Noisy Bilinear Low-Rank Matrix Sketching

Wenbin Wang, Xindi Ping, Cheng Cheng, Ziping Zhao

ShanghaiTech University, Shanghai, China
2025 IEEE Information Theory Workshop

October 2nd, 2025

Compressed Sensing

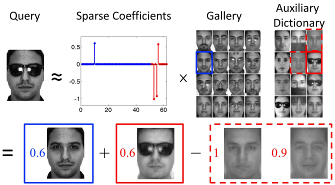
Classic Framework:

$$y_i = \boldsymbol{a}_i^\top \boldsymbol{x}, \quad i = 1, \ldots, m,$$

where $\{\boldsymbol{a}_i\}_{i=1}^m$ are the sketching/sensing vectors, $\boldsymbol{x}$ is the data/signal, $\{y_i\}_{i=1}^m$ are the measurements.

(a) Magnetic Resonance Imaging

(b) Image Denoising



(c) Robust Face Recognition

Challenges in Modern Data Acquisition

Data generation at unprecedented rate: data samples are

- high-dimensional (dimension $\gg$ date number);
- not observable due to privacy or security constraints;
- distributed at multiple locations.

Limited processing power at sensor platforms:

- **time-sensitive:** impossible to obtain a complete snapshot of the system;
- **storage-limited:** cannot store the whole data set;
- **power-hungry:** minimize the number of observations.

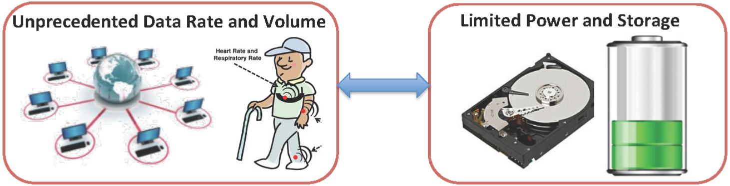


Figure 1: Mismatch streaming[1]

[1] https://yuejiechi.github.io/GeometricConstraints.html

Bilinear Matrix Sketching

$$\underbrace{\boldsymbol{Y}}_{\text{observation}} = \underbrace{\boldsymbol{A}}_{\text{measurement matrix}} \underbrace{\boldsymbol{X}}_{\text{unknown}} \underbrace{\boldsymbol{B}^\top}_{\text{measurement matrix}} + \underbrace{\boldsymbol{E}}_{\text{noise}}.$$

$$\boldsymbol{Y}, \boldsymbol{E} \in \boldsymbol{R}^{m\times m}, \quad \boldsymbol{X} \in \boldsymbol{R}^{d\times d}, \quad \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times d}, \quad m \ll d.$$

Bilinear Matrix Sketching

$$\underbrace{\boldsymbol{Y}}_{\text{observation}} = \underbrace{\boldsymbol{A}}_{\text{measurement matrix}} \underbrace{\boldsymbol{X}}_{\text{unknown}} \underbrace{\boldsymbol{B}^\top}_{\text{measurement matrix}} + \underbrace{\boldsymbol{E}}_{\text{noise}}.$$

$$\boldsymbol{Y}, \boldsymbol{E} \in \boldsymbol{R}^{m\times m}, \quad \boldsymbol{X} \in \boldsymbol{R}^{d\times d}, \quad \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times d}, \quad m \ll d.$$

Why Bilinear? Why Matrix? Why low-rank?

Bilinear Matrix Sketching

$$\underbrace{\boldsymbol{Y}}_{\text{observation}} = \underbrace{\boldsymbol{A}}_{\text{measurement matrix}} \underbrace{\boldsymbol{X}}_{\text{unknown}} \underbrace{\boldsymbol{B}^\top}_{\text{measurement matrix}} + \underbrace{\boldsymbol{E}}_{\text{noise}}.$$

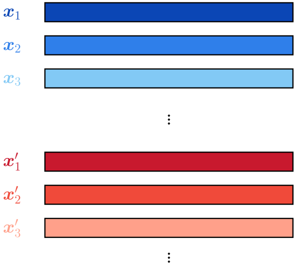
$$\boldsymbol{Y}, \boldsymbol{E} \in \boldsymbol{R}^{m\times m}, \quad \boldsymbol{X} \in \boldsymbol{R}^{d\times d}, \quad \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times d}, \quad m \ll d.$$

Why Bilinear? Why Matrix? Why low-rank?

