The CLIME Estimator: Gaussian Precision

- Given $n$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim N_p(0, \Sigma_0)$
- Estimate the sample covariance $\hat{S}_n = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{n}\hat{\bar{\mathbf{x}}}\hat{\bar{\mathbf{x}}}^T$
- Estimate precision $\hat{\Theta}_n$ by solving [Cai et al., 2011, 2014]

$$\min \|\hat{\Theta}\|_1 \quad \text{s.t.} \quad \|\hat{S}_n\hat{\Theta} - \mathbb{I}\|_\infty \leq \lambda_n$$

- Consider 'sparse' precision family: For $0 \leq q < 1$

$$\mathcal{U}(M, q, s_0(p)) = \left\{\Theta : \Theta \succ 0, |||\Theta|||_1 \leq M, \max_{1 \leq i \leq p} \sum_{j=1}^{p} |\theta_{ij}|^q \leq s_0(p)\right\}$$

# The CLIME Estimator: Gaussian Precision

- Given $n$ samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim N_p(0, \Sigma_0)$
- Estimate the sample covariance $\hat{S}_n = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{n}\hat{\bar{\mathbf{x}}}\hat{\bar{\mathbf{x}}}^T$
- Estimate precision $\hat{\Theta}_n$ by solving [Cai et al., 2011, 2014]

$$\min \|\hat{\Theta}\|_1 \quad \text{s.t.} \quad \|\hat{S}_n\hat{\Theta} - \mathbb{I}\|_\infty \leq \lambda_n$$

- Consider 'sparse' precision family: For $0 \leq q < 1$

$$\mathcal{U}(M, q, s_0(p)) = \left\{\Theta : \Theta \succ 0, |||\Theta|||_1 \leq M, \max_{1 \leq i \leq p} \sum_{j=1}^{p} |\theta_{ij}|^q \leq s_0(p)\right\}$$

- Let $\Theta_0 \in \mathcal{U}(M, q, s_0(p))$. If $\lambda_n \geq |||\Theta_0|||_1 \|\hat{S}_n - \Sigma_0\|_\infty$

$$\|\hat{\Theta}_n - \Theta_0\|_\infty \leq 4|||\Theta_0|||_1 \lambda_n \ ,$$

$$\left|\left|\left|\hat{\Theta}_n - \Theta_0\right|\right|\right|_2 \leq C s_0(p)(4|||\Theta_0|||_1)^{1-q} \lambda_n^{1-q} \ ,$$

$$\frac{1}{p}\left|\left|\left|\hat{\Theta}_n - \Theta_0\right|\right|\right|_F^2 \leq C s_0(p)(4|||\Theta_0|||_1)^{2-q} \lambda_n^{2-q} \ ,$$

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$
- Consider $\hat{S}_n$ for sub-Gaussian distributions

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$
- Consider $\hat{S}_n$ for sub-Gaussian distributions
- Perturbation matrix $\Delta$ with sub-exponential entries $\Delta_{ij}$

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$
- Consider $\hat{S}_n$ for sub-Gaussian distributions
- Perturbation matrix $\Delta$ with sub-exponential entries $\Delta_{ij}$
  - Perturbations $\Delta_{ij}$: independent, zero-mean

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$
- Consider $\hat{S}_n$ for sub-Gaussian distributions
- Perturbation matrix $\Delta$ with sub-exponential entries $\Delta_{ij}$
  - Perturbations $\Delta_{ij}$: independent, zero-mean
  - Bounded sub-exponential norm $\|\Delta_{ij}\|_{\psi_1} \leq 4\|X_{i\cdot}\|^2_{\psi_2}$

# The CLIME Estimator: Gaussian Precision

- Analysis needs bounds on $\|\hat{S}_n - \Sigma_0\|_\infty$
- Consider $\hat{S}_n$ for sub-Gaussian distributions
- Perturbation matrix $\Delta$ with sub-exponential entries $\Delta_{ij}$
  - Perturbations $\Delta_{ij}$: independent, zero-mean
  - Bounded sub-exponential norm $\|\Delta_{ij}\|_{\psi_1} \leq 4\|X_{i\cdot}\|_{\psi_2}^2$
- With probability at least $(1 - c_1 p^{-c_2})$

$$\|\hat{S}_n + \Delta - \Sigma_0\|_\infty \leq c_3 \sqrt{\frac{\log p}{n}}$$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian
    - Equivalent to Gaussian copulas (Tsukahara, 2005)

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian
    - Equivalent to Gaussian copulas (Tsukahara, 2005)
- Flexible family of semi-parametric distributions

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian
    - Equivalent to Gaussian copulas (Tsukahara, 2005)
- Flexible family of semi-parametric distributions
  - Needs to hold for some set of functions $f = \{f_1, \ldots, f_p\}$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian
    - Equivalent to Gaussian copulas (Tsukahara, 2005)
- Flexible family of semi-parametric distributions
  - Needs to hold for some set of functions $f = \{f_1, \ldots, f_p\}$
  - Each $f_j$ needs to be monotonic, e.g., $x_1 < x_2 \Leftrightarrow f_j(x_1) < f_j(x_2)$

# Gaussians Copula Distributions

- Multivariate distribution family over $X = (X_1, \ldots, X_p)$
  - Let $f = \{f_1, \ldots, f_p\}$ be a set of monotonic functions
  - Consider $Z_j = f_j(X_j), j = 1, \ldots, p$, and $Z = (Z_1, \ldots, Z_p)$
  - $Z$ follows $N(0, \Sigma_0)$, $\Sigma_0$ is a correlation matrix
  - Then $X \sim NPN_p(f, \Sigma_0)$
    - The 'non-paranormal' distribution (Liu et al., 2009, 2012)
    - Some monotonic transform of $X$ is multivariate Gaussian
    - Equivalent to Gaussian copulas (Tsukahara, 2005)
- Flexible family of semi-parametric distributions
  - Needs to hold for some set of functions $f = \{f_1, \ldots, f_p\}$
  - Each $f_j$ needs to be monotonic, e.g., $x_1 < x_2 \Leftrightarrow f_j(x_1) < f_j(x_2)$
  - $f = \{f_1, \ldots, f_p\}$ need not be known

