A Kaczmarz Method for Blind Deconvolution

Wen Huang

Xiamen University

Febuary 2, 2026

Joint work with Zilin Yang

Seminar on Large Models and Optimization
Kunming

Outline

- Blind deconvolution
- A review of Kaczmarz methods
- A Kaczmarz method for blind deconvolution
- Preliminary numerical results
- Summary and future work

Blind deconvolution

[Blind deconvolution]

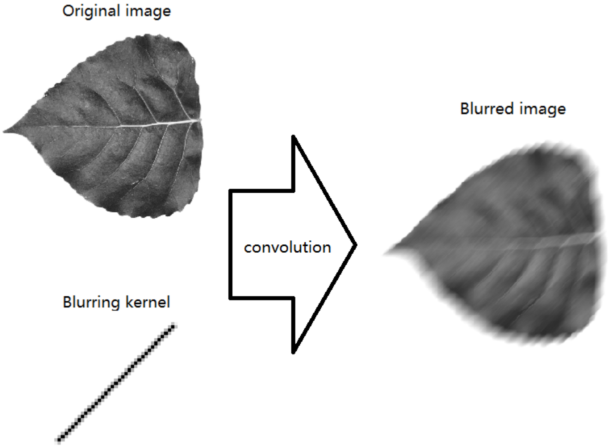
Blind deconvolution is to recover two unknown signals from their convolution.

Blurred image

Blind deconvolution

[Blind deconvolution]

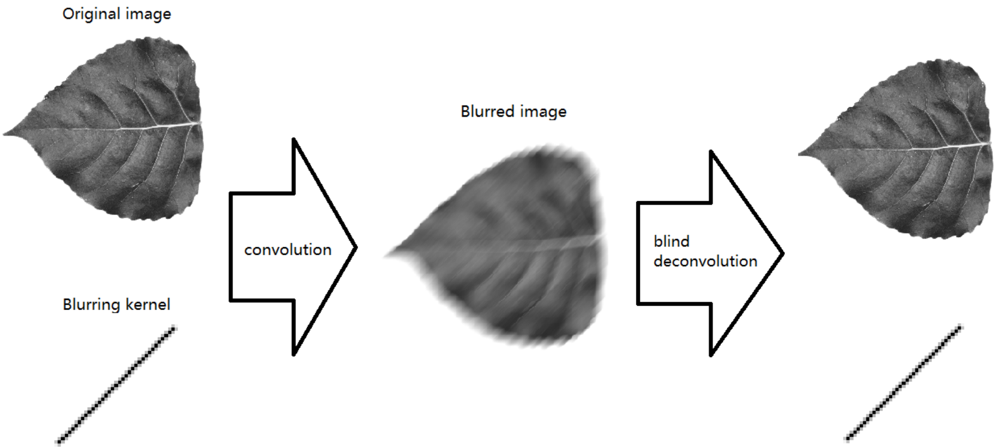
Blind deconvolution is to recover two unknown signals from their convolution.



Blind deconvolution

[Blind deconvolution]

Blind deconvolution is to recover two unknown signals from their convolution.



Blind deconvolution

[Blind deconvolution (Discretized version)]

Blind deconvolution is to recover two unknown signals $\mathbf{w} \in \mathbb{C}^L$ and $\mathbf{x} \in \mathbb{C}^L$ from their convolution $\mathbf{y} = \mathbf{w} * \mathbf{x} \in \mathbb{C}^L$.

- We only consider circular convolution:

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_L \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1 & \mathbf{w}_L & \mathbf{w}_{L-1} & \dots & \mathbf{w}_2 \\ \mathbf{w}_2 & \mathbf{w}_1 & \mathbf{w}_L & \dots & \mathbf{w}_3 \\ \mathbf{w}_3 & \mathbf{w}_2 & \mathbf{w}_1 & \dots & \mathbf{w}_4 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{w}_L & \mathbf{w}_{L-1} & \mathbf{w}_{L-2} & \dots & \mathbf{w}_1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_L \end{bmatrix}$$

- Let $y = \mathbf{F}\mathbf{y}$, $w = \mathbf{F}\mathbf{w}$, and $x = \mathbf{F}\mathbf{x}$, where $\mathbf{F}$ is the DFT matrix;
- $y = w \odot x$, where $\odot$ is the Hadamard product, i.e., $y_i = w_i x_i$.
- Equivalent question: Given y, find w and x.

Blind deconvolution

Problem: Given $y \in \mathbb{C}^L$, find $w, x \in \mathbb{C}^L$ so that $y = w \odot x$.

- An ill-posed problem. Infinite solutions exist;

Blind deconvolution

Problem: Given $y \in \mathbb{C}^L$, find $w, x \in \mathbb{C}^L$ so that $y = w \odot x$.

- An ill-posed problem. Infinite solutions exist;
- Assumption: w and x are in known subspaces, i.e., $w = Bh$ and $x = \overline{Cm}$, $B \in \mathbb{C}^{L\times K}$ and $C \in \mathbb{C}^{L\times N}$;

Blind deconvolution

Problem: Given $y \in \mathbb{C}^L$, find $w, x \in \mathbb{C}^L$ so that $y = w \odot x$.

- An ill-posed problem. Infinite solutions exist;
- Assumption: w and x are in known subspaces, i.e., $w = Bh$ and $x = \overline{Cm}$, $B \in \mathbb{C}^{L\times K}$ and $C \in \mathbb{C}^{L\times N}$;
  - Reasonable in various applications;
  - Leads to mathematical rigor; $(L/(K+N)$ reasonably large)

Blind deconvolution

Problem: Given $y \in \mathbb{C}^L$, find $w, x \in \mathbb{C}^L$ so that $y = w \odot x$.

- An ill-posed problem. Infinite solutions exist;
- Assumption: w and x are in known subspaces, i.e., $w = Bh$ and $x = \overline{Cm}$, $B \in \mathbb{C}^{L \times K}$ and $C \in \mathbb{C}^{L \times N}$;
  - Reasonable in various applications;
  - Leads to mathematical rigor; $(L/(K+N)$ reasonably large)

Problem under the assumption

Given $y \in \mathbb{C}^L$, $B \in \mathbb{C}^{L \times K}$ and $C \in \mathbb{C}^{L \times N}$, find $h \in \mathbb{C}^K$ and $m \in \mathbb{C}^N$ so that

$$y = Bh \odot \overline{Cm} = \mathrm{diag}(Bhm^* C^*).$$

Blind deconvolution

Find h, m, s. t. $y = \mathrm{diag}(Bhm^* C^*)$;

- Ahmed et al. [ARR14] [1]
  - Convex relaxation ($X = h\, m^*$):$\min_{y = \mathrm{diag}(B\ X\ C^*)} \|X\|_n$;
  - The unique minimizer $\xlongequal{\text{high probability}}$ true solution;
  - The convex problem may be expensive to solve;