Application-driven

Covariance Sketching

Consider two data $\boldsymbol{x}, \boldsymbol{x}'$ possibly distributedly observed at m sensors:



Bilinear Sketching:

- two sketching matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m \times d}$ with specific distribution;
- two observations $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{z}' = \boldsymbol{B}\boldsymbol{x}'$ with the cross-covariance matrix of the sketches:

$$\mathbb{E}(\boldsymbol{z}\boldsymbol{z}'^{\top}) = \boldsymbol{A} \underbrace{\boldsymbol{X}^{\star}}_{\mathbb{E}(\boldsymbol{x}\boldsymbol{x}'^{\top})} \boldsymbol{B}$$

Graph Sketching

Consider a directed graph $\mathcal{G}$ with d nodes with adjacency matrix $\boldsymbol{X}$.

- First, we consider $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}\boldsymbol{A}^\top$
- Define $\boldsymbol{A} \in \mathbb{R}^{m \times d}$ as composed of i.i.d. Bernoulli entries such as

$$\boldsymbol{A}_{u,i} = \begin{cases} 1, & \text{if } \ i \in u, \\ 0, & \text{otherwise.} \end{cases}$$

- Then,

$$\boldsymbol{Y}_{u,v} = \sum_{i=0}^{d-1} \sum_{j=0}^{d-1} \boldsymbol{A}_{u,i}\, \boldsymbol{X}_{i,j}\, \boldsymbol{A}_{v,j}$$

Illustration of Graph Sketching

$$\boldsymbol{X} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

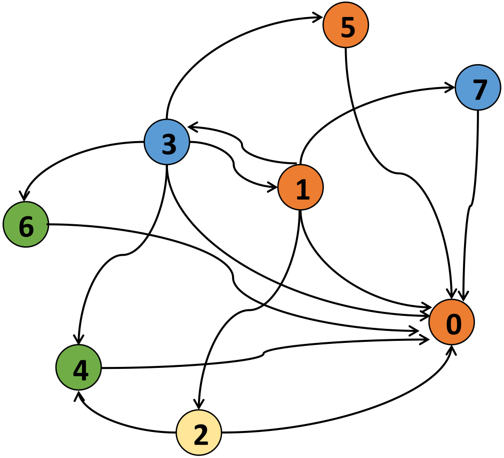


Figure 2: Original Graph: $\boldsymbol{X}$

Illustration of Graph Sketching

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

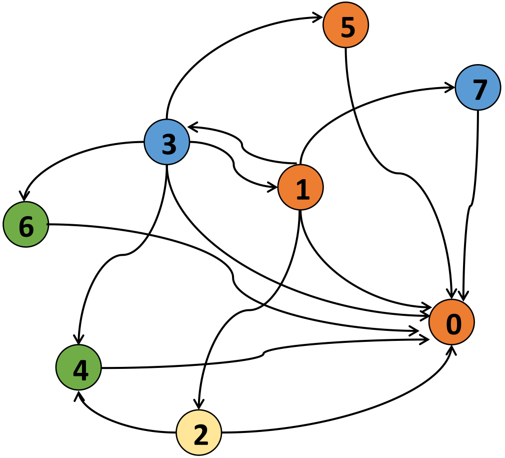


Figure 3: Original Graph: X

Illustration of Graph Sketching

$$\boldsymbol{Y} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 4 & 2 \\ 1 & 2 & 2 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix}$$