# Gaussian Copula: Precision Estimation

- Factorization structure of Gaussian copula (Liu et al., 2012)
  - Recall $Z = (f_1(X_1), \ldots, f_p(X_p)) \sim N(0, \Sigma_0)$
  - As before, factorization is given by zeros of $\Theta_0 = \Sigma_0^{-1}$
  - Since $f_j(X_j)$ are scaler functions
    - Do not change conditional independence, hence factorization
    - $X$ and $Z$ have the same structure

- Given samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$ from $X \sim NPN_p(f, \Sigma_0)$
  - Goal: Estimate sparse precision matrix $\Theta_0 = \Sigma_0^{-1}$
  - Functions $\{f_j\}$ are unknown
  - Some monotonic transform of $X$ has covariance $\Sigma_0$

# Gaussian Copula: Rank Correlations

- Ranking structure in samples $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$

# Gaussian Copula: Rank Correlations

- Ranking structure in samples $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$
  - $r_i^j$: rank of $x_i^j$ among $x_1^j, \ldots, x_n^j$

# Gaussian Copula: Rank Correlations

- Ranking structure in samples $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$
  - $r_i^j$: rank of $x_i^j$ among $x_1^j, \ldots, x_n^j$
  - Average rank $\bar{r}^j = \frac{1}{n} \sum_{i=1}^{n} r_i^j = \frac{n+1}{2}$

# Gaussian Copula: Rank Correlations

- Ranking structure in samples $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$
  - $r_i^j$: rank of $x_i^j$ among $x_1^j, \ldots, x_n^j$
  - Average rank $\bar{r}^j = \frac{1}{n}\sum_{i=1}^n r_i^j = \frac{n+1}{2}$
- Rank correlations

$$\text{(Kendall's } \tau) \quad \hat{\tau}_{jk} = \frac{1}{n(n-1)} \sum_{\substack{i,i'=1 \\ i \neq i'}}^{n} \text{sign}((x_i^j - x_{i'}^j)(x_i^k - x_{i'}^k))$$

$$\text{(Spearman's } \rho) \quad \hat{\rho}_{jk} = \frac{\sum_{i=1}^n (r_i^j - \bar{r}^j)(r_i^k - \bar{r}^k)}{\sqrt{\sum_{i=1}^n (r_i^j - \bar{r}^j)^2} \cdot \sqrt{\sum_{i=1}^n (r_i^k - \bar{r}^k)^2}}$$

# Gaussian Copula: Rank Correlations

- Ranking structure in samples $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$
  - $r_i^j$: rank of $x_i^j$ among $x_1^j, \ldots, x_n^j$
  - Average rank $\bar{r}^j = \frac{1}{n}\sum_{i=1}^n r_i^j = \frac{n+1}{2}$
- Rank correlations

$$\text{(Kendall's } \tau) \quad \hat{\tau}_{jk} = \frac{1}{n(n-1)} \sum_{\substack{i,i'=1 \\ i \neq i'}}^{n} \text{sign}((x_i^j - x_{i'}^j)(x_i^k - x_{i'}^k))$$

$$\text{(Spearman's } \rho) \quad \hat{\rho}_{jk} = \frac{\sum_{i=1}^n (r_i^j - \bar{r}^j)(r_i^k - \bar{r}^k)}{\sqrt{\sum_{i=1}^n (r_i^j - \bar{r}^j)^2} \cdot \sqrt{\sum_{i=1}^n (r_i^k - \bar{r}^k)^2}}$$

- [Kendall, 1948; Kruskal, 1958] Assuming $X \sim NPN_p(f, \Sigma_0)$

$$\Sigma_{jk}^0 = \sin\left(\frac{\pi}{2}\tau_{jk}\right) = 2\sin\left(\frac{\pi}{6}\rho_{jk}\right)$$

# The CLIME Estimator: Copula Precision

- Step 1: Estimate the copula correlation using K-K result (Liu et al., 2012, Xue et al., 2012)
  - From finite sample Kendall's $\tau$

$$\hat{S}^{\tau}_{jk} == \begin{cases} \sin\left(\frac{\pi}{2}\hat{\tau}_{jk}\right) & , \quad j \neq k \\ 1, & \quad j = k \end{cases}$$

  - From finite sample Spearman's $\rho$

$$\hat{S}^{\rho}_{jk} = \begin{cases} 2\sin\left(\frac{\pi}{6}\hat{\rho}_{jk}\right) & , \quad j \neq k \\ 1, & \quad j = k \end{cases}$$

- Step 2: Plug-in estimated correlation $\hat{S}$ (any one) to CLIME

$$\min \|\hat{\Theta}\|_1 \quad \text{s.t.} \quad \|\hat{S}\hat{\Theta} - \mathbb{I}\|_\infty \leq \lambda_n$$

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$
  - Effective number of samples $n_{jk} = \sum_{i=1}^{n} b_{ij} b_{ik}$

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$
  - Effective number of samples $n_{jk} = \sum_{i=1}^{n} b_{ij} b_{ik}$
- Several issues with missing values

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$
  - Effective number of samples $n_{jk} = \sum_{i=1}^{n} b_{ij} b_{ik}$
- Several issues with missing values
  - Need to define rank correlations with missing values

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$
  - Effective number of samples $n_{jk} = \sum_{i=1}^{n} b_{ij} b_{ik}$
- Several issues with missing values
  - Need to define rank correlations with missing values
  - Different samples $i$ are involved in different pairs $(j, k)$

# Copula Precision Estimation with Missing Values

- Data with missing values is increasingly common
- Entries in $\mathbf{x}_i$ (samples) missing with probability $\delta$
  - Let $b_{ij} = 0$ if $x_i^j$ is missing, and 1 otherwise
  - Then $P(b_{ij} = 0) = \delta, P(b_{ij} = 1) = 1 - \delta$
  - Effective number of samples $n_{jk} = \sum_{i=1}^{n} b_{ij} b_{ik}$
- Several issues with missing values
  - Need to define rank correlations with missing values
  - Different samples $i$ are involved in different pairs $(j, k)$
  - The effective number of samples $n_{jk}$ are dependent

# DoPinG Copula Estimator

- Double Plug-in Gaussian Copula Estimator
  - Estimate rank correlations: $\hat{\tau}_{jk}$ or $\hat{\rho}_{jk}$
  - Apply K-K formula: $\hat{S}^{\tau}$ or $\hat{S}^{\rho}$
  - Estimate $\hat{\Theta}$ based on CLIME using $\hat{S}$
- Idea: Act as if there is no missing value!
- Question: Will $\hat{\Theta}$ be a consistent estimate of $\Theta_0$?

