Gradient-based Sparse Principal Component Analysis

with Applications to Gene Co-expression Analysis

Yixuan Qiu

School of Statistics and Management

Shanghai University of Finance and Economics

Joint work with Dr. Kathryn Roeder, Jing Lei, and Jiebiao Wang

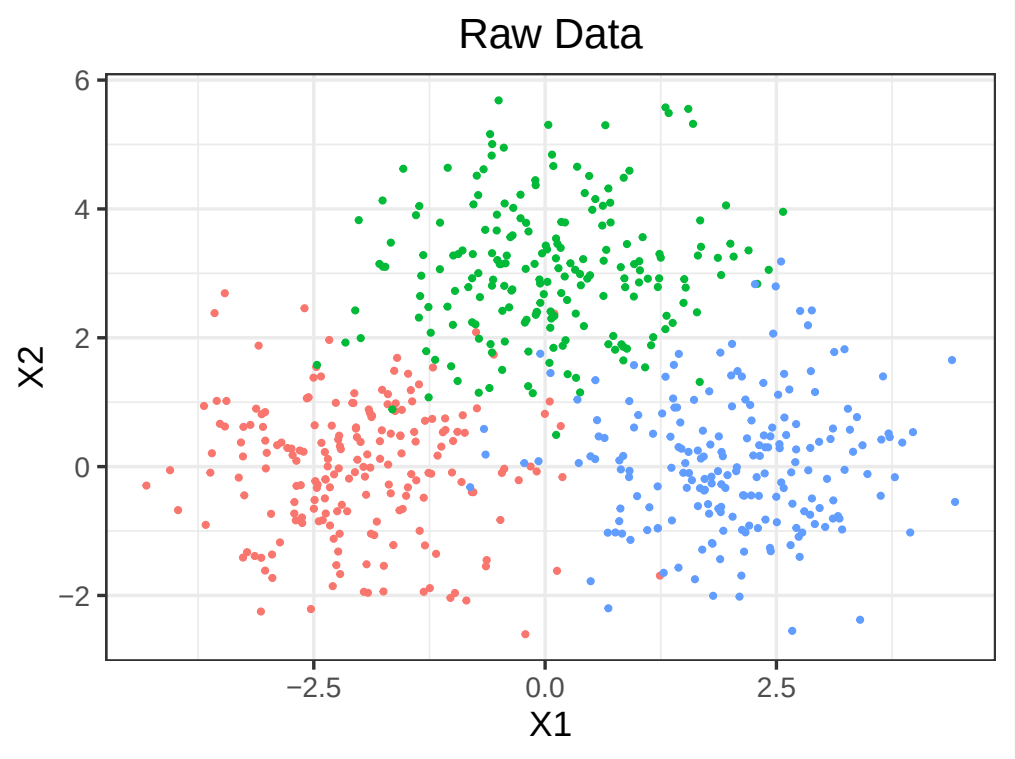
BioStat Seminar @ CityU Department of Biostatistics 香港城市大學 City University of Hong Kong

Outline

Motivation

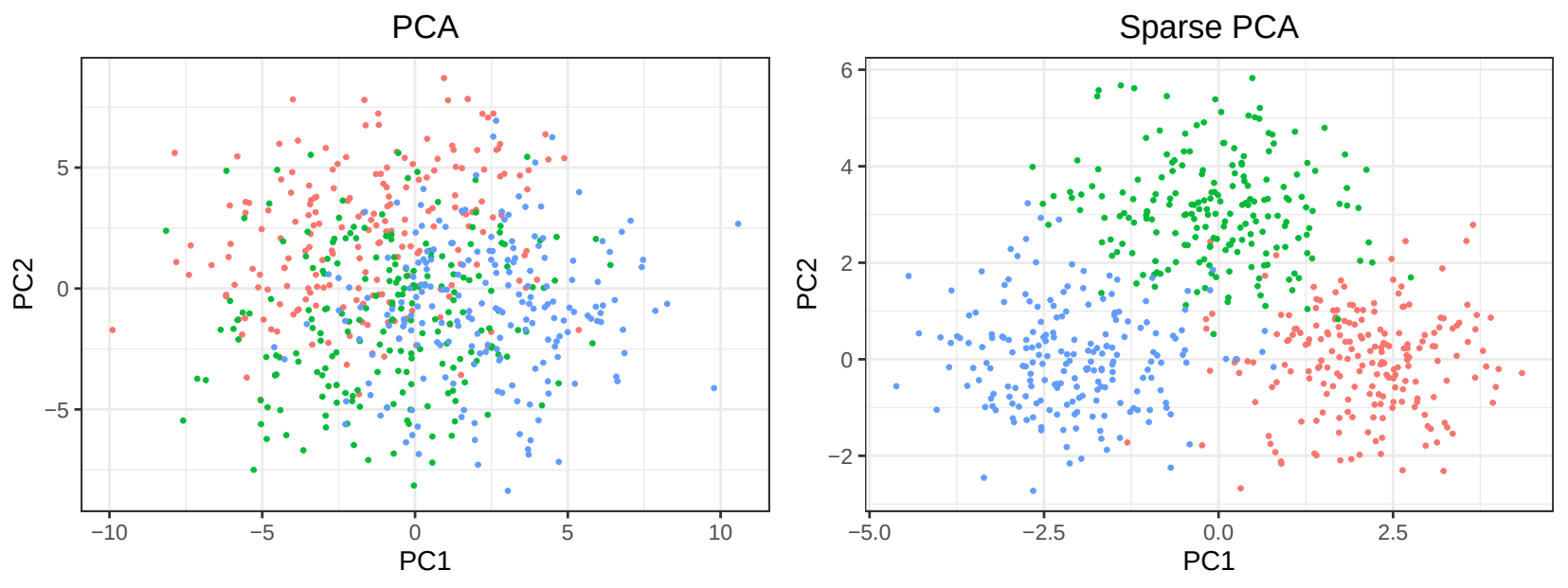
An Illustrative Example

- Simulate a data matrix with $n = 600$ and $p = 3000$
- $\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim \frac{1}{3} N\left(\begin{pmatrix} -2 \\ 0 \end{pmatrix}, I_2\right) + \frac{1}{3} N\left(\begin{pmatrix} 0 \\ 3 \end{pmatrix}, I_2\right) + \frac{1}{3} N\left(\begin{pmatrix} 2 \\ 0 \end{pmatrix}, I_2\right)$
- $X_3, \ldots, X_{3000} \sim N(0, 1)$, independent of (X_1, X_2)
- Clearly there are three clusters
- Imagine three cell types with two marker genes



PCA vs Sparse PCA

- Dimension reduced to 2
- Visualize PC1 vs PC2
- Left: conventional PCA
- Right: sparse PCA