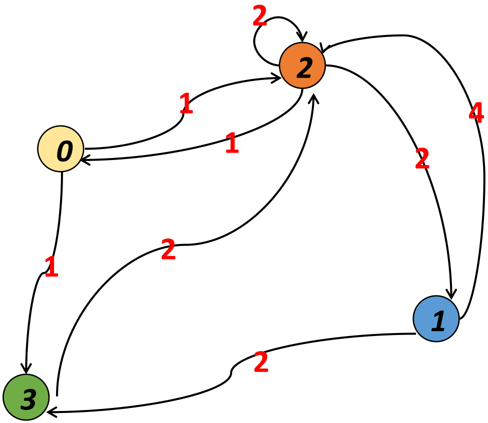


Figure 4: Compressed Graph: $\boldsymbol{Y}$

Graph Sketching

Consider a directed graph $\mathcal{G}$ with d nodes with adjacency matrix $\boldsymbol{X}$.

- Now, we consider $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}\boldsymbol{B}^\top$
- Define $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m \times d}$ as composed of i.i.d. Bernoulli entries such as

$$\boldsymbol{A}_{u,i} = \begin{cases} 1, & \text{if } i \in u, \\ 0, & \text{otherwise.} \end{cases} \qquad \boldsymbol{B}_{v,j} = \begin{cases} 1, & \text{if } j \in v, \\ 0, & \text{otherwise.} \end{cases}$$

- Then,

$$\boldsymbol{Y}_{u,v} = \sum_{i=0}^{d-1} \sum_{j=0}^{d-1} \boldsymbol{A}_{u,i} \boldsymbol{X}_{i,j} \boldsymbol{B}_{v,j} = \sum_{i \in u} \sum_{j \in v} \boldsymbol{X}_{i,j}$$

The sketching matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively partition the original graph $\mathcal{G}$ in two different dimensions.

Illustration of Graph Sketching

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

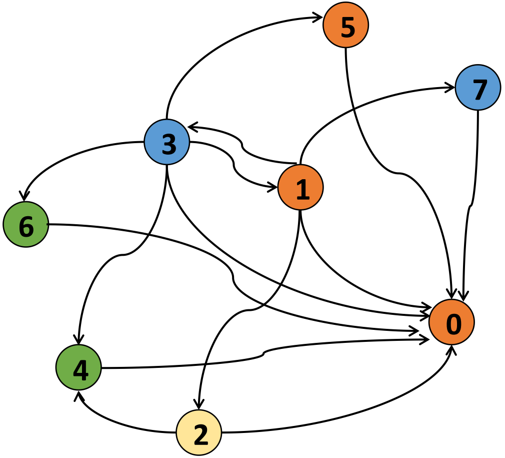


Figure 5: Original Graph: $\boldsymbol{Y}$

Illustration of Graph Sketching

$$X = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

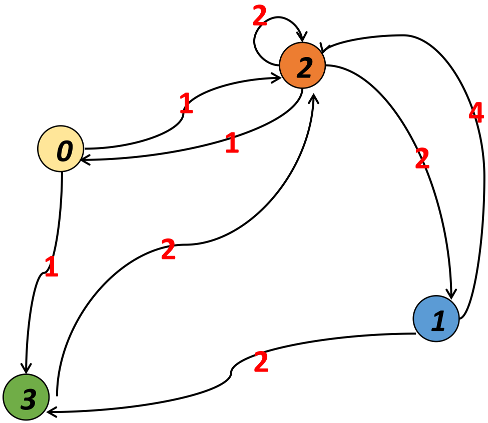


Figure 6: Compressed Graph: $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}\boldsymbol{A}^\top$

Illustration of Graph Sketching

$$\boldsymbol{X} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{B} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

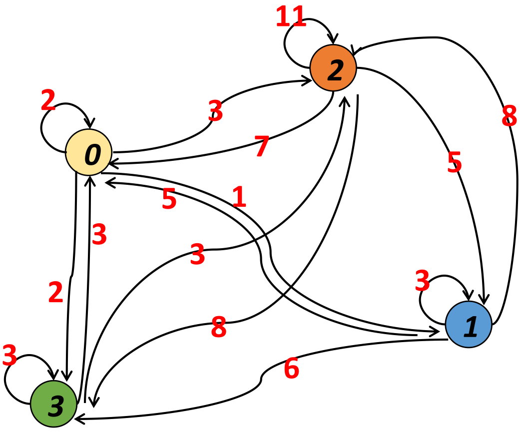


Figure 7: Compressed Graph: $\boldsymbol{Y} = \boldsymbol{B}\boldsymbol{X}\boldsymbol{B}^\top$