[1] A. Ahmed, B. Recht, and J. Romberg, Blind deconvolution using convex programming, *IEEE Transactions on Information Theory*, 60:1711-1732, 2014

Blind deconvolution

$$\boxed{\text{Find } h, m, \text{ s. t. } y = \mathrm{diag}(Bhm^* C^*);}$$

- Ahmed et al. [ARR14] [1]
  - Convex relaxation ($X = h\,m^*$):$\min_{y = \mathrm{diag}(B\,X\,C^*)} \|X\|_n$;
  - The unique minimizer $\xlongequal{\text{high probability}}$ true solution;
  - The convex problem may be expensive to solve;
- Li et al. [LLSW19] [2]
  - Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$;
  - (Wirtinger flow + good initialization) $\xRightarrow{\text{high probability}}$ true solution;
  - Lower successful recovery rate than alternating minimization;

[1] A. Ahmed, B. Recht, and J. Romberg, Blind deconvolution using convex programming, *IEEE Transactions on Information Theory*, 60:1711-1732, 2014

[2] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *Applied and Computational Harmonic Analysis*, 47:3, 893-934, 2019

Blind deconvolution

$\boxed{\text{Find } h, m, \text{ s. t. } y = \mathrm{diag}(Bhm^* C^*);}$

- Ahmed et al. [ARR14] [1]
  - Convex relaxation ($X = h\, m^*$):$\min_{y = \mathrm{diag}(B\ X\ C^*)} \|X\|_n$;
  - The unique minimizer $\xrightarrow[]{\text{high probability}}$ true solution;
  - The convex problem may be expensive to solve;
- Li et al. [LLSW19] [2]
  - Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$;
  - (Wirtinger flow + good initialization) $\xrightarrow{\text{high probability}}$ true solution;
  - Lower successful recovery rate than alternating minimization;
- Huang and Hand [HH18] [3]
  - Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$;
  - (Riemannian method + good initialization) $\xrightarrow{\text{high probability}}$ true solution;
  - Higher successful recovery rate;

[1] A. Ahmed, B. Recht, and J. Romberg, Blind deconvolution using convex programming, *IEEE Transactions on Information Theory*, 60:1711-1732, 2014

[2] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *Applied and Computational Harmonic Analysis*, 47:3, 893-934, 2019

[3] Wen Huang, Paul Hand. Blind Deconvolution by a Steepest Descent Algorithm on a Quotient Manifold, *SIAM Journal on Imaging Sciences*, 11:4, pp. 2757-2785, 2018.

Blind deconvolution

$\boxed{\text{Find } h, m, \text{ s. t. } y = \mathrm{diag}(Bhm^* C^*);}$

- Ahmed et al. [ARR14]
  - Convex relaxation ($X = h\, m^*$):$\min_{y = \mathrm{diag}(B\, X\, C^*)} \|X\|_n$;
  - The unique minimizer $\xlongequal{\text{high probability}}$ true solution;
  - The convex problem may be expensive to solve;
- Li et al. [LLSW19]
  - Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$;
  - (Wirtinger flow + good initialization) $\xRightarrow{\text{high probability}}$ true solution;
  - Lower successful recovery rate than alternating minimization;
- Huang and Hand [HH18]
  - Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$;
  - (Riemannian method + good initialization) $\xRightarrow{\text{high probability}}$ true solution;
  - Higher successful recovery rate;

The theoretical results assume (1) C: complex Gaussian distribution and (2) $B^* B = I_K$ and $\|b_i\|^2 < \phi K/L, i = 1, \ldots, L$.

Blind deconvolution

Problem

Given $y \in \mathbb{C}^L$, $B \in \mathbb{C}^{L\times K}$ and $C \in \mathbb{C}^{L\times N}$, find $h \in \mathbb{C}^K$ and $m \in \mathbb{C}^N$ so that

$$y = Bh \odot \overline{Cm} = \mathrm{diag}(Bhm^* C^*).$$

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

- Solution set:
$$\mathcal{N} = \{X \in \mathbb{C}^{K\times N} : \mathrm{rank}(X) = 1, y = \mathrm{diag}(BXC^*)\};$$
- Space:
$$\mathcal{N}_i = \{X \in \mathbb{C}^{K\times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}, i = 1, 2, \ldots, L.$$

Outline

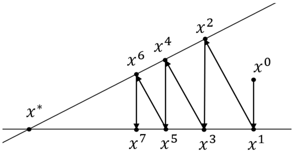
- Blind deconvolution
- A review of Kaczmarz methods
- A Kaczmarz method for blind deconvolution
- Preliminary numerical results
- Summary and future work

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for the linear system $y = Ax$

- $\mathcal{N}_i = \{x \mid a_i^T x = y_i\}$ is a hyperplane;
- Projection to a hyperplane has a closed-form solution;



Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for the linear system $y = Ax$

Input: y, A, an initial point x_0;
1: **for** $l = 0, 1, 2, \ldots,$
2: Select a row of A, denoted by a_i^T;
3: Compute $x_{l+1} = x_l + \frac{y_i - a_i^T x_l}{\|a_i\|_2^2} a_i$;
4: **end for**

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for the linear system $y = Ax$

- Originally for linear equations [Kac37];
- Sweeps through the rows cyclically [DH97, Gal05, XZ02];
- Selects i row with probability $\frac{\|a_i\|_2^2}{\|A\|_F^2}$ and converges linearly [SV09];

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for the phase retrieval $y = |Ax|^2$

- $\mathcal{N}_i = \{x \mid |a_i^* x|^2 = y_i\}$ is a set of hyperplanes;
- Projection to a union of hyperplanes has a closed-form solution;

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for phase retrieval $y = |Ax|^2$

Input: y, A, an initial point x_0;
1: **for** $l = 0, 1, 2, \ldots,$
2: Select a row of A, denoted by a_i^T;
3: Compute $\theta_l = \angle\langle a_i, x_l\rangle$;
4: Compute $x_{l+1} = x_l + \frac{\sqrt{y_i}e^{\sqrt{-1}\theta_l} - \langle a_i, x_l\rangle}{\|a_i\|_2^2} a_i$;
5: **end for**

Kaczmarz method

It is an iterative projection technique by repeatedly projecting a current estimate onto a space that contains the solution set.