# Kendall's $\tau$ with missing values

- Pairwise Kendall's $\tau$ with missing value

$$\hat{\tau}_{jk} = \frac{1}{n_{jk}(n_{jk}-1)} \sum_{\substack{i,i'=1 \\ i \neq i'}}^{n} b_{ij} b_{ik} b_{i'j} b_{i'k} \ \text{sign}((x_i^j - x_{i'}^j)(x_i^k - x_{i'}^k))$$

- Apply K-K formula, to get $\hat{S}^{\tau}_{jk}$
  - Matrix $\hat{S}^{\tau}$ need not be positive semi-definite
- Main Result: With high probability

$$\|\hat{S}^{\tau} - \Sigma_0\|_{\infty} \leq O\left(\frac{1}{1-\delta}\sqrt{\frac{\log p}{n}}\right)$$

- Leads to bounds on $\left|\left|\left|\hat{\Theta} - \Theta_0\right|\right|\right|_2$, and other norms

# Spearman's $\rho$ with missing values

- Pairwise Spearman's $\rho$ with missing value
  - Let $r_i^j$ be the rank of $x_i^j$ among $n_{jk}$ samples
  - Effective mean rank $\bar{r}^{jk} = \frac{1}{n_{jk}} \sum_{i=1}^{n} r_i^j b_{ij} b_{ik} = \frac{1}{n_{jk}} \sum_{i=1}^{n} r_i^k b_{ij} b_{ik}$

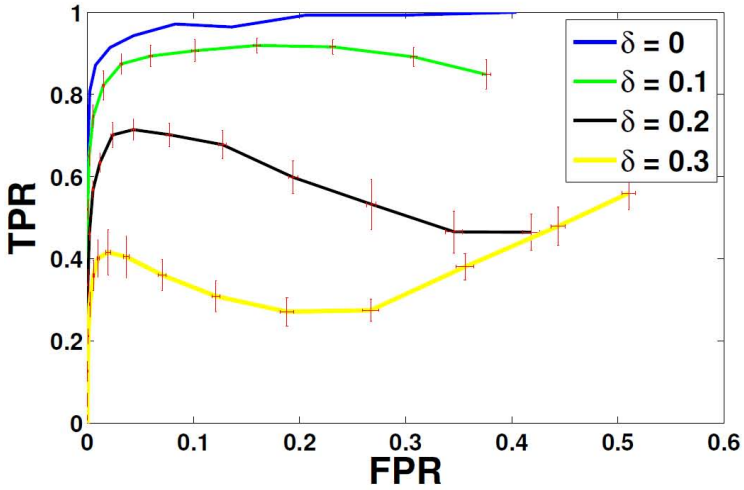
$$\hat{\rho}^{jk} = \frac{\sum_{i=1}^{n} (r_i^j - \bar{r}^{jk})(r_i^k - \bar{r}^{jk}) b_{ij} b_{ik}}{\sqrt{\sum_{i=1}^{n} (r_i^j - \bar{r}^{jk})^2 b_{ij} b_{ik}} \cdot \sqrt{\sum_{i=1}^{n} (r_i^k - \bar{r}^{jk})^2 b_{ij} b_{ik}}}$$

- Apply K-K formula, to get $\hat{S}^{\rho}_{jk}$
  - Matrix $\hat{S}^{\rho}$ need not be positive semi-definite
- Main Result: For $n \geq \frac{c}{(1-\delta)^2 \log p}$, with high probability

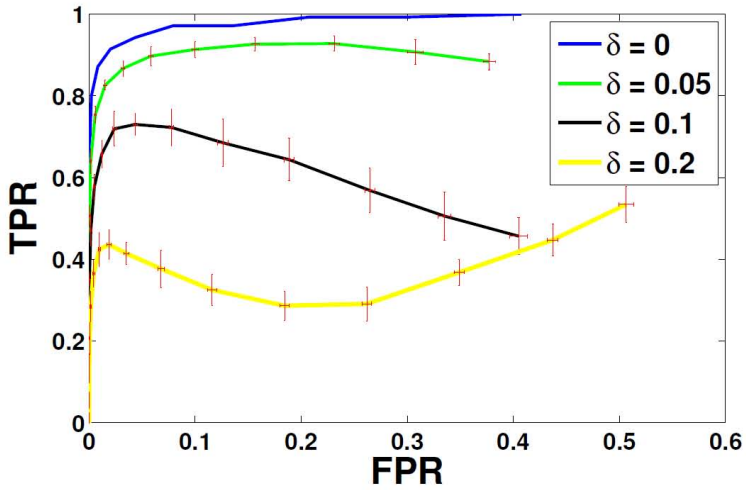
$$\|\hat{S}^{\rho} - \Sigma_0\|_{\infty} \leq O\left(\frac{1}{1-\delta}\sqrt{\frac{\log p}{n}}\right)$$

- Leads to bounds on $\left|\left|\left|\hat{\Theta} - \Theta_0\right|\right|\right|_2$, and other norms