Illustration of Graph Sketching

$$AXA^\top \quad \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 4 & 2 \\ 1 & 2 & 2 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix}$$

$$BXB^\top \quad \begin{bmatrix} 2 & 1 & 3 & 2 \\ 5 & 3 & 8 & 6 \\ 7 & 5 & 11 & 8 \\ 3 & 0 & 3 & 3 \end{bmatrix}$$

$$AXB^\top \quad \begin{bmatrix} 1 & 1 & 2 & 1 \\ 3 & 2 & 5 & 4 \\ 3 & 2 & 4 & 3 \\ 2 & 0 & 2 & 2 \end{bmatrix}$$

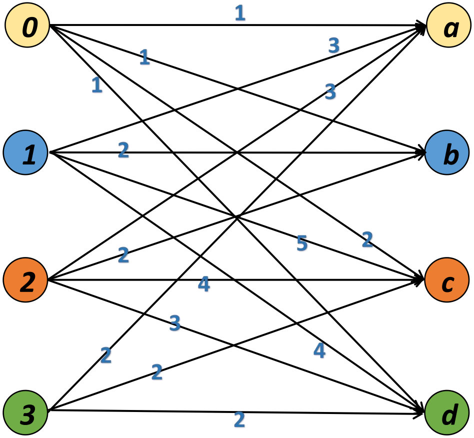


Figure 8: Compressed Graph: $Y = AXB^\top$

Related Work

The *distributed sparsity* assumption: the matrix $\boldsymbol{X}^\star$ is called d-distributed sparse if each row/column of $\boldsymbol{X}$ cannot have more than d non-zeros.

- Convex optimization program (Dasarathy 2012)

$$\widehat{\boldsymbol{X}} = \arg\min_{\boldsymbol{X}} \left\{ \left\| \boldsymbol{A}\boldsymbol{X}\boldsymbol{B}^\top - \boldsymbol{Y} \right\|_{\mathrm{F}}^2 + \lambda \left\| \boldsymbol{X} \right\|_1 \right\}$$

  ☺ is *easy* to obtain by iterative algorithm due to its convexity,

  ☹ introduces a non-negligible bias.

- **Our work:**

  ☺ no bias

  ☹ low-rank

Problem Formulation

We propose to estimate the low-rank matrices from bilinear measurements using the non-convex penalty

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \frac{1}{2m^2}\left\|\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{X}\boldsymbol{B}^\top\right\|_{\mathrm{F}}^2+P_\lambda(\boldsymbol{X})$$

where $P_\lambda(\boldsymbol{X})=\sum_{i=1}^d p_\lambda(\sigma_i(\boldsymbol{X}))$ is a decomposable nonconvex penalty imposed on the singular values of $\boldsymbol{X}$ such as $P_\lambda(\boldsymbol{X})=\lambda\|\boldsymbol{X}\|_*+\sum_{i=1}^d q_\lambda(\sigma_i(\boldsymbol{X}))$.