Kaczmarz method for phase retrieval $y = |Ax|^2$

- Extended for phase retrieval problem [Wei15, LGL15];
- An increasing number of mathematicians have contributed to this area of research [JG17, TV19, RJ20, RFK21, ZF22, HW22];
- Linearly converge to a neighborhood [Wei15] and linearly converge to the solution [HW22] under restricted strong convexity;

A general setting of Kaczmarz methods

Input: An initial point x_0;
1: **for** $l = 0, 1, 2, \ldots,$ **do**
2: Select a set $\mathcal{N}_i$ that contains the solution space;
3: Compute $x_{l+1} = \arg\min_{x \in \mathcal{N}_i} \|x - x_l\|^2$;
4: **end for**

Linear system

- $\mathcal{N}_i$: a hyperplane $a^T x = b_i$;
- Closed-form solution exists;

Phase retrieval

- $\mathcal{N}_i$: a union of hyperplanes
- Closed-form solution exists;

A general setting of Kaczmarz methods

Input: An initial point x_0;

1: **for** $l = 0, 1, 2, \ldots,$ **do**

2: Select a set $\mathcal{N}_i$ that contains the solution space;

3: Compute $x_{l+1} = \arg\min_{x \in \mathcal{N}_i} \|x - x_l\|^2$;

4: **end for**

Blind deconvolution

- $\mathcal{N}_i = \{X \in \mathbb{C}^{K \times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
- Projection: $X_{l+1} = \arg\min_{X \in \mathcal{N}_i} \|X - X_l\|$;

A general setting of Kaczmarz methods

Input: An initial point x_0;
1: **for** $l = 0, 1, 2, \dots,$ **do**
2: Select a set $\mathcal{N}_i$ that contains the solution space;
3: Compute $x_{l+1} = \arg\min_{x \in \mathcal{N}_i} \|x - x_l\|^2$;
4: **end for**

Blind deconvolution

- $\mathcal{N}_i = \{X \in \mathbb{C}^{K \times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
- Projection: $X_{l+1} = \arg\min_{X \in \mathcal{N}_i} \|X - X_l\|$;

A general setting of Kaczmarz methods

Input: An initial point x_0;

1: **for** $l = 0, 1, 2, \ldots,$ **do**

2: Select a set $\mathcal{N}_i$ that contains the solution space;

3: Compute $x_{l+1} = \arg\min_{x \in \mathcal{N}_i} \|x - x_l\|^2$;

4: **end for**

Blind deconvolution

- $\mathcal{N}_i = \{X \in \mathbb{C}^{K \times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
- Projection: $X_{l+1} = \arg\min_{X \in \mathcal{N}_i} \|X - X_l\|$;

Does the projection yield a closed form solution?

A general setting of Kaczmarz methods

Input: An initial point x_0;

1: **for** $l = 0, 1, 2, \ldots,$ **do**

2: Select a set $\mathcal{N}_i$ that contains the solution space;

3: Compute $x_{l+1} = \arg\min_{x \in \mathcal{N}_i} \|x - x_l\|^2$;

4: **end for**

Blind deconvolution

- $\mathcal{N}_i = \{X \in \mathbb{C}^{K \times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
- Projection: $X_{l+1} = \arg\min_{X \in \mathcal{N}_i} \|X - X_l\|$;

Does the projection yield a closed form solution?

No, as far as we know

Outline

- Blind deconvolution
- A review of Kaczmarz methods
- A Kaczmarz method for blind deconvolution
- Preliminary numerical results
- Summary and future work

A Kaczmarz method for blind deconvolution

Input: An initial point X_0;
1: **for** $l = 0, 1, 2, \ldots,$ **do**
2: Select $\mathcal{N}_i = \{X \in \mathbb{C}^{K\times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
3: Compute $X_{l+1} = \arg\min_{X\in\mathcal{N}_i} \|X - X_l\|^2$;
4: **end for**

Key question: can we compute the projection efficiently?

[1] Yixian Liu, An efficient method for non-convex blind deconvolution, *IEEE Access*, 7:113663–113674, 2019

A Kaczmarz method for blind deconvolution

Input: An initial point X_0;

1: **for** $l = 0, 1, 2, \ldots,$ **do**
2: Select $\mathcal{N}_i = \{X \in \mathbb{C}^{K\times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
3: Compute $X_{l+1} \approx \arg\min_{X\in\mathcal{N}_i} \|X - X_l\|^2$;
4: **end for**

Key question: can we compute the projection efficiently?

An existing Kaczmarz method ([Liu19][1])

- Nonconvex problem: $\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \sum_{i=1}^{L} |y_i - b_i^* h m^* c_i|^2$;
- Alternating minimization for one round;
- Good initialization $\xrightarrow{\text{high probability}}$ true solution;

[1] Yixian Liu, An efficient method for non-convex blind deconvolution, *IEEE Access*, 7:113663–113674, 2019

A Kaczmarz method for blind deconvolution

Input: An initial point X_0;
1: **for** $l = 0, 1, 2, \dots,$ **do**
2: Select $\mathcal{N}_i = \{X \in \mathbb{C}^{K\times N} : \mathrm{rank}(X) = 1, y_i = b_i^* X c_i\}$;
3: Compute $X_{l+1} = \arg\min_{X\in\mathcal{N}_i} \|X - X_l\|^2$;
4: **end for**

Key question: can we compute the projection efficiently?

The proposed Kaczmarz method

- Equivalently reduce to 2×2 subproblem;
- Multiple approaches for the reduced problem;

Equivalently reduce to 2×2 subproblem;

Assumption

$y_i \neq 0, i = 1, \ldots, L.$

Theorem

Suppose the above assumption holds. Then the two problems

$$\min_{X \in \mathbb{C}^{K \times N}, \mathrm{rank}(X)=1, y_i = b_i^* X c_i} \|X - h_l m_l^*\|_F^2 \text{ and } \min_{\tilde{X} \in \mathbb{C}^{2 \times 2}, \mathrm{rank}(\tilde{X})=1, y_i = \tilde{b}_i^* \tilde{X} \tilde{c}_i} \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2$$

are equivalent in the sense that

$$X_*: \text{stationary/local/global} \Longleftrightarrow \tilde{X}_* = Q_1^* X_* Q_2: \text{stationary/local/global};$$

- QR decomposition: $[h_l \; b_i] = Q_1 R_1$ and $[m_l \; c_i] = Q_2 R_2$;
- $\tilde{h}_l = Q_1^* h_l$, $\tilde{m}_l = Q_2^* m_l$, $\tilde{b}_i = Q_1^* b_i$, and $\tilde{c}_i = Q_2^* c_i$;

Equivalently reduce to 2×2 subproblem;

Assumption

$y_i \neq 0, i = 1, \ldots, L.$

- Easy to verify;
- $\mathbb{C}_1^{K \times N}$ and $\mathcal{H}_i$ intersect transversally $\Longrightarrow \mathcal{N}_i = \mathbb{C}_1^{K \times N} \cap \mathcal{H}_i$ is an embedded submanifold;
  - $\mathrm{T}_X \mathcal{N}_i = \mathrm{T}_X(\mathbb{C}_1^{K \times N} \cap \mathcal{H}_i) = \mathrm{T}_X \mathbb{C}_1^{K \times N} \cap \mathrm{T}_X \mathcal{H}_i$;
  - $\mathrm{N}_X \mathcal{N}_i = \mathrm{N}_X(\mathbb{C}_1^{K \times N} \cap \mathcal{H}_i) = \mathrm{N}_X \mathbb{C}_1^{K \times N} + \mathrm{N}_X \mathcal{H}_i$;