# Experimental Results: Structure Recovery, No Projection




(a) Kendall no projection

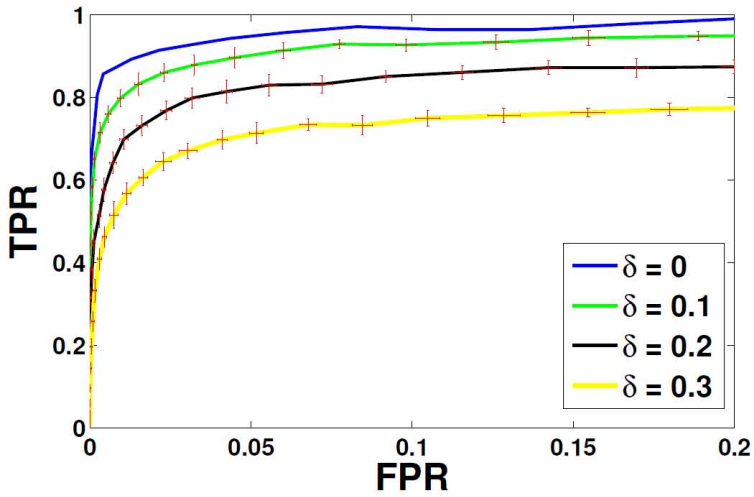



(b) Spearman no projection

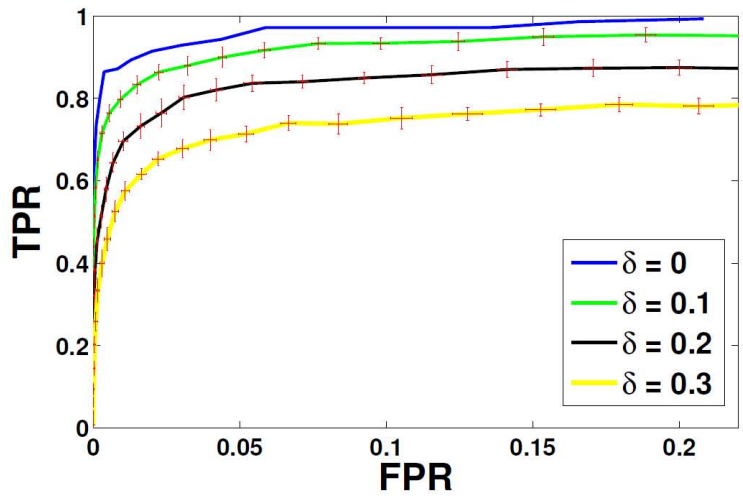
Recovery of true non-zero in $\Theta_0$ in $\hat{\Theta}$

- $\hat{S}$ need not be positive semi-definite
- $p = 100, n = 200$, changing $\lambda$
- TPR: True Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} \neq 0)$
- FPR: False Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} = 0)$

# Experimental Results: Structure Recovery, Projection




(c) Kendall, projection

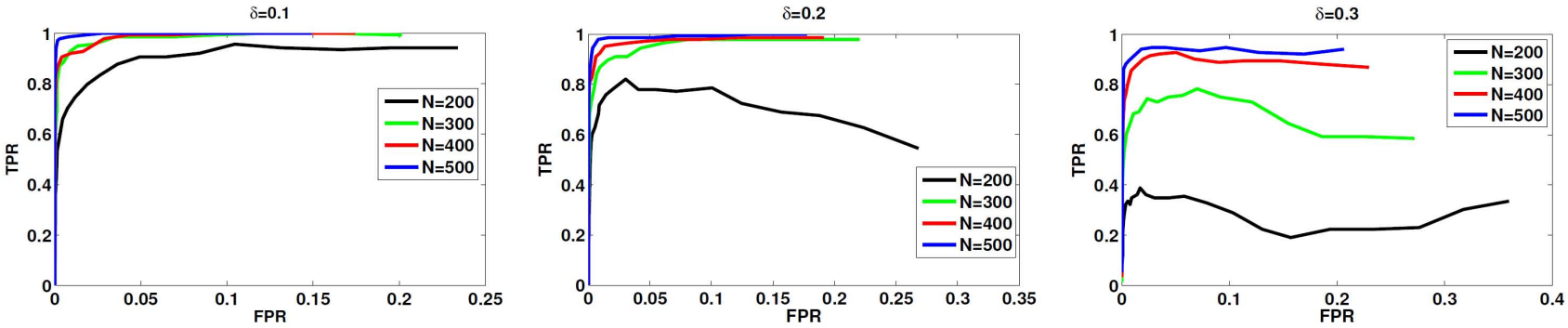



(d) Spearman, projection

Recovery of true non-zero in $\Theta_0$ in $\hat{\Theta}$

- $\hat{S}$ is projected to positive semi-definite cone, then estimate $\hat{\Theta}$
- $p = 100, n = 200$, changing $\lambda$
- TPR: True Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} \neq 0)$
- FPR: False Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} = 0)$

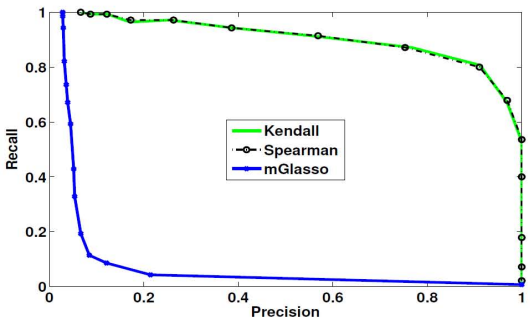
# Experimental Results: Increasing Samples



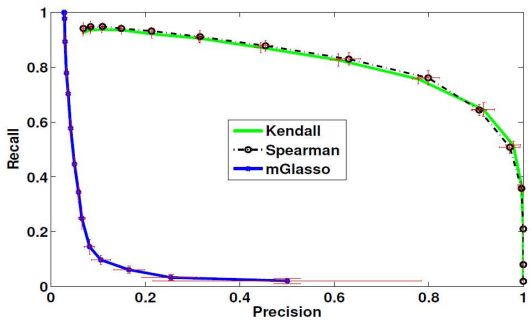

Recovery with increasing number of samples, different missing probability

- $\hat{S}$ need not be positive semi-definite
- $p = 100$, $n$ is increased, $\delta$ is increased, changing $\lambda$
- TPR: True Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} \neq 0)$
- FPR: False Positive Rate, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} = 0)$