Overview of Sparse PCA

- Sparse PCA = PCA + Sparsity
- Factor loadings are sparse

Overview of Sparse PCA

- Sparse PCA = PCA + Sparsity
- Factor loadings are sparse

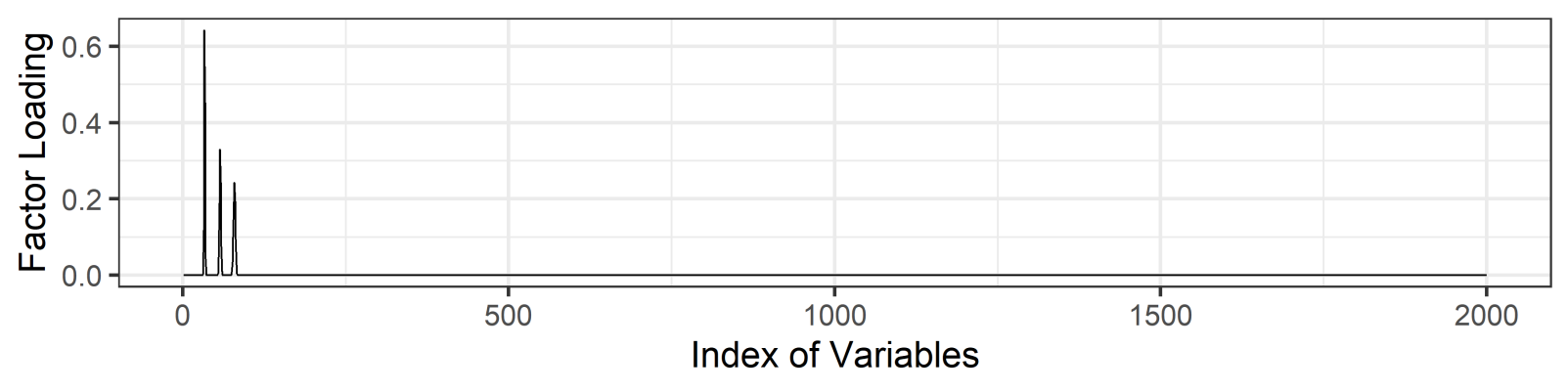
- Why sparse?
- PCA may be inconsistent in high dimensions (Johnstone and Lu, 2009; Jung and Marron, 2009)
  - Sparsity $\Rightarrow$ Denoising
- Each principal component only depends on a small number of variables
  - Sparsity $\Rightarrow$ Better interpretation

Example in Johnstone and Lu (2009)

- Model studied in Johnstone and Lu (2009)

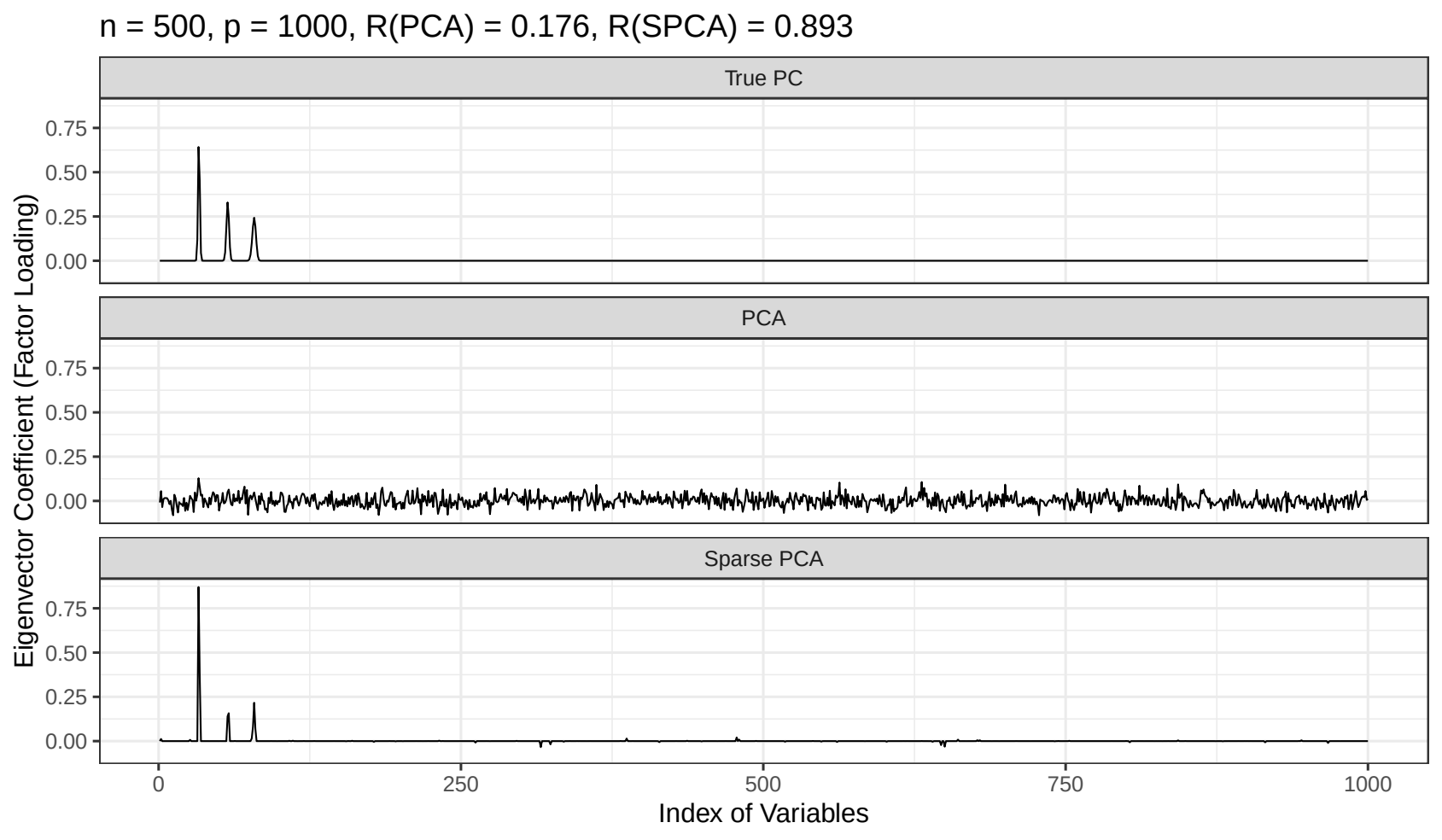
$$\vec{X} = v\vec{\rho} + \sigma\vec{\varepsilon}, \quad v \sim N(0,1), \varepsilon \sim N(0, I_p), v \perp \varepsilon$$

- $\mathrm{Cov}(X) = \rho\rho^{\mathrm{T}} + \sigma^2 I_p$, so $\rho/\|\rho\|$ is the leading eigenvector
- Assume that the true ρ vector is sparse:

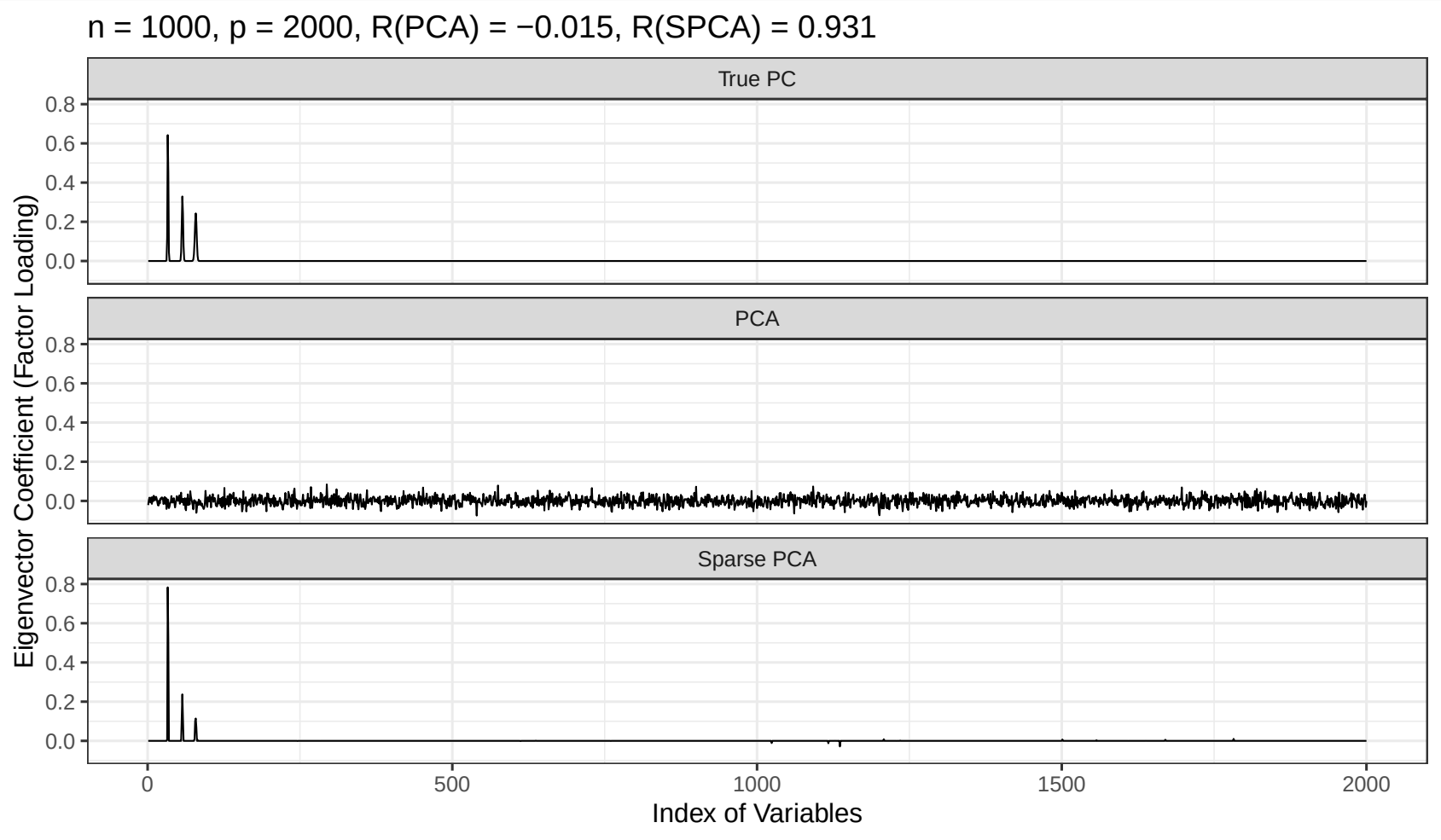


- Collect sample $X_1, \ldots, X_n$ and fix $p/n = 2$ and $\sigma = (n/p)^{0.25}$
- For an estimator $\hat{\rho}$, compute $R(\hat{\rho}, \rho) = \cos\angle(\hat{\rho}, \rho)$
  - $|R| = 1$, perfect estimate; $|R| = 0$, no information at all

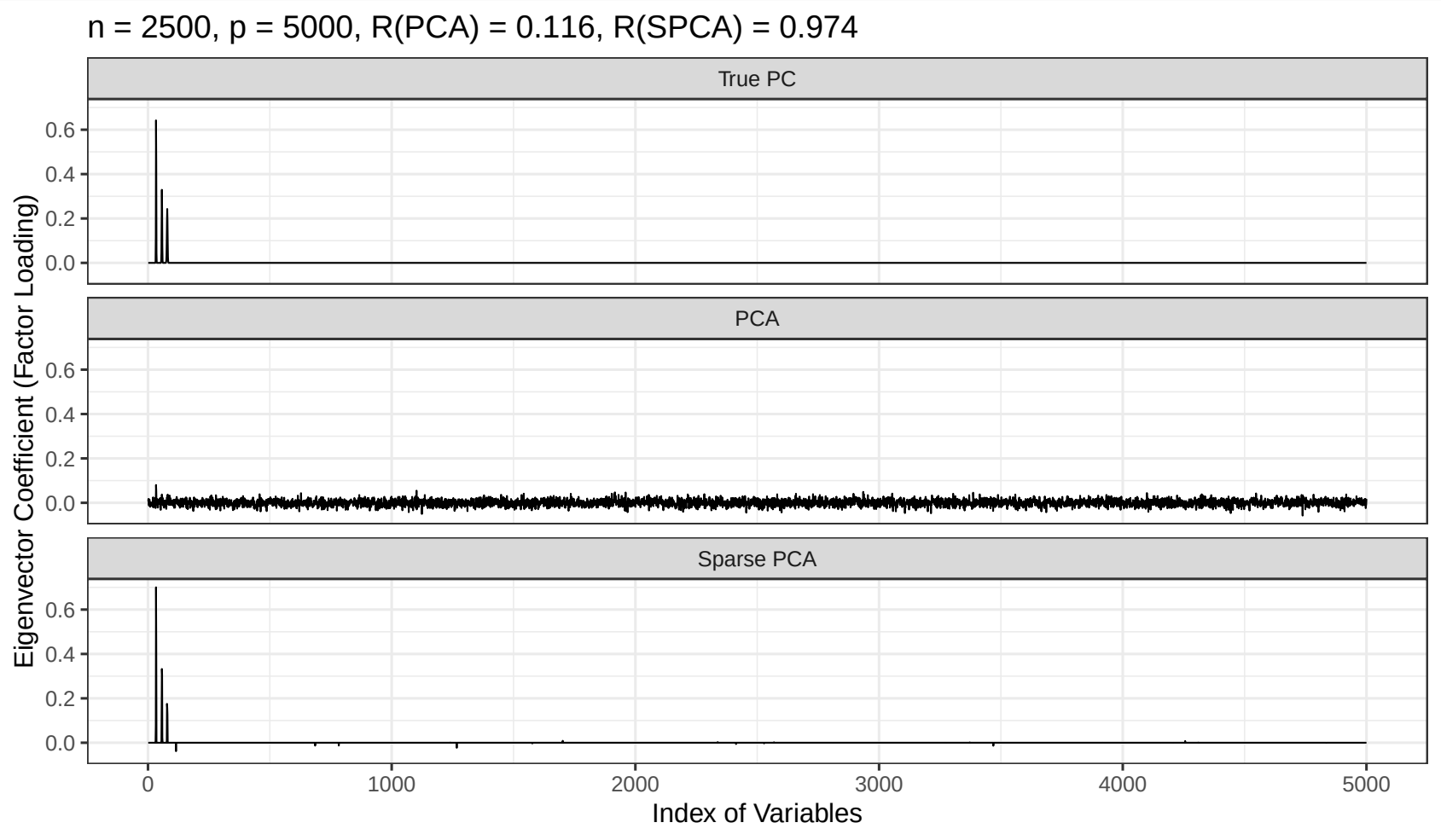
PCA in High Dimensions ($p = 1000$)



PCA in High Dimensions ($p = 2000$)



PCA in High Dimensions ($p = 5000$)