Theorem

Suppose the above assumption holds. Then the two problems

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_i=b_i^*Xc_i}\|X-h_lm_l^*\|_F^2 \text{ and } \min_{\tilde{X}\in\mathbb{C}^{2\times 2},\mathrm{rank}(\tilde{X})=1,y_i=\tilde{b}_i^*\tilde{X}\tilde{c}_i}\|\tilde{X}-\tilde{h}_l\tilde{m}_l^*\|_F^2$$

are equivalent in the sense that

$$X_*\text{: stationary/local/global} \Longleftrightarrow \tilde{X}_* = Q_1^*X_*Q_2\text{: stationary/local/global;}$$

where $[h_l\ b_i] = Q_1R_1$ and $[m_l\ c_i] = Q_2R_2$, $[m_l\ c_i] = Q_2R_2$, $\tilde{h}_l = Q_1^*h_l$, $\tilde{m}_l = Q_2^*m_l$, $\tilde{b}_i = Q_1^*b_i$, and $\tilde{c}_i = Q_2^*c_i$;

A Sketch of Proof

1. Let $h_*m_*^*$ denote X_*. Then $h_* \in \mathrm{span}(Q_1)$ and $m_* \in \mathrm{span}(Q_2)$;
2. Equivalence of the first-order optimality conditions (manifold structures);
3. Equivalence of local and global minimizers;

Solve

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_i=b_i^* Xc_i} \|X-h_l m_l^*\|_F^2$$

by

$$\min_{\tilde{X}\in\mathbb{C}^{2\times 2},\mathrm{rank}(\tilde{X})=1,y_i=\tilde{b}_i^* \tilde{X}\tilde{c}_i} \|\tilde{X}-\tilde{h}_l \tilde{m}_l^*\|_F^2$$

Algorithm: Compute $X_{l+1} = \arg\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_i=b_i^* Xc_i} \|X-h_l m_l^*\|_F^2$

Input: y_i, b_i, c_i, h_l, m_l;

1: Compute QR decomposition $Q_1R_1 = \begin{bmatrix} h_l & b_i \end{bmatrix}$, $Q_2R_2 = \begin{bmatrix} m_l & c_i \end{bmatrix}$;

2: Compute $\tilde{h}_l \leftarrow Q_1^* h_l = R_1(:,1)$, $\tilde{m}_l \leftarrow Q_2^* m_l = R_2(:,1)$,
$\tilde{b}_i \leftarrow Q_1^* b_i = R_1(:,2)$, $\tilde{c}_i \leftarrow Q_2^* c_i = R_2(:,2)$;

3: Find $(\tilde{h}_{l+1}, \tilde{m}_{l+1}) = \underset{y_i=\tilde{b}_i^* hm^* \tilde{c}_i}{\operatorname{argmin}} \|hm^* - \tilde{h}_l\tilde{m}_l^*\|_{\mathrm{F}}^2$;

4: Compute $h_{l+1} \leftarrow Q_1\tilde{h}_{l+1}$, $m_{l+1} \leftarrow Q_2\tilde{m}_{l+1}$;

Solve

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_i=b_i^* X c_i} \|X - h_l m_l^*\|_F^2$$

by

$$\min_{\tilde{X}\in\mathbb{C}^{2\times 2},\mathrm{rank}(\tilde{X})=1,y_i=\tilde{b}_i^* \tilde{X}\tilde{c}_i} \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2$$

Algorithm: Compute $X_{l+1} = \arg\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_i=b_i^* X c_i} \|X - h_l m_l^*\|_F^2$

Input: y_i, b_i, c_i, h_l, m_l;

1: Compute QR decomposition $Q_1 R_1 = \begin{bmatrix} h_l & b_i \end{bmatrix}, Q_2 R_2 = \begin{bmatrix} m_l & c_i \end{bmatrix}$;

2: Compute $\tilde{h}_l \leftarrow Q_1^* h_l = R_1(:,1), \tilde{m}_l \leftarrow Q_2^* m_l = R_2(:,1)$,
$\tilde{b}_i \leftarrow Q_1^* b_i = R_1(:,2), \tilde{c}_i \leftarrow Q_2^* c_i = R_2(:,2)$;

3: Find $(\tilde{h}_{l+1}, \tilde{m}_{l+1}) = \operatorname*{argmin}_{y_i=\tilde{b}_i^* hm^* \tilde{c}_i} \|hm^* - \tilde{h}_l \tilde{m}_l^*\|_\mathrm{F}^2$;

4: Compute $h_{l+1} \leftarrow Q_1 \tilde{h}_{l+1}, m_{l+1} \leftarrow Q_2 \tilde{m}_{l+1}$;

How to solve this small-dimensional problem?

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

All stationary points hm^* satisfy that $\exists \alpha, \beta \in \mathbb{C}$ such that

$$\underbrace{\tilde{h}\tilde{m}^* - \tilde{h}_l\tilde{m}_l}_{\text{Euclidean gradient}} = \underbrace{\alpha \tilde{h}_\perp \tilde{m}_\perp^* + \beta \tilde{b}_i \tilde{c}_i^*}_{\text{Normal space}} \Longrightarrow$$

$$\beta \tilde{b}_i \tilde{c}_i^* + \tilde{h}_l \tilde{m}_l = \tilde{h}\tilde{m}^* - \alpha \tilde{h}_\perp \tilde{m}_\perp^* \Longrightarrow$$

given β, then $\tilde{h}$ and $\tilde{m}$ can be found.

View $\tilde{h}$ and $\tilde{m}$ as a function of β. Use $y_i = \tilde{b}^* \tilde{h}\tilde{m}^* \tilde{c}_i$ to find β.

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

All stationary points hm^* satisfy that $\exists \alpha, \beta \in \mathbb{C}$ such that

$$\underbrace{\tilde{h}\tilde{m}^* - \tilde{h}_l \tilde{m}_l}_{\text{Euclidean gradient}} = \underbrace{\alpha \tilde{h}_\perp \tilde{m}_\perp^* + \beta \tilde{b}_i \tilde{c}_i^*}_{\text{Normal space}} \Longrightarrow$$

$$\beta \tilde{b}_i \tilde{c}_i^* + \tilde{h}_l \tilde{m}_l = \tilde{h}\tilde{m}^* - \alpha \tilde{h}_\perp \tilde{m}_\perp^* \Longrightarrow$$

given β, then $\tilde{h}$ and $\tilde{m}$ can be found.