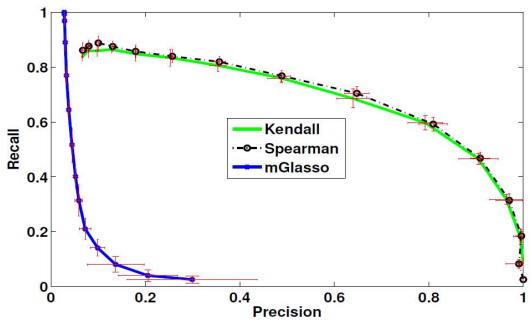
# Experimental Results: Gaussian vs Copula




(a) $\delta = 0$




(b) $\delta = 10\%$




(c) $\delta = 20\%$

Comparison of multivariate Gaussian vs. Copula recovery

- Precision: Accuracy of recovery, $P(\theta_{ij} \neq 0 | \hat{\theta}_{ij} \neq 0)$
- Recall: Coverage of recovery, $P(\hat{\theta}_{ij} \neq 0 | \theta_{ij} \neq 0)$

# Optimization for Precision Estimation

- For a given $\hat{S} \in \mathbb{R}^{p \times p}$

$$\min_{\Theta} \|\hat{\Theta}\|_1 \quad \text{s.t.} \quad \|\hat{S}\hat{\Theta} - \mathbb{I}\|_\infty \leq \lambda$$

- Non-smooth optimization problem
- Idea: "Lifting" and Linear Programming (LP)
  - Lifting: Add auxiliary variables, solve it in higher dimensions
  - LP: Parallel first order approach using "ADMM"

# Alternating Direction Method of Multipliers (ADMM)

- ADMM solves the following problem

$$\min_{\mathbf{x}\in\mathcal{X},\mathbf{z}\in\mathcal{Z}} \; f(\mathbf{x}) + g(\mathbf{z}) \quad \text{s.t.} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c}$$

  - $f, g$: (nonsmooth) convex functions, including linear and indicator functions
- Augmented Lagrangian, with dual variable $\mathbf{y}$ and $\rho > 0$

$$L_\rho(\mathbf{x}, \mathbf{z}, \mathbf{y}) = f(\mathbf{x}) + g(\mathbf{z}) + \langle \mathbf{y}, \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c} \rangle + \frac{\rho}{2} \|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}\|_2^2$$

- ADMM Algorithm [Gabay et al., 1976, Boyd et al., 2011]

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}} \; f(\mathbf{x}) + \langle \mathbf{y}_t, \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}_t - \mathbf{c} \rangle + \frac{\rho}{2} \|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}_t - \mathbf{c}\|^2 \;,$$

$$\mathbf{z}_{t+1} = \operatorname*{argmin}_{\mathbf{z}} \; g(\mathbf{z}) + \langle \mathbf{y}_t, \mathbf{A}\mathbf{x}_{t+1} + \mathbf{B}\mathbf{z} - \mathbf{c} \rangle + \frac{\rho}{2} \|\mathbf{A}\mathbf{x}_{t+1} + \mathbf{B}\mathbf{z} - \mathbf{c}\|^2$$

$$\mathbf{y}_{t+1} = \mathbf{y}_t + \rho(\mathbf{A}\mathbf{x}_{t+1} + \mathbf{B}\mathbf{z}_{t+1} - \mathbf{c}) \;.$$

# CLIME Estimator in ADMM form

- $\mathbf{X}$ is $k$ columns of $\hat{\Theta}$, $\mathbf{E}$ is the corresponding $k$ columns of $\mathbb{I}$
- Estimating precision matrix in terms of column block

$$\min_{\mathbf{X}} \|\mathbf{X}\|_1 \quad \text{s.t.} \quad \|\hat{\mathbf{S}}\mathbf{X} - \mathbf{E}\|_\infty \leq \lambda \ ,$$

$$\Downarrow$$

$$\text{ADMM form :} \quad \min_{\mathbf{X},\mathbf{Z}} \|\mathbf{X}\|_1 \quad \text{s.t.} \quad \|\mathbf{Z} - \mathbf{E}\|_\infty \leq \lambda, \hat{\mathbf{S}}\mathbf{X} = \mathbf{Z} \ .$$

- $\mathbf{X}$ is unconstrained, $\mathbf{Z}$ satisfies $\|\mathbf{Z} - \mathbf{E}\|_\infty \leq \lambda$

# CLIME-ADMM Updates

- Augmented Lagrangian

$$L_\rho = \|\mathbf{X}\|_1 + \rho\langle \mathbf{Y}, \hat{\mathbf{S}}\mathbf{X} - \mathbf{Z}\rangle + \frac{\rho}{2}\|\hat{\mathbf{S}}\mathbf{X} - \mathbf{Z}\|_2^2$$

- Inexact ADMM updates
  - By linearizing the $\mathbf{X}$ update

$$\text{Exact: } \mathbf{X}^{t+1} = \text{argmin}_{\mathbf{X}} \|\mathbf{X}\|_1 + \frac{\rho}{2}\|\hat{\mathbf{S}}\mathbf{X} - \mathbf{Z}^t + \mathbf{Y}^t\|_2^2$$

$$\text{Inexact: } \mathbf{X}^{t+1} = \text{argmin}_{\mathbf{X}} \|\mathbf{X}\|_1 + \eta\langle \mathbf{V}^t, \mathbf{X}\rangle + \frac{\eta}{2}\|\mathbf{X} - \mathbf{X}^t\|_2^2 \,,$$

$$\mathbf{Z}^{t+1} = \underset{\|\mathbf{Z}-\mathbf{E}\|_\infty \le \lambda}{\text{argmin}} \frac{\rho}{2}\|\hat{\mathbf{S}}\mathbf{X}^{t+1} - \mathbf{Z} + \mathbf{Y}^t\|_2^2 \,,$$