Sparse PCA Formulations

- Many different formulations
- Nonconvex objective functions
  - The lasso approach in PCA (Jolliffe, Trendafilov, and Uddin, 2003)
  - Regression-based (Zou, Hastie, and Tibshirani, 2006)
  - Penalized matrix decomposition (Witten, Tibshirani, and Hastie, 2009)
  - Generalized power method (Journée et al., 2010)
  - Iterative thresholding method (Shen and Huang, 2008; Ma, 2013)
  - ...
- Convex objective functions
  - DSPCA (d'Aspremont et al., 2005)
  - Fantope projection and selection (Vu et al., 2013)

Computation of Sparse PCA

- Nonconvex methods
  - Fast
  - Little global convergence guarantee
  - Heavily relies on model assumptions and initial values
- Convex methods
  - Global convergence
  - Weak assumptions
  - Slow

Gradient-based Sparse PCA Algorithm

Model Setting

- Convex formulation proposed by Vu et al. (2013)

$$\begin{aligned} \min_{X} \quad & -\operatorname{tr}(SX) + \lambda \|X\|_1 \\ \text{s.t.} \quad & O \preceq X \preceq I \text{ and } \operatorname{tr}(X) = d \end{aligned}$$

- $\Gamma_{p\times d}$: factor loading matrix (eigenvectors, our target)
- $X_{p\times p}$: estimator of the projection matrix $\Pi = \Gamma\Gamma^{\mathrm{T}}$ (almost Γ)
- $S_{p\times p}$: sample covariance matrix (data)
- λ: sparsity parameter
- d: number of components

Intuition

- Traditional PCA

$$\max_{\Gamma} \quad \mathrm{tr}(\Gamma^{\mathrm{T}} S \Gamma) \qquad \text{(maximum explained variance)}$$
$$\text{s.t.} \quad \Gamma^{\mathrm{T}}\Gamma = I_d \qquad \text{(orthogonality)}$$

- Adding nonconvex sparsity term

$$\max_{\Gamma} \quad \mathrm{tr}(\Gamma^{\mathrm{T}} S \Gamma) - \lambda d \|\Gamma\|_{2,0}^2 \qquad \text{(number of nonzero rows)}$$
$$\text{s.t.} \quad \Gamma^{\mathrm{T}}\Gamma = I_d$$

- Convex formulation, $X = \Gamma\Gamma^{\mathrm{T}}$

$$\max_{X} \quad \mathrm{tr}(SX) - \lambda \|X\|_1 \qquad \text{(approximation to } \|\Gamma\|_{2,0}^2\text{)}$$
$$\text{s.t.} \quad O \preceq X \preceq I \text{ and } \mathrm{tr}(X) = d \qquad \text{(convex version of } \Gamma^{\mathrm{T}}\Gamma = I_d\text{)}$$

  - $\mathrm{tr}(\Gamma^{\mathrm{T}} S \Gamma) = \mathrm{tr}(S\Gamma\Gamma^{\mathrm{T}}) = \mathrm{tr}(SX)$: explained variance

Existing Computation Method

- ADMM algorithm

$$
\begin{aligned}
X_{k+1} &= \mathcal{P}_{\mathcal{F}^d}(Y_k - U_k + \alpha S) \\
Y_{k+1} &= \mathcal{S}_{\alpha\lambda}(X_{k+1} + U_k) \\
U_{k+1} &= U_k + X_{k+1} - Y_{k+1}
\end{aligned}
$$

- $\mathcal{S}_{\alpha\lambda}$: soft-thresholding operator, easy
- $\mathcal{P}_{\mathcal{F}^d}$: projection operator onto $\mathcal{F}^d = \{X : O \preceq X \preceq I \text{ and } \mathrm{tr}(X) = d\}$, hard
  - Requires a full eigen decomposition in each iteration
  - $\mathcal{O}(p^3)$ complexity

Timing Comparison

Unit: milliseconds

expr		min	mean	median	max
Full	[1000]	150.631973	155.367721	151.872986	171.347982
Largest	[1000]	1.314212	1.686147	1.766314	1.895186
Smallest	[1000]	4.746720	5.035787	4.977878	5.373219
Full	[2000]	1146.316032	1239.926216	1169.956945	1605.542461
Largest	[2000]	7.502122	8.635942	7.897849	12.450570
Smallest	[2000]	13.257879	13.783933	13.676452	14.535811
Full	[5000]	17278.650632	17677.653736	17705.457132	18283.440595
Largest	[5000]	51.513812	57.926554	53.511937	80.093321
Smallest	[5000]	51.155627	54.081903	52.349482	64.919859

Optimization on Intersection of Convex Sets

- Let $f(X) = -\mathrm{tr}(SX) + \lambda \|X\|_1$, then the solution is

$$X_* = \arg\min_{X \in \mathcal{F}^d} f(X)$$

- A constrained problem on the intersection of convex sets $\mathcal{F}^d = C_1 \cap G_1 \cap G_2$, where
- $C_1 = \{X : \mathrm{tr}(X) = d\}$
- $G_1 = \{X : g_1(X) \leq 0\}$, $g_1(X) = \theta_{\max}(X) - 1$
- $G_2 = \{X : g_2(X) \leq 0\}$, $g_2(X) = -\theta_{\min}(X)$

A Nearly Projection-free Algorithm

- Let $\mathcal{L}(X) = f(X) + \mu\left(d_{C_1}(X) + r_1[g_1(X)]_+ + r_2[g_2(X)]_+\right)$
- $d_{C_1}(X)$: distance between X and C_1
- $[x]_+ = \max\{x, 0\}$
- An unconstrained problem $\min_{X \in \mathcal{X}} \mathcal{L}(X)$
- Projection onto $\mathcal{X} = \{X : \|X\|_F \leq \sqrt{d}\}$ is trivial

Theorem
If $\mu \geq (\sqrt{2}+1)(\|S\|_F + \lambda p + 1)\sqrt{p/(d+1)}$, $r_1 = \sqrt{d(d+1)}$, $r_2 = \sqrt{p(d+1)}$, *then* $\min_{X \in \mathcal{F}^d} f(X) = \min_{X \in \mathcal{X}} \mathcal{L}(X)$.

General Form

- Many statistical models need to solve a complicated constrained optimization problem

$$\min_{x \in \mathcal{K} \subset \mathcal{X}} f(x), \quad \mathcal{K} = C_1 \cap \cdots \cap C_l \cap G_1 \cap \cdots \cap G_m$$

- Projection onto C_i is easy
- $G_i = \{x : g_i(x) \leq 0\}$, and $g_i(x)$ is easy to compute
- For some constants μ and ρ_i, and some function $h(\cdot)$, we can construct a new function

$$\begin{aligned} &\mathcal{L}(x; \mu) \\ =& f(x) + \mu h\left(d_{C_1}(x), \ldots, d_{C_l}(x), \rho_1^{-1}[g_1(x)]_+, \ldots, \rho_m^{-1}[g_m(x)]_+\right) \end{aligned}$$

- Under some mild conditions,

$$\min_{x \in \mathcal{K}} f(x) = \min_{x \in \mathcal{X}} \mathcal{L}(x; \mu)$$

Convergence Analysis

- Many different algorithms to solve $\min_{X \in \mathcal{X}} \mathcal{L}(X)$
  - Subgradient descent
  - Proximal-proximal gradient method (Ryu and Yin, 2019)
- For the proximal-proximal gradient method, after T iterations,

$$\mathcal{L}(\hat{X}) \leq \min_{X \in \mathcal{X}} \mathcal{L}(X) + \frac{C}{T} \quad \text{and} \quad d_{\mathcal{F}^d}(\hat{X}) \leq \frac{C}{T},$$

where C is some constant.