View $\tilde{h}$ and $\tilde{m}$ as a function of β. Use $y_i = \tilde{b}^* \tilde{h} \tilde{m}^* \tilde{c}_i$ to find β.

Too complicated

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

QCQP formulation

$$\begin{aligned} \min_{(u,v)} \quad & \|\bar{h}\bar{m}^* + \tilde{b}_{i,\perp} u^* + v \tilde{c}_{i,\perp}^* - \tilde{h}_l \tilde{m}_l^*\|_F^2 \\ \text{subject to} \quad & \det(\bar{h}\bar{m}^* + \tilde{b}_{i,\perp} u^* + v \tilde{c}_{i,\perp}^*) = 0, \end{aligned}$$

where $\bar{h}\bar{m}^*$ denotes a particular solution to the equation $y_i = \tilde{b}_i^* h m^* \tilde{c}_i$.

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

QCQP formulation

$$\begin{aligned} \min_{(u,v)} \quad & \|\bar{h}\bar{m}^* + \tilde{b}_{i,\perp} u^* + v \tilde{c}_{i,\perp}^* - \tilde{h}_l \tilde{m}_l^*\|_F^2 \\ \text{subject to} \quad & \det(\bar{h}\bar{m}^* + \tilde{b}_{i,\perp} u^* + v \tilde{c}_{i,\perp}^*) = 0, \end{aligned}$$

where $\bar{h}\bar{m}^*$ denotes a particular solution to the equation $y_i = \tilde{b}_i^* h m^* \tilde{c}_i$.

Find a global minimizer

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_i,$$

where $\mathcal{N}_i = \{\tilde{X} \in \mathbb{C}^{2\times 2} \mid \mathrm{rank}(\tilde{X}) = 1, y_i = \tilde{b}_i^* \tilde{X} \tilde{c}_i\}$.

We have tried three approaches

- Find all KKT points and find the global minimizer;
- QCQP formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_i,$$

where $\mathcal{N}_i = \{\tilde{X} \in \mathbb{C}^{2\times 2} \mid \mathrm{rank}(\tilde{X}) = 1, y_i = \tilde{b}_i^* \tilde{X} \tilde{c}_i\}$.

Most efficient in our experiments

Not guarantee to find a global minimizer

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_i,$$

where $\mathcal{N}_i = \{\tilde{X} \in \mathbb{C}^{2\times 2} \mid \mathrm{rank}(\tilde{X}) = 1, y_i = \tilde{b}_i^* \tilde{X} \tilde{c}_i\}$.

Manifold of $\mathcal{N}_i$:

- $\mathrm{T}_{\tilde{X}}\mathcal{N}_i = \{\sigma \tilde{h}\tilde{m}^* + h_p \tilde{m}^* + \tilde{h} m_p^*\} \bigcap \{\tilde{X} \in \mathbb{C}^{2\times 2} : 0 = \tilde{b}_i^* \tilde{X} \tilde{c}_i\}$
  $= \{\begin{bmatrix} \tilde{h} & \tilde{h}_\perp \end{bmatrix} \begin{bmatrix} \sigma & \alpha \\ \beta & 0 \end{bmatrix} \begin{bmatrix} \tilde{m}^* \\ \tilde{m}_\perp^* \end{bmatrix}\} \bigcap \{\tilde{X} \in \mathbb{C}^{2\times 2} : 0 = \tilde{b}_i^* \tilde{X} \tilde{c}_i\};$
- $\mathrm{N}_{\tilde{X}}\mathcal{N}_i = \{\alpha \tilde{h}_\perp \tilde{m}_\perp^* + \beta \tilde{b}_i \tilde{c}_i^*\} = \mathrm{span}\{\tilde{h}_\perp \tilde{m}_\perp^*, \tilde{b}_i \tilde{c}_i^*\};$
- $\mathrm{Proj}_{\tilde{X}}(G) = (I - QQ^*)G$, where Q is an orthonormal basis of $\mathrm{N}_{\tilde{X}}\mathcal{N}_i$;
- $\mathrm{grad}\, f(\tilde{X}) = \mathrm{Proj}_X(\nabla f(\tilde{X}))$, where $\nabla f(\tilde{X}) = 2(\tilde{X} - \tilde{h}_l \tilde{m}_l^*)$;
- $R(W) = \frac{\pi_1(W)}{\tilde{b}_i^* \pi_1(W) \tilde{c}_i} y_i$, where $\pi_1(W) = \sigma_1 u_1 v_1^*$;

- Initial point Z_0
  - projection $\tilde{X}_l = \tilde{h}_l \tilde{m}_l^*$ to the hyperplane gets $\tilde{X}_l + \alpha \tilde{b}_i \tilde{c}_i^*$, where $\alpha = \frac{y_i - \tilde{b}_i^* \tilde{X}_l \tilde{c}_i}{\|\tilde{b}_i^*\|_2^2 \|\tilde{c}_i^*\|_2^2}$;
  - $Z_0 = R(\tilde{X}_l + \alpha \tilde{b}_i \tilde{c}_i^*)$, i.e., truncated SVD for $\tilde{X}_l + \alpha \tilde{b}_i \tilde{c}_i^*$ and then rescale the resulting rank one;

Convergent results in existing blind deconvolution [ARR14, LLSW19, HH18]

- Good initialization $\xrightarrow{\text{high probability}}$ true solution;
- Conditions on B and C
  - C: complex Gaussian distribution;
  - $B^*B = I_K$ and $\|b_i\|^2 < \phi K/L, i = 1, \ldots, L$;

Convergent results in existing Kaczmarz methods

- Linear system: converges linearly [SV09];
- Phase retrieval: converges linearly to a neighborhood of the solution [Wei15];
- Weak conditions on the matrices (solution exists);

Convergent results in existing blind deconvolution [ARR14, LLSW19, HH18]

- Good initialization $\xrightarrow{\text{high probability}}$ true solution;
- Conditions on B and C
  - C: complex Gaussian distribution;
  - $B^*B = I_K$ and $\|b_i\|^2 < \phi K/L, i = 1, \ldots, L$;

Convergent results in existing Kaczmarz methods

- Linear system: converges linearly [SV09];
- Phase retrieval: converges linearly to a neighborhood of the solution [Wei15];
- Weak conditions on the matrices (solution exists);

Convergent results in existing blind deconvolution [ARR14, LLSW19, HH18]

- Good initialization $\xrightarrow{\text{high probability}}$ true solution;
- Conditions on B and C
  - C: complex Gaussian distribution;
  - $B^* B = I_K$ and $\|b_i\|^2 < \phi K/L, i = 1, \ldots, L$;

Convergent results in existing Kaczmarz methods

- Linear system: converges linearly [SV09];
- Phase retrieval: converges linearly to a neighborhood of the solution [Wei15];
- Weak conditions on the matrices (solution exists);

local convergence analysis with weak conditions

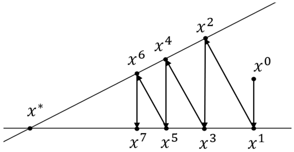


Figure 1: Diagram of Kaczmarz method for a linear system.