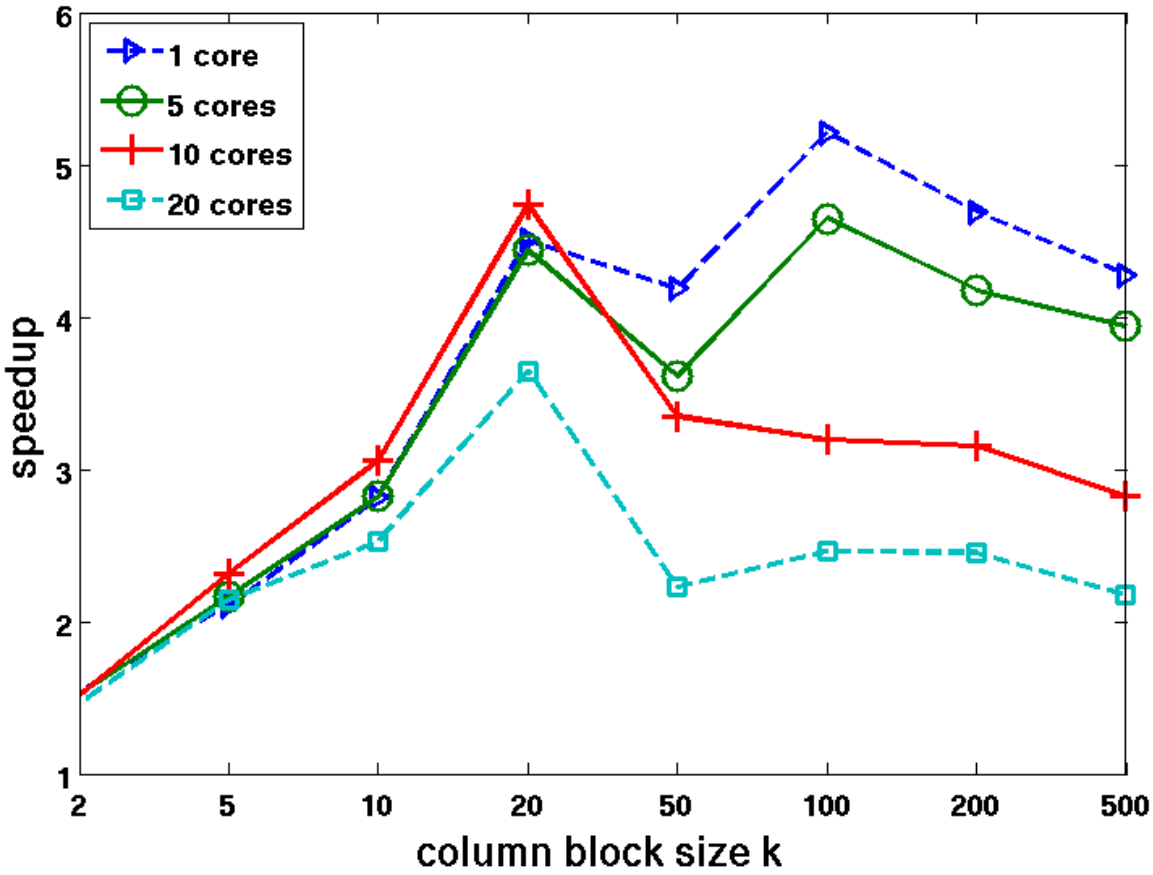
$$\mathbf{Y}^{t+1} = \mathbf{Y}^t + \hat{\mathbf{S}}\mathbf{X}^{t+1} - \mathbf{Z}^{t+1} \,.$$

where $\mathbf{V}^t = \frac{\rho}{\eta}\hat{\mathbf{S}}(\mathbf{Y}^t + \hat{\mathbf{S}}\mathbf{X}^t - \mathbf{Z}^t)$.