Statistical Property

- Assumptions
  - Sparsity: the factor loading matrix has at most s nonzero rows
  - Identifiability: the d-th eigengap $\delta_d = \theta_d(\Sigma) - \theta_{d+1}(\Sigma) > 0$
  - Sub-exponential elements:
    $\max_{i,j} P(|S_{ij} - \Sigma_{ij}| \geq u) \leq 2\exp(-4nu^2/\sigma^2)$ for all $u \leq \sigma$

Theorem
Take $\lambda = \sigma\sqrt{\log(p)/n}$, and then with probability at least $1 - 2/p^2$,

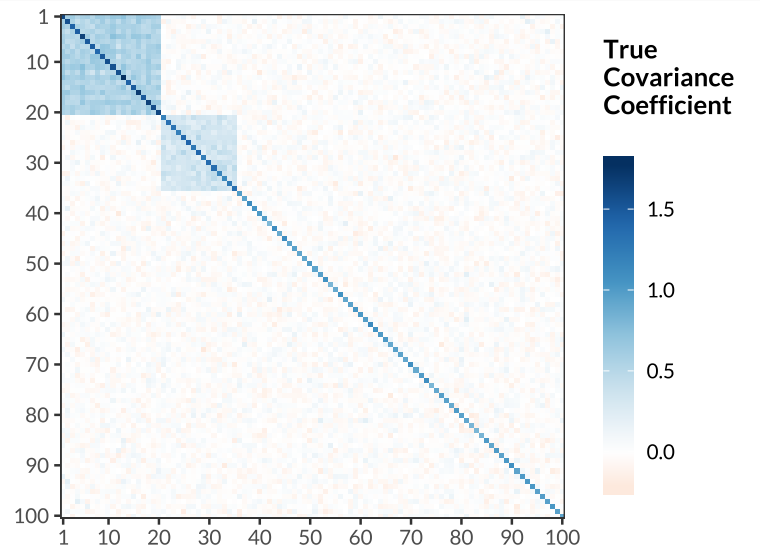
$$\|\hat{X} - \Pi\|_F \leq \frac{4\sigma s\sqrt{\log(p)}}{\delta_d\sqrt{n}} + \frac{\sqrt{2C/\delta_d}}{\sqrt{T}} + \frac{C}{T}.$$

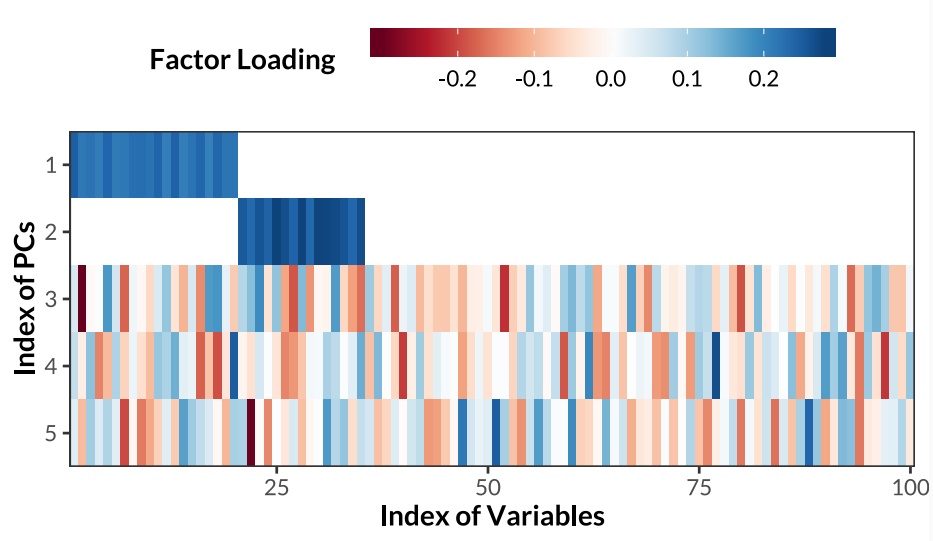
- Interpretation: statistical error + optimization error + feasibility error