Assumption 1

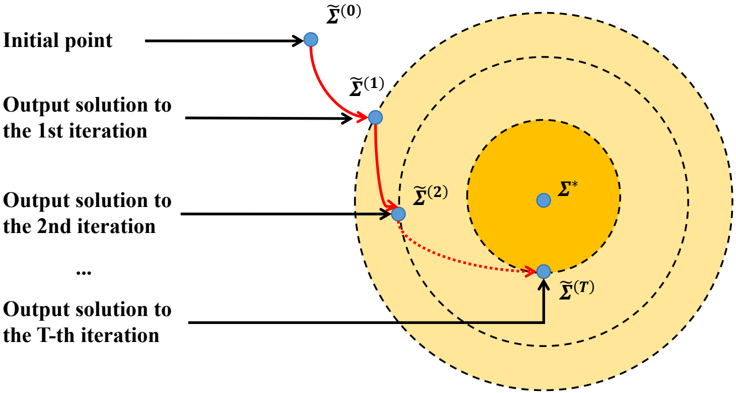
- There exists $\nu>0$ such that the derivative satisfies $p'_\lambda(t)=0$ for all $t\geq\nu$;
- Both $p_\lambda(t)$ and $q_\lambda(t)$ are symmetric about zero, i.e., $p_\lambda(t)=p_\lambda(-t)$, $q_\lambda(t)=q_\lambda(-t)$;
- The derivative $q'_\lambda(t)$ is monotonic and Lipschitz continuous in the interval $[0,\infty)$. Explicitly, for $t_2\geq t_1\geq 0$, there exist constants $\zeta^-\geq\zeta^+>0$ such that $-\zeta^-\leq\frac{q'_\lambda(t_2)-q'_\lambda(t_1)}{(t_2-t_1)}\leq-\zeta^+$.
- Both $q_\lambda(t)$ and its derivative vanish at zero, i.e., $q_\lambda(0)=q'_\lambda(0)=0$;
- There exists a constant $\lambda>0$ bounding the magnitude of the derivative, i.e., $|q'_\lambda(t)|\leq\lambda$.

Optimization Algorithm

Algorithm 1: Proximal Gradient Algorithm

Input $\lambda_0 > 0$, $\epsilon > 0$, $L_{\min} > 0$, $\eta \in (0,1)$, $\delta \in (0,1)$
Initialize $\boldsymbol{X}^0 = \mathbf{0}$, $L_0 = L_{\min}$
for $t = 0, 1, \ldots, T-1$ **do**
    $\lambda_{t+1} = \eta\lambda_t$; $\epsilon_{t+1} = \lambda_t/4$;
    $k = 0$; $\boldsymbol{X}^k = \boldsymbol{X}^t$;
    while $\omega_{\lambda_{t+1}}(\boldsymbol{X}^k) > \epsilon_{t+1}$ **do**
        $k = k + 1$;
        $\boldsymbol{X}^k = \arg\min_{\boldsymbol{X}} \widetilde{F}_{L,\lambda}(\boldsymbol{X}; \boldsymbol{X}^{k-1})$;
        If $F(\boldsymbol{X}^k) > \widetilde{F}(\boldsymbol{X}^k; \boldsymbol{X}^{k-1})$ **then**
            $L_{k-1} = 2L_{k-1}$;
        end if
        $L_k = \max\{L_{\min}, L_{k-1}/2\}$
    end while
    $\boldsymbol{X}^{t+1} = \boldsymbol{X}^k$; $L_{t+1} = L_k$
end for
Output$\{\boldsymbol{X}^t\}_{t=1}^T$

Algorithm Illustration



Preliminaries

Consider the singular value decomposition $\boldsymbol{X}^\star = \boldsymbol{U}^\star \boldsymbol{\Sigma}^\star \boldsymbol{V}^{\star\top}$, where $\boldsymbol{U}^\star, \boldsymbol{V}^\star \in \mathbb{R}^{d\times r}$, and $\boldsymbol{\Sigma}^\star = \mathrm{diag}(\sigma_1^\star, \ldots, \sigma_r^\star)$. We introduce the subspace $\mathcal{F}$ and $\mathcal{F}^\perp$, which are defined in terms of the row and column spaces of the matrices:

$$\mathcal{F}\left(\boldsymbol{U}^\star, \boldsymbol{V}^\star\right) := \left\{\boldsymbol{\Delta} \mid \mathrm{row}\left(\boldsymbol{\Delta}\right) \subseteq \boldsymbol{V}^\star, \mathrm{col}\left(\boldsymbol{\Delta}\right) \subseteq \boldsymbol{U}^\star\right\},$$
$$\mathcal{F}^\perp\left(\boldsymbol{U}^\star, \boldsymbol{V}^\star\right) := \left\{\boldsymbol{\Delta} \mid \mathrm{row}\left(\boldsymbol{\Delta}\right) \perp \boldsymbol{V}^\star, \mathrm{col}\left(\boldsymbol{\Delta}\right) \perp \boldsymbol{U}^\star\right\}.$$