# CLIME-ADMM Algorithm

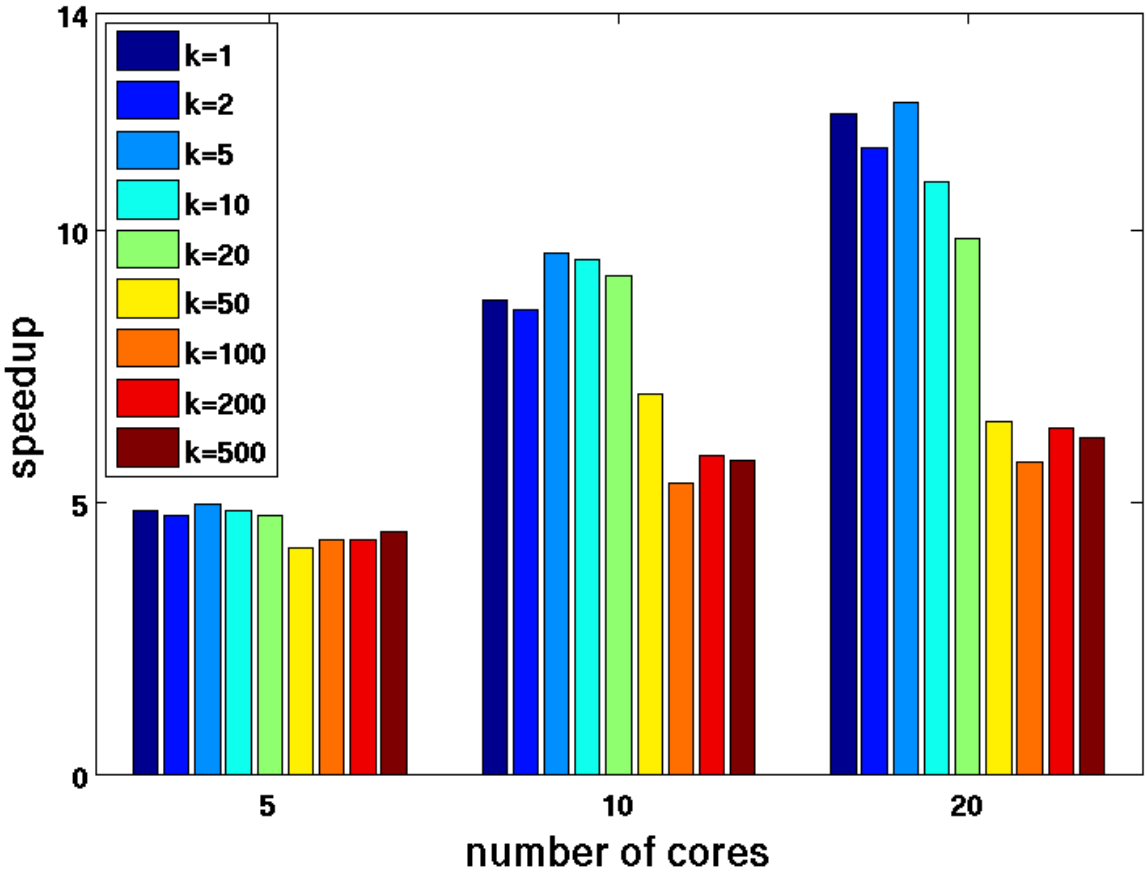
1: Input: $\hat{\mathbf{S}}, \lambda, \rho, \eta$
2: Output: $\mathbf{X}$
3: Initialization: $\mathbf{X}^0, \mathbf{Z}^0, \mathbf{Y}^0, \mathbf{V}^0, \hat{\mathbf{V}}^0 = 0$
4: **for** $t = 0$ to $T - 1$ **do**
5: **X-update:** $\mathbf{X}^{t+1} = \mathit{soft}(\mathbf{X}^t - \mathbf{V}^t, \frac{1}{\eta})$,
6: **Mat-Mul:** $\begin{cases} \text{sparse}: & \mathbf{U}^{t+1} = \hat{\mathbf{S}}\mathbf{X}^{t+1} \\ \text{low rank}: & \mathbf{U}^{t+1} = \mathbf{A}(\mathbf{A}'\mathbf{X}^{t+1}) \end{cases}$
7: **Z-update:** $\mathbf{Z}^{t+1} = \mathit{box}(\mathbf{U}^{t+1} + \mathbf{Y}^t, \lambda)$,
8: **Y-update:** $\mathbf{Y}^{t+1} = \mathbf{Y}^t + \mathbf{U}^{t+1} - \mathbf{Z}^{t+1}$
9: **Mat-Mul:** $\begin{cases} \text{sparse}: & \hat{\mathbf{V}}^{t+1} = \hat{\mathbf{S}}\mathbf{Y}^{t+1} \\ \text{low rank}: & \hat{\mathbf{V}}^{t+1} = \mathbf{A}(\mathbf{A}'\mathbf{Y}^{t+1}) \end{cases}$
10: **V-update:** $\mathbf{V}^{t+1} = \frac{\rho}{\eta}(2\hat{\mathbf{V}}^{t+1} - \hat{\mathbf{V}}^t)$
11: **end for**