Numerical Experiment

Computational Efficiency

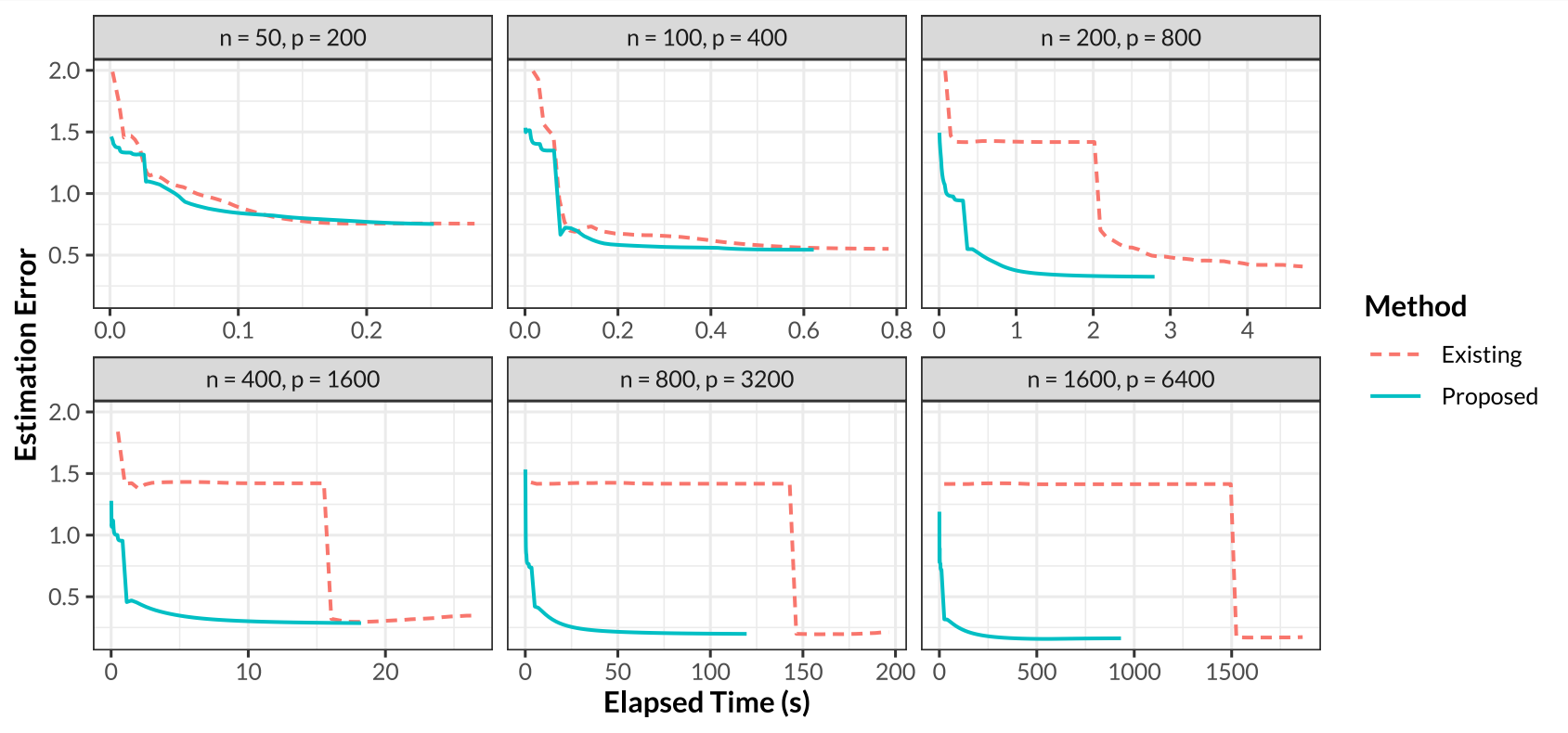
- Model setting





Computational Efficiency

- Comparing with the existing ADMM algorithm



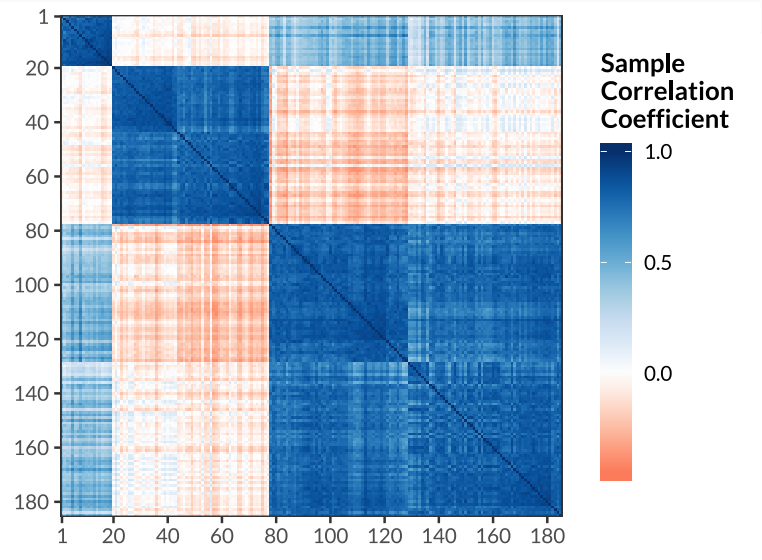
Applications to Gene Co-expression Analysis

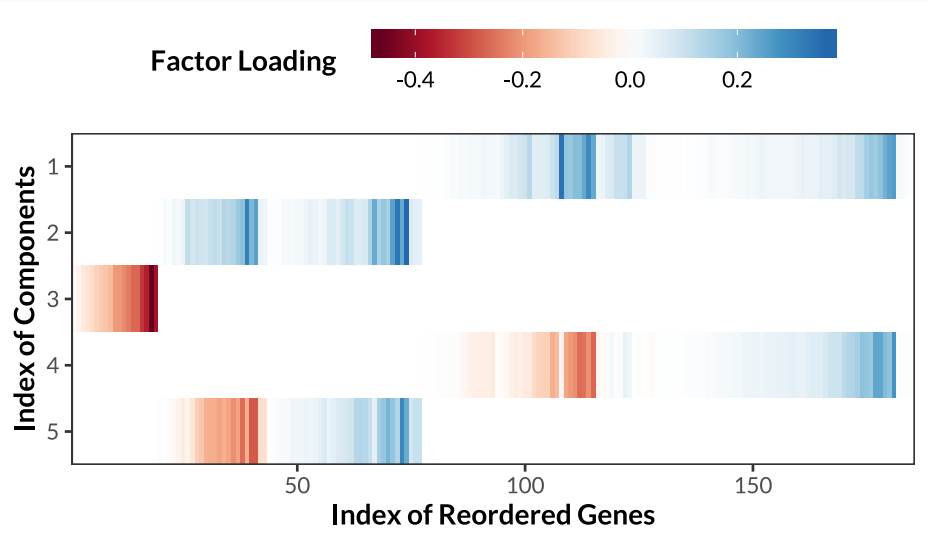
Application I - Gene Co-expression Network

- Brain gene expression data collected by the CommonMind Consortium (Fromer et al., 2016)
- To detect groups of genes such that genes in the same group have high mutual correlations
- $p = 16,423$ genes from 258 schizophrenia subjects and 279 control subjects
- Compute $d = 5$ sparse principal components
- Cluster selected genes into $k = 5$ groups based on the factor loadings

Gene Co-expression Network

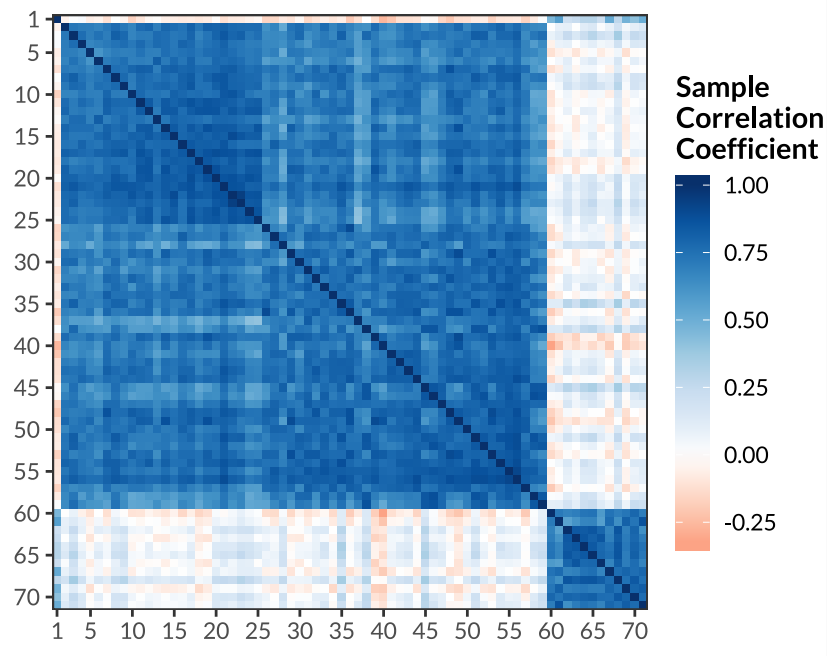
- Clustering result for the schizophrenia group