Restricted Region

Define a local region $\mathcal{R}$ as

$$\mathcal{R} = \left\{\boldsymbol{\Delta} \mid \left\|\Pi_{\mathcal{F}^\perp}\left(\boldsymbol{\Delta}\right)\right\|_* \le 5\left\|\Pi_{\mathcal{F}}\left(\boldsymbol{\Delta}\right)\right\|_*\right\},$$

where $\Pi_{\mathcal{F}(\cdot)}$ is the projection operator that projects matrices into the subspace $\mathcal{F}$.

Essential Assumptions

Assumption 2 (RSC & RSM)

- The empirical loss function $f(\cdot)$ is ρ^--strongly convex and ρ^+-smooth over $\mathcal{R}$ with $\infty > \rho^+ \geq \rho^- > 0$. Specifically, for all $\boldsymbol{X} - \boldsymbol{X}' \in \mathcal{R}$, we have:

$$\left\langle \boldsymbol{X} - \boldsymbol{X}', \nabla f(\boldsymbol{X}) - \nabla f\left(\boldsymbol{X}'\right)\right\rangle \geq \rho^- \left\|\boldsymbol{X} - \boldsymbol{X}'\right\|_{\mathrm{F}}^2,$$
$$\left\langle \boldsymbol{X} - \boldsymbol{X}', \nabla f(\boldsymbol{X}) - \nabla f\left(\boldsymbol{X}'\right)\right\rangle \geq \frac{\|\nabla f(\boldsymbol{X}) - \nabla f(\boldsymbol{X}')\|_{\mathrm{F}}^2}{\rho^+}.$$

Assumption 3 (Minimal Signal Strength)

- The singular value of the ground truth $\boldsymbol{X}^\star$ satisfies:

$$\min_{i \in \mathcal{S}_1 \cup \mathcal{S}_2} |\sigma_i^\star| \geq \nu + 2\sqrt{s_1 + s_2}\|\boldsymbol{A}^\top \boldsymbol{E}\boldsymbol{B}\|_{\mathrm{F}}/(m^2\rho).$$

Theoretical Results I

Define $\mathcal{S}_1 = \{i \mid \sigma_i^\star \geq \nu\}$, $\mathcal{S}_2 = \{i \mid \nu > \sigma_i^\star > 0\}$ with their corresponding cardinalities given by $s_1 = |\mathcal{S}_1|$ and $s_2 = |\mathcal{S}_2|$.

Theorem 1

Suppose Assumptions 1 and 2 hold, if $\rho^- > \zeta^-$, $\lambda \gtrsim \|\boldsymbol{A}^\top \boldsymbol{E} \boldsymbol{B}\|_{\mathrm{F}}/m^2$, *we have:*

$$\|\widehat{\boldsymbol{X}} - \boldsymbol{X}^\star\|_{\mathrm{F}} \lesssim \tau\sqrt{s_1} + \sqrt{s_2}$$

where $\tau = \|\Pi_{\mathcal{F}_{S_1}}\left(\nabla f\left(\boldsymbol{X}^\star\right)\right)\|_{\mathrm{F}}$ *and* $\mathcal{F}_{S_1}$ *is a subspace of* $\mathcal{F}$ *associated with* S_1*.*

- The *oracle rate* refers to the statistical convergence rate of the *oracle estimator*, defined as $\widehat{\boldsymbol{X}}^O = \arg\min_{\boldsymbol{X} \in \mathcal{F}(\boldsymbol{U}^\star, \boldsymbol{V}^\star)} f\left(\boldsymbol{X}\right)$, which knows the true rank spaces in advance. By the mean value theorem, it is easy to obtain that $\|\widehat{\boldsymbol{X}}^O - \boldsymbol{X}^\star\|_{\mathrm{F}} \lesssim \|\Pi_{\mathcal{F}}\left(\nabla f\left(\boldsymbol{X}^\star\right)\right)\|_{\mathrm{F}}$.