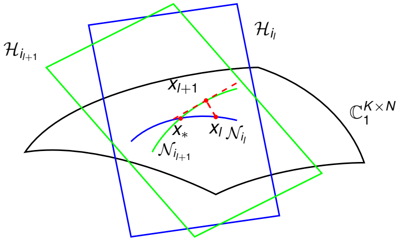


Figure 2: Diagram of Kaczmarz method for Blind deconvolution at l-th step.

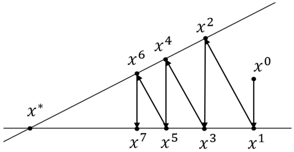


Figure 1: Diagram of Kaczmarz method for a linear system.

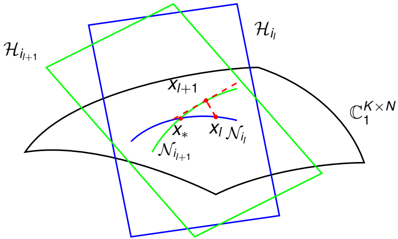


Figure 2: Diagram of Kaczmarz method for Blind deconvolution at l-th step.

The curvature near x_* plays an important role.

Lemma

Let $\mathcal{N}_i$ denote the manifold $\{X \in \mathbb{C}^{K\times N} \mid y = b_i^* X c_i, \mathrm{rank}(X) = 1\}$. For any $\delta > 0$, there exists $K_{\delta,i} > 0$ such that for any $\eta_X \in \mathrm{T}_X\mathcal{N}_i$ with $\|\eta_X\| = 1$, it holds that the geodesic γ_X satisfies

$$\gamma_X(t\eta_X) = X + t\eta_X + t^2 G(X, t, \eta_X), \forall |t| < \delta$$

and $\|G(X, t, \eta_X)\| \leq K_{\delta,i}$, where $G(X, t, \eta_X) \in \mathrm{N}_X\mathcal{N}_i$.

- a straight line + higher-order term;
- $K_{\delta,i}$ depends on the curvature of $\mathcal{N}_i$ near X;
- $K_{\delta,i} = 0$ in a Euclidean space;

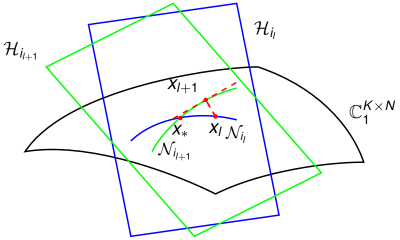


Figure 3: Diagram of Kaczmarz method for Blind deconvolution at l-th step.

$$\mathcal{N}_{i_l} \text{ from } X_* \text{ to } X_l: \quad \gamma_{X_*}(t\eta) = X_* + t\eta + t^2 G(X_*, t, \eta)$$

$$\mathcal{N}_{i_{l+1}}: \text{ along } \xi = P_{\mathrm{T}_{X_*}\mathcal{N}_{i_{l+1}}}\eta: \quad \gamma_{X_*}(t\xi) = X_* + t\xi + t^2 G(X_*, t, \xi)$$

where $\eta \in \mathrm{T}_{X_*}\mathcal{N}_{i_l}$, $\xi \in \mathrm{T}_{X_*}\mathcal{N}_{i_{l+1}}$, $G(X_*, t, \eta) \in \mathrm{N}_{X_*}\mathcal{N}_{i_l}$, $G(X_*, t, \xi) \in \mathrm{N}_{X_*}\mathcal{N}_{i_{l+1}}$, $\|G(X_*, t, \eta)\| \leq K_1$ and $\|G(X_*, t, \xi)\| \leq K_2$.

Theorem

For a sufficiently small $\delta > 0$, there exists $\rho > 0$ such that for any $\alpha \leq \rho$, it holds that

$$\|X_{l+1} - X_*\| \leq \frac{\langle \eta, \xi \rangle + \delta + \alpha(\langle \eta, \xi \rangle + \delta)^2 K}{1 - \alpha K} \|X_l - X_*\|,$$

where $K = \max\{K_1, K_2\}$.

Outline

- Blind deconvolution
- A review of Kaczmarz methods
- A Kaczmarz method for blind deconvolution
- Preliminary numerical results
- Summary and future work

- B and C are both Gaussian random matrices;
- h_* and m_* are Gaussian random vectors;
- y are noiseless, i.e., $y = Bh_* \odot \overline{Cm_*}$;
- stopping criterion $\|y - Bh \odot \overline{Cm}\|_2 / \|y\| \leq 10^{-4}$;

Numerical results

Efficiency

- $K = N = 100$;
- *RMSE* denotes the relative error $\frac{\|hm^T - h_* m_*^T\|}{\|h_*\|\|m_*\|}$;

	$L = 500$				
Algorithms	NCBT	NCBB	AMA	ROBB	RKM
nBh/nCm	225	79	157	83	60
RMSE	5.74_{-4}	5.57_{-4}	5.73_{-4}	5.71_{-4}	2.09_{-4}

	$L = 600$				
Algorithms	NCBT	NCBB	AMA	ROBB	RKM
nBh/nCm	137	66	109	59	33
RMSE	4.82_{-4}	1.78_{-4}	4.26_{-4}	4.65_{-4}	1.86_{-4}

- NCBT: gradient descent [LLSW19];
- NCBB: [LLSW19] with BB step size;
- AMA: alternating minimization in [HH18];
- ROBB: Riemannian SD in [HH18];
- RKM: proposed one;

Outline

- Blind deconvolution
- A review of Kaczmarz methods
- A Kaczmarz method for blind deconvolution
- Preliminary numerical results
- Summary and future work

Summary and future work

Summary

- Introduce blind deconvolution;
- Review Kaczmarz methods for linear system, phase retrieval, and blind deconvolution;
- Propose a Kaczmarz method for blind deconvolution;
- Local convergence analysis;
- Preliminary numerical results;

Future work

- Block Kaczmarz method;
- Systematic numerical experiments;

Thank you!