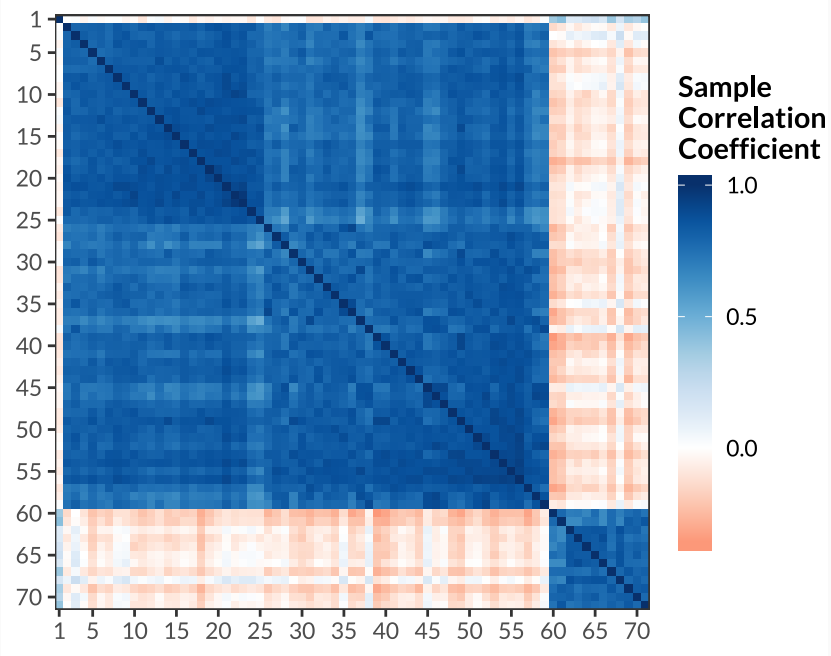




Gene Co-expression Network

- Differential analysis, control vs schizophrenia



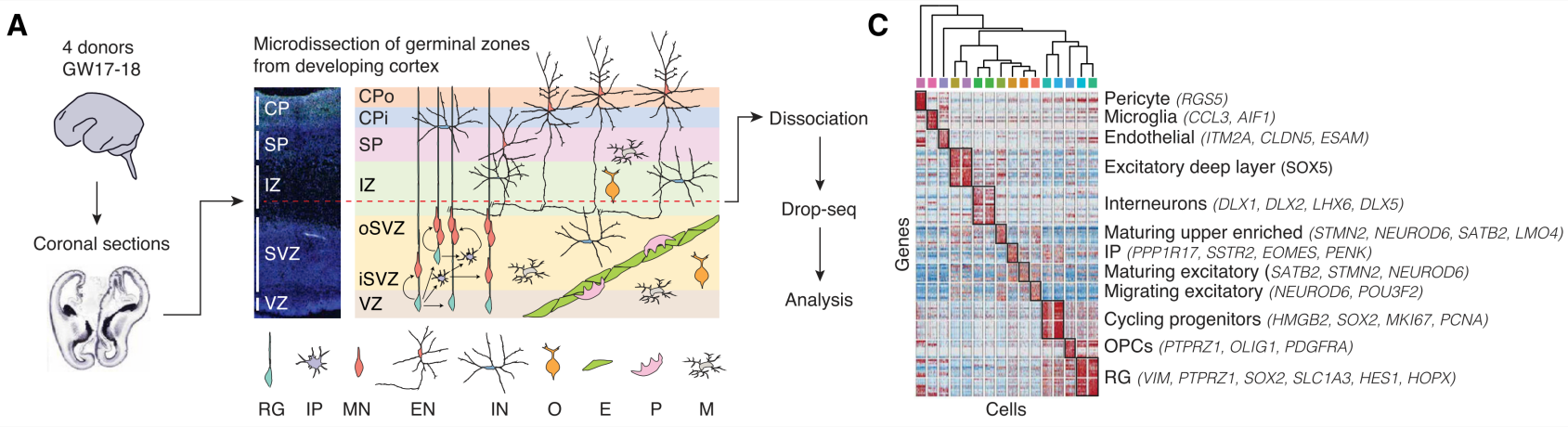


Application II - Identify Cell-type-specific Marker Genes

- Cell-type-specific genes, also known as marker genes
- Highly expressed in one cell type, but lowly expressed in other types
- Help to annotate cell clusters and study cellular composition of bulk tissues
- Key to the analysis of RNA transcriptional data

Application II - Identify Cell-type-specific Marker Genes

- Typically identified using single-cell RNA sequencing data
- Challenges
  - Data availability and cost
  - Data quality and noise



(Figure from Polioudakis et al., 2019, Neuron.)

The Proposed Approach

- Develop semi-supervised statistical technique to identify marker genes from bulk transcriptome (high quality and low cost)
- Input 1: The existing gene list treated as "prior knowledge"
- Input 2: A bulk RNA sequencing data set
- Output: Refined marker gene list

The Proposed Approach

- Develop semi-supervised statistical technique to identify marker genes from bulk transcriptome (high quality and low cost)
- Input 1: The existing gene list treated as "prior knowledge"
- Input 2: A bulk RNA sequencing data set
- Output: Refined marker gene list

- Why this is possible?

The Proposed Approach

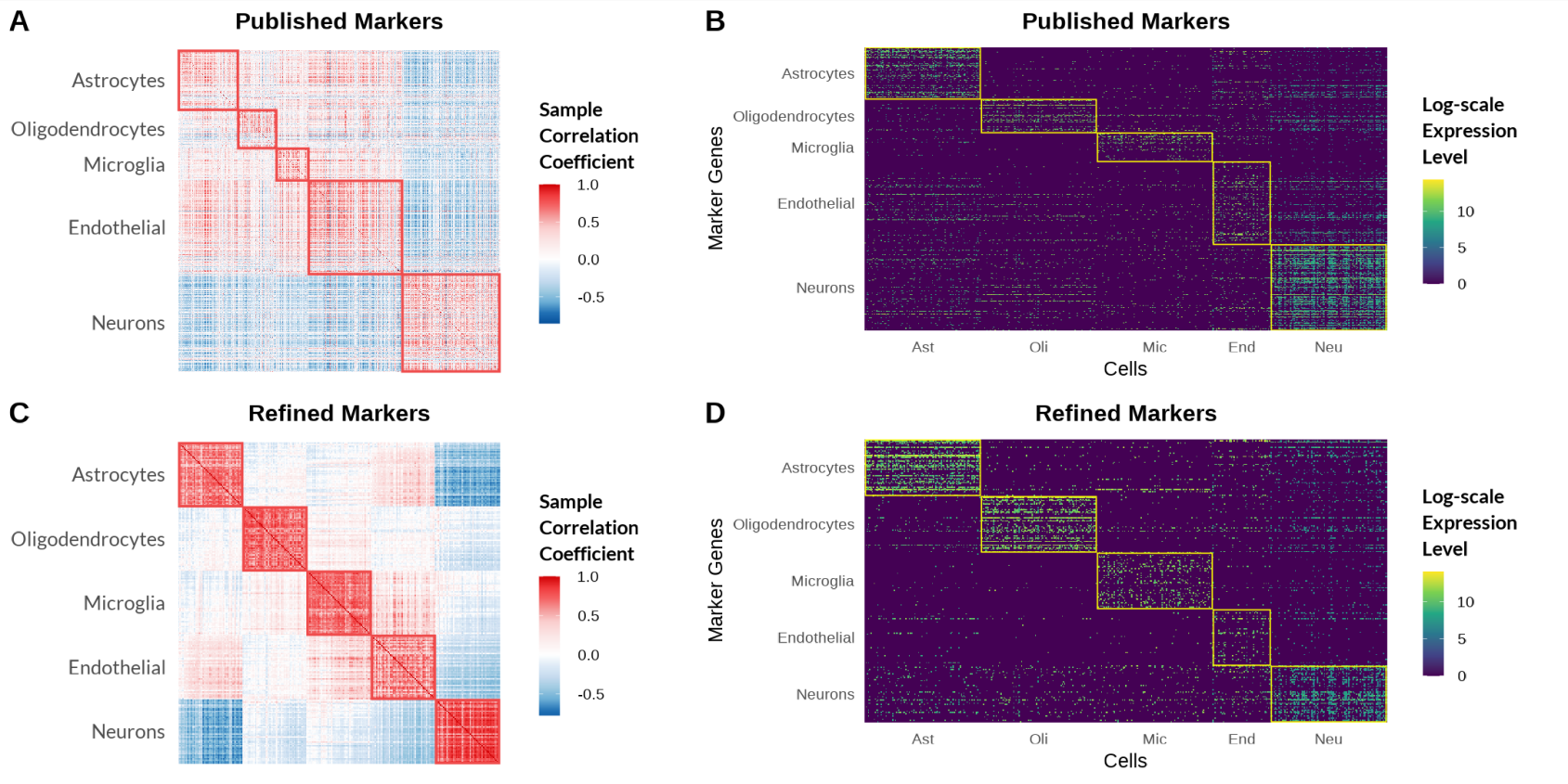
- Develop semi-supervised statistical technique to identify marker genes from bulk transcriptome (high quality and low cost)
- Input 1: The existing gene list treated as "prior knowledge"
- Input 2: A bulk RNA sequencing data set
- Output: Refined marker gene list

- Why this is possible?