Theoretical Results II

Theorem 2 (oracle property)

Suppose Assumptions 1, 2 and 3 hold.
If $\rho > \zeta^{-}$, *and*

$$\lambda \geq \frac{(\rho^{-} + \sqrt{s_1 + s_2}\rho^{+})\|\boldsymbol{A}^{\top}\boldsymbol{E}\boldsymbol{B}\|_{\mathrm{F}}}{2m^2\rho^{-}},$$

we have

$$\operatorname{rank}(\widehat{\boldsymbol{X}}) = \operatorname{rank}(\widehat{\boldsymbol{X}}^{O}) = \operatorname{rank}\left(\boldsymbol{X}^{\star}\right)$$

and

$$\|\widehat{\boldsymbol{X}} - \boldsymbol{X}^{\star}\|_{\mathrm{F}} \lesssim \sqrt{s_1}\tau,$$

where $\tau = \|\Pi_{\mathcal{F}}\left(\nabla f\left(\boldsymbol{X}^{\star}\right)\right)\|_{\mathrm{F}}$.

Theoretical Results III

Corollary 3

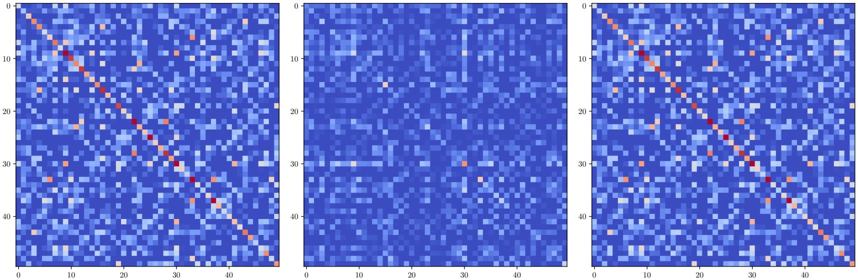
Consider the noise entries are i.i.d. sub-Gaussian random variables with variance κ, *and the vectorized sketching matrices* $\mathrm{vec}(\boldsymbol{A})$ *and* $\mathrm{vec}(\boldsymbol{B})$ *follow sub-Gaussian distributions* $\mathrm{vec}(\boldsymbol{A}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta}_1)$, $\mathrm{vec}(\boldsymbol{B}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta}_2)$. *We term these as* $\boldsymbol{\Theta}_1$*-ensemble and* $\boldsymbol{\Theta}_2$*-ensemble, and* $\varpi_1(\boldsymbol{\Theta}_1) = \sqrt{\sup_{\|\boldsymbol{u}\|_2=1, \|\boldsymbol{v}\|_2=1} \mathrm{Var}(\boldsymbol{u}^\top \boldsymbol{A} \boldsymbol{v})}$, $\varpi_2(\boldsymbol{\Theta}_2) = \sqrt{\sup_{\|\boldsymbol{u}\|_2=1, \|\boldsymbol{v}\|_2=1} \mathrm{Var}(\boldsymbol{u}^\top \boldsymbol{B} \boldsymbol{v})}$. *Assuming Assumptions 1 and 2 hold, and* $\boldsymbol{A}$ *and* $\boldsymbol{B}$ *are sampled from* $\boldsymbol{\Theta}_1$*-ensemble and* $\boldsymbol{\Theta}_2$*-ensemble, respectively, if* $\rho \gtrsim \sqrt{\lambda_{\min}(\boldsymbol{\Theta}_1)\lambda_{\min}(\boldsymbol{\Theta}_2)} > \zeta^-$ *and* $\lambda \gtrsim \kappa\sqrt{\varpi_1\varpi_2 d}/m$, *then with probability at least* $1 - \exp(-d)$. *Additionally, according to Theorem 1, we have:*

$$\left\| \widehat{\boldsymbol{X}} - \boldsymbol{X}^\star \right\|_{\mathrm{F}} \lesssim \mathcal{O}\left(\sqrt{\frac{\varpi_1\varpi_2}{\lambda_{\min}(\boldsymbol{\Theta}_1)\lambda_{\min}(\boldsymbol{\Theta}_2)}} \frac{\kappa(\sqrt{s_2 d} + s_1)}{m} \right).$$