A block Kaczmarz method for blind deconvolution

Input: An initial point X_0, partition $T = \{\Gamma_1, \cdots, \Gamma_{N_b}\}$ of the row indices $\{1, \cdots, L\}$;

1: **for** $l = 0, 1, 2, \ldots,$ **do**
2: Select a block Γ_i and get $\mathcal{N}_{\Gamma_i} = \{X \in \mathbb{C}^{K \times N} : \mathrm{rank}(X) = 1, y_{\Gamma_i} = b^*_{\Gamma_i} X c_{\Gamma_i}\}$;
3: Compute $X_{l+1} = \arg\min_{X \in \mathcal{N}_{\Gamma_i}} \|X - X_l\|^2$;
4: **end for**

Equivalently reduce to $(1 + \frac{L}{N_b}) \times (1 + \frac{L}{N_b})$ subproblem;

Equivalently reduce to $(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})$ subproblem;

Solve

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_{\Gamma_i}=b_{\Gamma_i}^* X c_{\Gamma_i}} \|X - h_l m_l^*\|_F^2$$

by

$$\min_{\tilde{X}\in\mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})},\mathrm{rank}(\tilde{X})=1,y_{\Gamma_i}=\tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}} \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2$$

- QR decomposition: $[h_l\ b_{\Gamma_i}] = Q_1 R_1$ and $[m_l\ c_{\Gamma_i}] = Q_2 R_2$;
- $\tilde{h}_l = Q_1^* h_l$, $\tilde{m}_l = Q_2^* m_l$, $\tilde{b}_{\Gamma_i} = Q_1^* b_{\Gamma_i}$, and $\tilde{c}_{\Gamma_i} = Q_2^* c_{\Gamma_i}$;
- $\tilde{h}_l, \tilde{m}_l \in \mathbb{C}^{(1+\frac{L}{N_b})\times 1}$, $\tilde{b}_{\Gamma_i}, \tilde{c}_{\Gamma_i} \in \mathbb{C}^{(1+\frac{L}{N_b})\times\frac{L}{N_b}}$

Solve

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_{\Gamma_i}=b_{\Gamma_i}^* Xc_{\Gamma_i}} \|X - h_l m_l^*\|_F^2$$

by

$$\min_{\tilde{X}\in\mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})},\mathrm{rank}(\tilde{X})=1,y_{\Gamma_i}=\tilde{b}_{\Gamma_i}^* \tilde{X}\tilde{c}_{\Gamma_i}} \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2$$

Algorithm: Compute $X_{l+1} = \arg\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_{\Gamma_i}=b_{\Gamma_i}^* Xc_{\Gamma_i}} \|X - h_l m_l^*\|_F^2$

Input: $y_{\Gamma_i}, b_{\Gamma_i}, c_{\Gamma_i}, h_l, m_l$;

1: Compute QR decomposition $Q_1 R_1 = \begin{bmatrix} h_l & b_{\Gamma_i} \end{bmatrix}$, $Q_2 R_2 = \begin{bmatrix} m_l & c_{\Gamma_i} \end{bmatrix}$;

2: Compute $\tilde{h}_l \leftarrow Q_1^* h_l = R_1(:,1)$, $\tilde{m}_l \leftarrow Q_2^* m_l = R_2(:,1)$,
$\tilde{b}_{\Gamma_i} \leftarrow Q_1^* b_{\Gamma_i} = R_1(:,2:end)$, $\tilde{c}_{\Gamma_i} \leftarrow Q_2^* c_{\Gamma_i} = R_2(:,2:end)$;

3: Find $(\tilde{h}_{l+1}, \tilde{m}_{l+1}) = \underset{y_{\Gamma_i}=\tilde{b}_{\Gamma_i}^* hm^* \tilde{c}_{\Gamma_i}}{\mathrm{argmin}} \|hm^* - \tilde{h}_l \tilde{m}_l^*\|_{\mathrm{F}}^2$;

4: Compute $h_{l+1} \leftarrow Q_1 \tilde{h}_{l+1}$, $m_{l+1} \leftarrow Q_2 \tilde{m}_{l+1}$;

Solve

$$\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_{\Gamma_i}=b_{\Gamma_i}^*Xc_{\Gamma_i}} \|X-h_l m_l^*\|_F^2$$

by

$$\min_{\tilde{X}\in\mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})},\mathrm{rank}(\tilde{X})=1,y_{\Gamma_i}=\tilde{b}_{\Gamma_i}^*\tilde{X}\tilde{c}_{\Gamma_i}} \|\tilde{X}-\tilde{h}_l\tilde{m}_l^*\|_F^2$$

Algorithm: Compute $X_{l+1}=\arg\min_{X\in\mathbb{C}^{K\times N},\mathrm{rank}(X)=1,y_{\Gamma_i}=b_{\Gamma_i}^*Xc_{\Gamma_i}} \|X-h_l m_l^*\|_F^2$

Input: $y_{\Gamma_i}, b_{\Gamma_i}, c_{\Gamma_i}, h_l, m_l$;

1: Compute QR decomposition $Q_1R_1=\begin{bmatrix} h_l & b_{\Gamma_i}\end{bmatrix}$, $Q_2R_2=\begin{bmatrix} m_l & c_{\Gamma_i}\end{bmatrix}$;

2: Compute $\tilde{h}_l\leftarrow Q_1^*h_l=R_1(:,1)$, $\tilde{m}_l\leftarrow Q_2^*m_l=R_2(:,1)$,
$\tilde{b}_{\Gamma_i}\leftarrow Q_1^*b_{\Gamma_i}=R_1(:,2:end)$, $\tilde{c}_{\Gamma_i}\leftarrow Q_2^*c_{\Gamma_i}=R_2(:,2:end)$;

3: Find $(\tilde{h}_{l+1},\tilde{m}_{l+1})=\underset{y_{\Gamma_i}=\tilde{b}_{\Gamma_i}^*hm^*\tilde{c}_{\Gamma_i}}{\operatorname{argmin}} \|hm^*-\tilde{h}_l\tilde{m}_l^*\|_{\mathrm{F}}^2$;

4: Compute $h_{l+1}\leftarrow Q_1\tilde{h}_{l+1}$, $m_{l+1}\leftarrow Q_2\tilde{m}_{l+1}$;

We have tried two approaches

- A formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

We have tried two approaches

- A formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

A formulation

$$\begin{array}{ll} \min\limits_{(u,v)} & \|\bar{h}\bar{m}^* + \tilde{b}_{\Gamma_i,\perp} u^* + v \tilde{c}^*_{\Gamma_i,\perp} - \tilde{h}_l \tilde{m}^*_l\|_F^2 \\ \text{subject to} & \operatorname{rank}(\bar{h}\bar{m}^* + \tilde{b}_{\Gamma_i,\perp} u^* + v \tilde{c}^*_{\Gamma_i,\perp}) = 1, \end{array}$$

where $\bar{h}\bar{m}^*$ denotes a particular solution to the equation $y_i = \tilde{b}^*_{\Gamma_i} h m^* \tilde{c}_{\Gamma_i}$.

We have tried two approaches

- A formulation and find the global solution (Gurobi);
- Riemannian optimization formulation;

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_{\Gamma_i},$$

where $\mathcal{N}_{\Gamma_i} = \{\tilde{X} \in \mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})} \mid \mathrm{rank}(\tilde{X}) = 1, y_{\Gamma_i} = \tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}\}$.