- Marker genes are highly correlated in the bulk data!

The Proposed Approach

- Why refinement?
  - The gene list is typically obtained from external data sets, or even from different species
  - There exist errors and noise



Modified Sparse PCA

- The proposed algorithm, called MarkerPen, solves a modified sparse PCA problem:

$$\begin{aligned} \max_{X} \quad & \mathrm{tr}(SX) - \lambda p_{G,w}(X) \\ \text{s.t.} \quad & O \preceq X \preceq I, \ \mathrm{tr}(X) = 1, \ X \geq 0 \end{aligned}$$

- $p_{G,w}(X) = \sum_{i,j} \tilde{p}_{G,w}(X_{ij})$ is a penalty function with

$$\tilde{p}_{G,w}(X_{ij}) = \begin{cases} |X_{ij}|, & i,j \in G \\ w^2|X_{ij}|, & i \notin G, j \notin G \\ w|X_{ij}|, & \text{otherwise} \end{cases}$$

- G is the prior gene list for some cell type C, and $w \geq 1$ a hyperparameter

Modified Sparse PCA

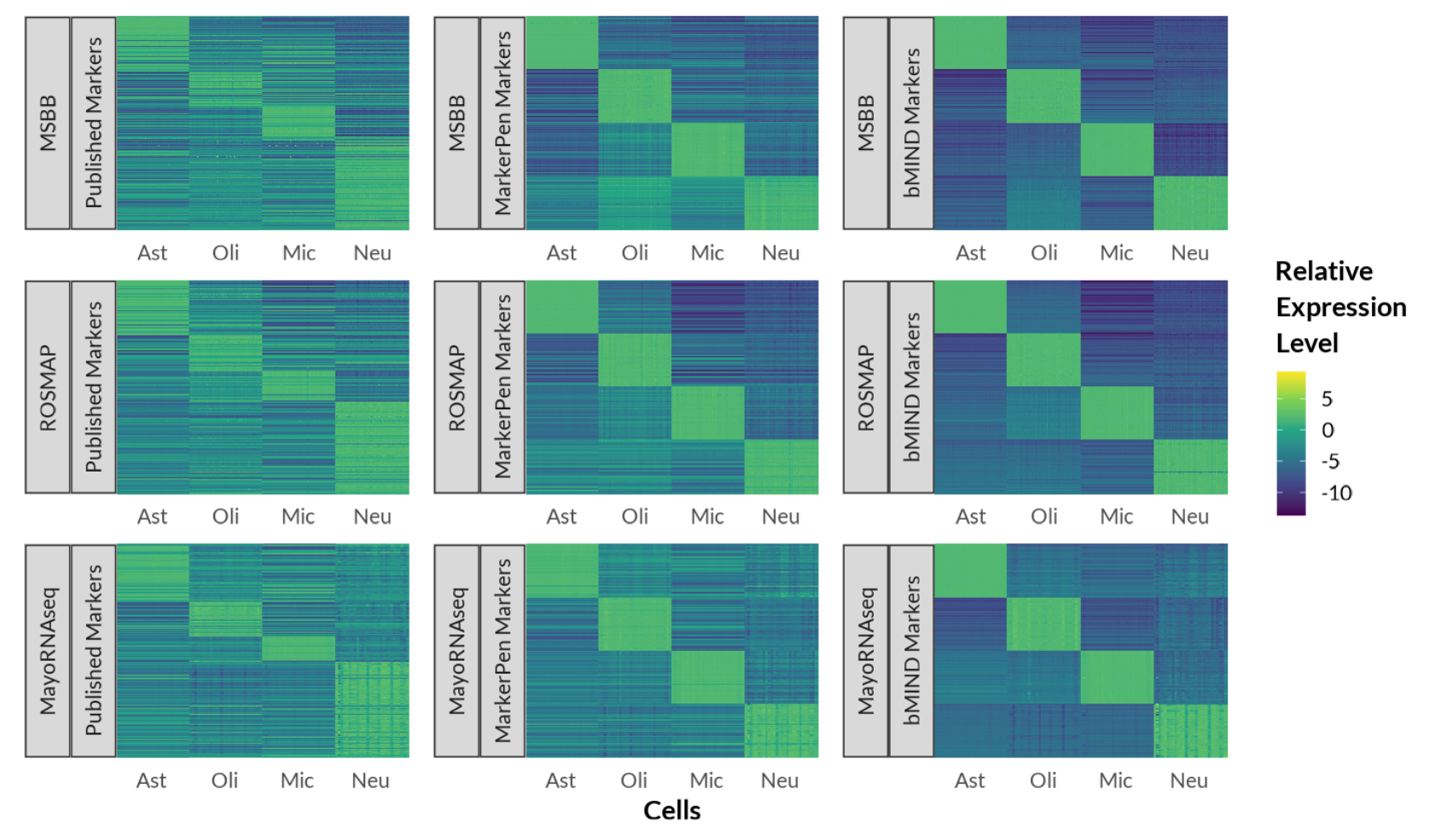
- Intuition 1: We want to find genes that are highly positively correlated
  - Correlation matrix with high positive mutual correlations has a leading eigenvector γ of positive $\gamma\gamma^{\mathrm{T}}$ coefficients

Modified Sparse PCA

- Intuition 1: We want to find genes that are highly positively correlated
  - Correlation matrix with high positive mutual correlations has a leading eigenvector γ of positive $\gamma\gamma^{\mathrm{T}}$ coefficients
- Intuition 2: Coefficients for genes outside the given list G are more likely to receive larger sparsity penalty
  - $p_{G,w}(X)$ controls which genes are more likely to be retained

More Results

- Published, MarkerPen-refined, and "ground truth" marker genes



Summary

Summary

- Many statistical models are limited by their computational difficulty on large-scale data sets
- The convex sparse PCA and its extensions are such examples
- The challange largely comes from the optimization problem
- We develop a general technique to transform highly constrained optimization problems to nearly unconstrained ones
- The new algorithm has visible advantages on computational performance
- Enables reproducible statistical analyses on high-dimensional genetic data

References

Qiu, Y., Wang, J., Lei, J., and Roeder, K. (2021). Identification of Cell-type-specific Marker Genes from Co-expression Patterns in Tissue Samples. *Bioinformatics*.

Qiu, Y., Lei, J., and Roeder, K. (2022+). Gradient-based Sparse Principal Component Analysis with Extensions to Online Learning. *Biometrika*.

R packages available at `https://statr.me/research/`

THANK YOU!