With Assumption 3, the convergence rate improves to

$$\mathcal{O}\left(\sqrt{\frac{\varpi_1\varpi_2}{\lambda_{\min}(\boldsymbol{\Theta}_1)\lambda_{\min}(\boldsymbol{\Theta}_2)}} \frac{\kappa s_1}{m} \right).$$

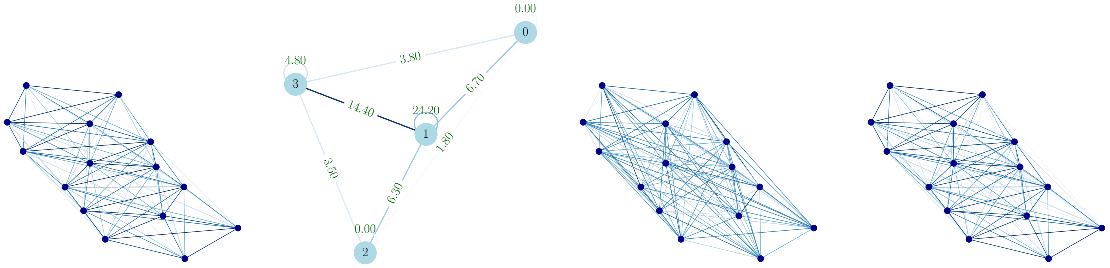
Experiment I



(a) Ground Truth (b) Nuclear (c) MCP

Figure 9: Heatmaps show the recovery of a 50×50 low-rank matrix (rank = 10, Gaussian generated) from noisy bilinear measurements with $\mathcal{N}(0, 0.01)$ noise. MCP achieves near-perfect recovery, while nuclear-norm minimization exhibits excessive smoothing and weakens low-rank features.

Experiment II



(a) (b) (c) (d)

Figure 10: An illustrative example of graph sketching is shown as follows: (a) The original graph $\mathcal{G}$ with 15 nodes; (b) The sketch of the graph $\mathcal{G}$, where the nodes represent the partitions and the edges represent the total number of edges of G that cross these partitions; (c) The graph recovered using least squares error minimization; (d) The graph recovered using the SCAD penalty.

Experiment III

Dataset	Nuclear	Weighted Nuclear	SCAD	MCP
Fashion-MNIST	0.7683 ± 0.1076	0.7286 ± 0.0630	0.0124 ± 0.0033	$\mathbf{0.0108 \pm 0.0012}$
Places365	0.4472 ± 0.0827	0.4647 ± 0.0484	0.0079 ± 0.0013	$\mathbf{0.0066 \pm 0.0021}$
ImageNet-O	0.4574 ± 0.1502	0.5069 ± 0.1149	$\mathbf{0.0137 \pm 0.0079}$	0.0138 ± 0.0056

Table 1: Low-rank recovery experiments on three real-world image datasets: Fashion-MNIST ($d = 28, r = 10$), Places365 ($d = 256, r = 100$), and ImageNet-O ($d = 512, r = 200$). Observations are formed using bilinear sketching with $m = 5, 50, 80$ and additive $\mathcal{N}(0, 0.01)$ noise. We compare nuclear norm, weighted nuclear norm, and nonconvex methods (SCAD and MCP).

Conclusions

- We have proposed a novel approach for low-rank matrix estimation from bilinear measurements using the non-convex penalty.

- We have presented both the theoretical and empirical results.

Thank you!

Website: https://wenbinwang12.github.io
Email: wangwb2023@shanghaitech.edu.cn