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_{\Gamma_i},$$

where $\mathcal{N}_{\Gamma_i} = \{\tilde{X} \in \mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})} \mid \mathrm{rank}(\tilde{X}) = 1, y_{\Gamma_i} = \tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}\}$.

Manifold of $\mathcal{N}_{\Gamma_i}$:

- $\mathrm{T}_{\tilde{X}}\mathcal{N}_{\Gamma_i} = \{\sigma \tilde{h}\tilde{m}^* + h_p \tilde{m}^* + \tilde{h} m_p^*\} \bigcap \{\tilde{X} : 0 = \tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}\}$
$$= \{\begin{bmatrix} \tilde{h} & \tilde{h}_\perp \end{bmatrix} \begin{bmatrix} \sigma & \alpha \\ \beta & 0 \end{bmatrix} \begin{bmatrix} \tilde{m}^* \\ \tilde{m}_\perp^* \end{bmatrix}\} \bigcap \{\tilde{X} : 0 = \tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}\};$$
- $\mathrm{N}_{\tilde{X}}\mathcal{N}_{\Gamma_i} = span\{\tilde{h}_\perp \tilde{m}_\perp^*, \tilde{b}_{\Gamma_i}(:,1)\tilde{c}_{\Gamma_i}(:,1)^*, \ldots, \tilde{b}_{\Gamma_i}(:,\frac{L}{N_b})\tilde{c}_{\Gamma_i}(:,\frac{L}{N_b})^*\}$;
- $\mathrm{Proj}_{\tilde{X}}(G) = (I - QQ^*)G$, where Q is an orthonormal basis of $\mathrm{N}_{\tilde{X}}\mathcal{N}_{\Gamma_i}$;
- $\mathrm{grad}\, f(\tilde{X}) = \mathrm{Proj}_X(\nabla f(\tilde{X}))$, where $\nabla f(\tilde{X}) = 2(\tilde{X} - \tilde{h}_l \tilde{m}_l^*)$;

Riemannian optimization formulation

$$\min \|\tilde{X} - \tilde{h}_l \tilde{m}_l^*\|_F^2, \text{ subject to } \tilde{X} \in \mathcal{N}_{\Gamma_i},$$

where $\mathcal{N}_{\Gamma_i} = \{\tilde{X} \in \mathbb{C}^{(1+\frac{L}{N_b})\times(1+\frac{L}{N_b})} \mid \mathrm{rank}(\tilde{X}) = 1, y_{\Gamma_i} = \tilde{b}_{\Gamma_i}^* \tilde{X} \tilde{c}_{\Gamma_i}\}$.

Manifold of $\mathcal{N}_{\Gamma_i}$:

- $R(W)$
    - Caculate $q_0 r^* = \pi_1(W)$ and $QR = \tilde{b}_{\Gamma_i}$;
    - $\alpha_1 = y_{\Gamma_i,1} / \left(\tilde{b}_{\Gamma_i}(:,1)^* q_0 r^* \tilde{c}_{\Gamma_i}(:,1)\right)$;
    - **for** $k = 2, \ldots, \frac{L}{N_b}$

      $$\alpha_k = \frac{\left(y_{\Gamma_i,k} - \sum_{j=0}^{k-2} \alpha_{j+1} \tilde{b}_{\Gamma_i}(:,k) q_j r^* \tilde{c}_{\Gamma_i}(:,k)\right)}{\tilde{b}_{\Gamma_i}(:,k)^* q_{k-1} r^* \tilde{c}_{\Gamma_i}(:,k)};$$

      end for
    - $R(W) = (\sum_{k=1}^{\frac{L}{N_b}} \alpha_k q_{k-1}) r^*$;

References I

Ali Ahmed, Benjamin Recht, and Justin Romberg.
Blind deconvolution using convex programming.
IEEE Transactions on Information Theory, 60(3):1711–1732, 2014.

Frank Deutsch and Hein Hundal.
The rate of convergence for the method of alternating projections, ii.
Journal of Mathematical Analysis and Applications, 205(2):381–405, 1997.

A Galántai.
On the rate of convergence of the alternating projection method in finite dimensional spaces.
Journal of mathematical analysis and applications, 310(1):30–44, 2005.

Wen Huang and Paul Hand.
Blind deconvolution by a steepest descent algorithm on a quotient manifold.
SIAM Journal on Imaging Sciences, 11(4):2757–2785, 2018.

Meng Huang and Yang Wang.
Linear convergence of randomized kaczmarz method for solving complex-valued phaseless equations.
SIAM Journal on Imaging Sciences, 15(2):989–1016, 2022.

Halyun Jeong and C Sinan Güntürk.
Convergence of the randomized kaczmarz method for phase retrieval.
arXiv preprint arXiv:1706.10291, 2017.

S Kaczmarz.
Angenäherte auflösung von systemen linearer gleichungen (english translation by jason stockmann): Bulletin international de l'académie polonaise des sciences et des lettres.
1937.

References II

Gen Li, Yuantao Gu, and Yue M Lu.
Phase retrieval using iterative projections: Dynamics in the large systems limit.
In *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1114–1118. IEEE, 2015.

Yixian Liu.
An efficient method for non-convex blind deconvolution.
IEEE Access, 7:113663–113674, 2019.

Xiaodong Li, Shuyang Ling, Thomas Strohmer, and Ke Wei.
Rapid, robust, and reliable blind deconvolution via nonconvex optimization.
Applied and Computational Harmonic Analysis, 47(3):893–934, November 2019.

Patricia Römer, Frank Filbir, and Felix Krahmer.
On the randomized kaczmarz algorithm for phase retrieval.
In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pages 847–851. IEEE, 2021.

Martin Reiche and Peter Jung.
Deepinit phase retrieval.
arXiv preprint arXiv:2007.08214, 2020.

Thomas Strohmer and Roman Vershynin.
A randomized kaczmarz algorithm with exponential convergence.
Journal of Fourier Analysis and Applications, 15(2):262–278, 2009.

Yan Shuo Tan and Roman Vershynin.
Phase retrieval via randomized kaczmarz: theoretical guarantees.
Information and Inference: A Journal of the IMA, 8(1):97–123, 2019.

References III

Ke Wei.
Solving systems of phaseless equations via kaczmarz methods: A proof of concept study.
Inverse Problems, 31(12):125008, 2015.

Jinchao Xu and Ludmil Zikatanov.
The method of alternating projections and the method of subspace corrections in hilbert space.
Journal of the American Mathematical Society, 15(3):573–597, 2002.

Teng Zhang and Yu Feng.
Phase retrieval of complex-valued objects via a randomized kaczmarz method.
Information and Inference: A Journal of the IMA, 11(3):823–843, 2022.