# Experimental Results: Shared Memory



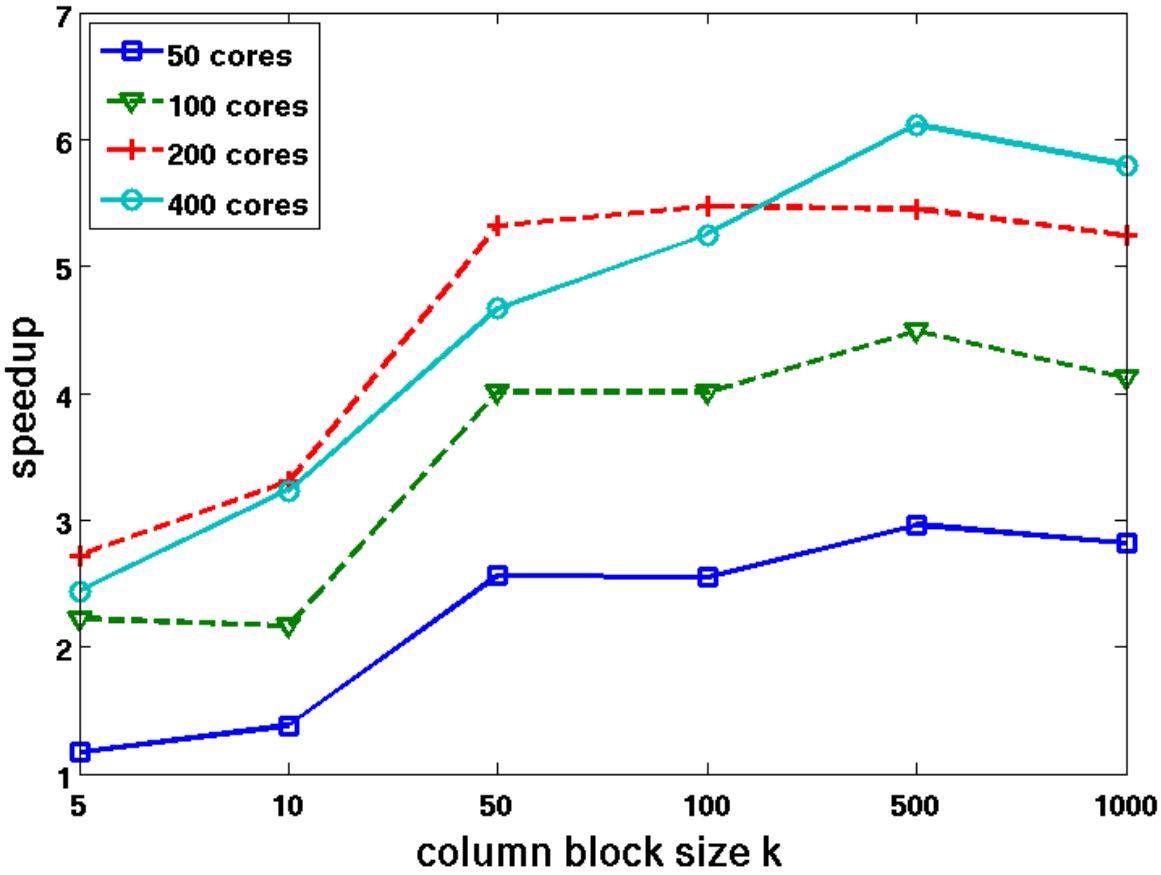

Speed-up with increasing block sizes, different number of cores

# Experimental Results: Shared Memory



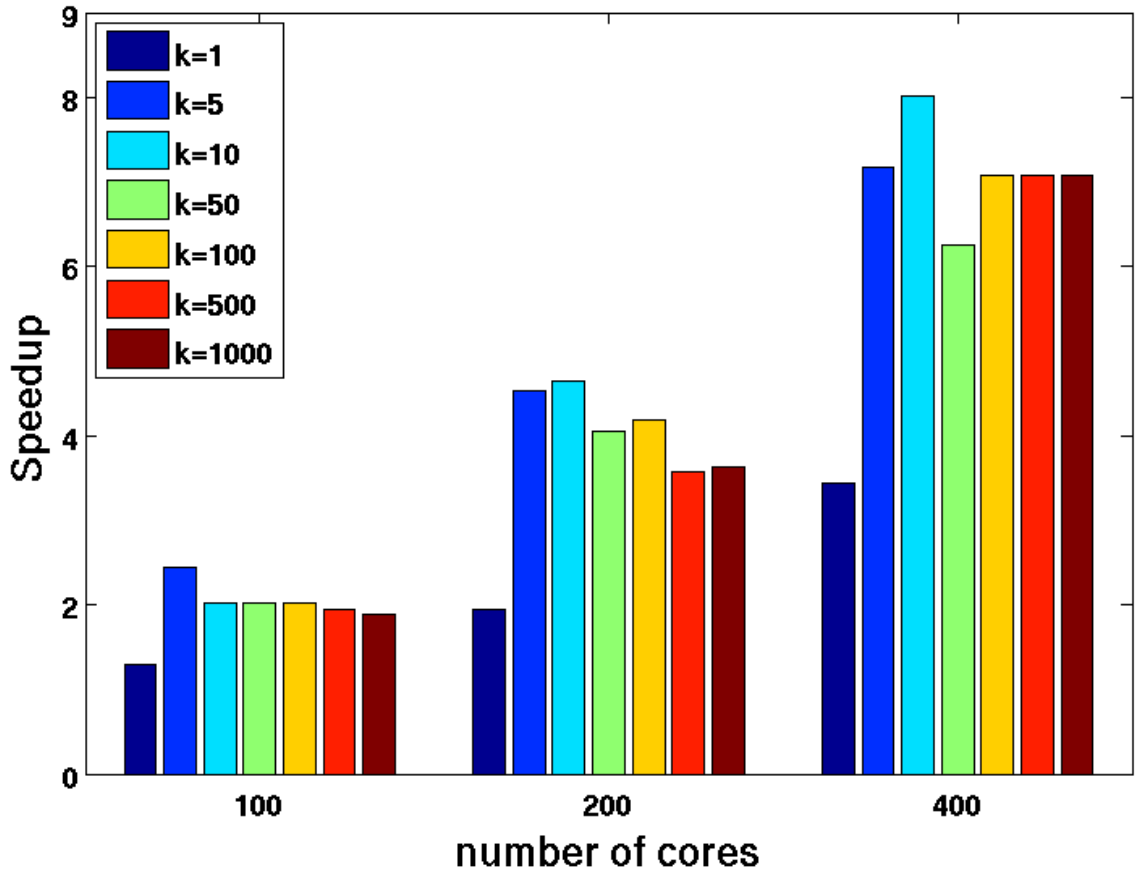

Speed-up with increasing cores, different block sizes

# Experimental Results: Distributed Memory




Speed-up with increasing block sizes, different number of cores

# Experimental Results: Distributed Memory




Speed-up with increasing cores, different block sizes

# Experimental Results: Million dimensions

| node ×core | k = 1 | k = 5 | k = 10 | k = 50 | k = 100 | k = 500 | k = 1000 |
|---|---|---|---|---|---|---|---|
| 100×1 | 0.56 | 1.26 | 2.59 | 6.98 | 13.97 | 62.35 | 136.96 |
| 25× 4 | 1.02 | 2.40 | 3.42 | 8.25 | 16.44 | 84.08 | 180.89 |
| 200×1 | 0.37 | 0.68 | 1.12 | 3.48 | 6.76 | 33.95 | 70.59 |
| 50×4 | 0.74 | 1.44 | 2.33 | 4.49 | 8.33 | 48.20 | 103.87 |

- Runtime (secs) with different cores per node
- Dimension $p = 10^6$, total 1 trillion ($p^2 = 10^{12}$) entries
- Using one core/node is the most efficient
- No resource sharing with other cores

# Conclusions

- Structure learning in graphical models
  - Conditional independence structure
  - Gaussians, Copula: Sparse precision matrix estimation
- Double plug-in estimator for Gaussian copula
  - Rank correlation estimates with missing values
  - Use Kendall-Kruskal formula to get correlation
  - Precision estimation using CLIME
  - Finite sample estimation error bounds
- Efficient optimization using parallel inexact ADMM
  - Lifted linear programming
  - Scales well, results on 1 million dimensions
- Structure learning for other multi-variate models

# References


- H. Wang, F. Fazayeli, S. Chatterjee, and A. Banerjee, Gaussian Copula Precision Estimation with Missing Values, AISTATS, 2014.
- H. Wang, A. Banerjee, C. Hsieh, P. Ravikumar, and I. Dhillon, Large Scale Distributed Sparse Precision Estimation, NIPS, 2013.
- H. Wang and A. Banerjee, Bregman Alternating Direction Method of Multipliers, arXiv, 2013.
- H. Wang and A. Banerjee, Online Alternating Direction Method, ICML, 2012.



**Acknowledgements:** NSF grants IIS-0953274, IIS-0916750, IIS-1029711, NASA grant NNX12AQ39A; University of Minnesota Supercomputing Institute.