Blind deconvolution by optimizing over a quotient manifold

Speaker: Wen Huang

Rice University

November 13, 2017

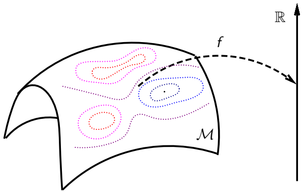
This is joint work with Paul Hand at Rice university

Riemannian Optimization

Problem: Given $f(x) : \mathcal{M} \to \mathbb{R}$,
solve

$$\min_{x \in \mathcal{M}} f(x)$$

where $\mathcal{M}$ is a Riemannian manifold.



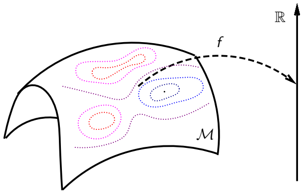
What is a Riemannian manifold? manifold + Riemannian metric

Riemannian Optimization

Problem: Given $f(x) : \mathcal{M} \to \mathbb{R}$, solve

$$\min_{x \in \mathcal{M}} f(x)$$

where $\mathcal{M}$ is a Riemannian manifold.



What is a Riemannian manifold? manifold + Riemannian metric

Examples of manifolds:

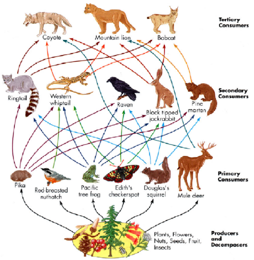
- $\{X \in \mathbb{R}^{n \times p} \mid X^T X = I_p\}$
- $\{X \in \mathbb{R}^{m \times n} \mid \mathrm{rank}(X) = p\}$
- $\{X \in \mathbb{R}^{n \times n} \mid X = X^T, X \succ 0\}$
- All p-dimensional linear subspaces of $\mathbb{R}^n$

Riemannian metric:

- Inner product on tangent spaces
- Define angles and lengths

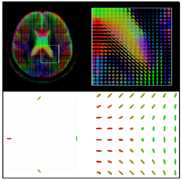
Applications

Network topology: Role model extraction



3D face recognition and reconstruction

Signature recognition

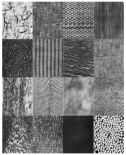


Diffusion tensor magnetic resonance imaging (MRI)

2D face recognition

Tracking and identifying people

Texture recognition



See [LKS+12, DBS+13, HGSA15, CS15, MHB+16, YHAG17]

ROPTLIB

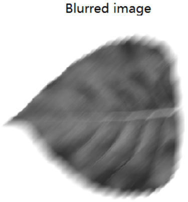
Riemannian manifold optimization library (ROPTLIB) is used to optimize a function on a manifold.

- Most state-of-the-art methods;
- Commonly-encountered manifolds;
- Written in C++;
- Interfaces with Matlab, Julia and R;
- BLAS and LAPACK;
- `www.math.fsu.edu/~whuang2/Indices/index_ROPTLIB.html`

Blind deconvolution

[Blind deconvolution]

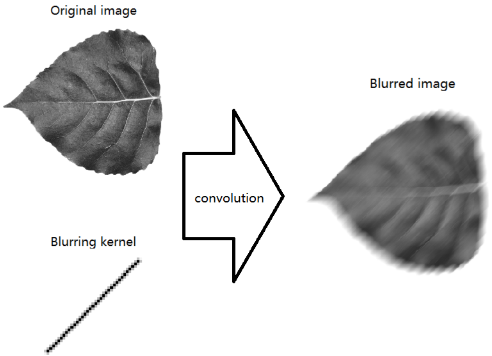
Blind deconvolution is to recover two unknown signals from their convolution.



Blind deconvolution

[Blind deconvolution]

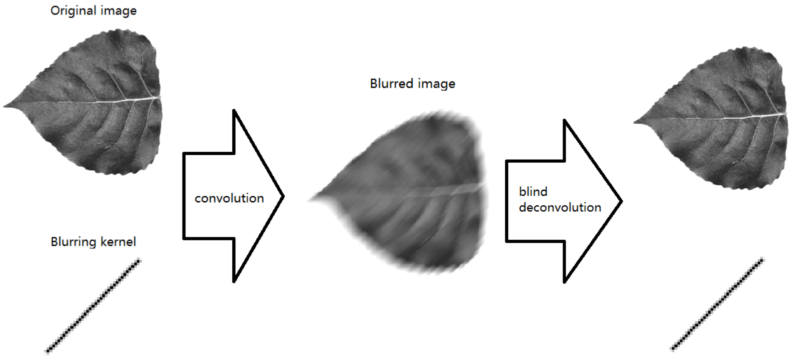
Blind deconvolution is to recover two unknown signals from their convolution.



Blind deconvolution

[Blind deconvolution]

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



Problem Statement

[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., $y_i = w_i x_i$.

Problem Statement

[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.

Problem Statement

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;

Problem Statement

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 $C \in \mathbb{C}^{L \times N}$;

Problem Statement

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 Statement

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

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

Related work

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

- Ahmed et al. [ARR14][1]
  - Convex problem:

$$\min_{X \in \mathbb{C}^{K \times N}} \|X\|_n, \text{ s. t. } y = \mathrm{diag}(BXC^*),$$

where $\|\cdot\|_n$ denotes the nuclear norm, and $X = hm^*$;

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

Related work

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

- Ahmed et al. [ARR14][1]
  - Convex problem:

$$\min_{X \in \mathbb{C}^{K \times N}} \|X\|_n, \text{ s. t. } y = \mathrm{diag}(BXC^*),$$

where $\|\cdot\|_n$ denotes the nuclear norm, and $X = hm^*$;

  - (Theoretical result): the unique minimizer $\xlongequal{\text{high probability}}$ the true solution;

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

Related work

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

- Ahmed et al. [ARR14][1]
  - Convex problem:

$$\min_{X \in \mathbb{C}^{K \times N}} \|X\|_n, \text{ s. t. } y = \mathrm{diag}(BXC^*),$$

    where $\|\cdot\|_n$ denotes the nuclear norm, and $X = hm^*$;
  - (Theoretical result): the unique minimizer $\overset{\text{high probability}}{=\!=\!=}$ the true solution;
  - The convex problem is 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

Related work

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

- Li et al. [LLSW16][2]
  - Nonconvex problem[3]:

$$\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2;$$

[2] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016

[3] The penalty in the cost function is not added for simplicity

Related work

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

- Li et al. [LLSW16][2]
  - Nonconvex problem[3]:

$$\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2;$$

  - (Theoretical result):
    - A good initialization
    - (Wirtinger flow method + a good initialization) $\xrightarrow{\text{high probability}}$ the true solution;

[2] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016

[3] The penalty in the cost function is not added for simplicity

Related work

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

- Li et al. [LLSW16][2]
  - Nonconvex problem[3]:

$$\min_{(h,m)\in\mathbb{C}^K\times\mathbb{C}^N} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2;$$

  - (Theoretical result):
    - A good initialization
    - (Wirtinger flow method + a good initialization) $\xrightarrow{\text{high probability}}$ the true solution;
  - Lower successful recovery probability than alternating minimization algorithm empirically.

[2] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016

[3] The penalty in the cost function is not added for simplicity

Manifold Approach

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

- The problem is defined on the set of rank-one matrices (denoted by $\mathbb{C}_1^{K \times N}$), neither $\mathbb{C}^{K \times N}$ nor $\mathbb{C}^K \times \mathbb{C}^N$; Why not work on the manifold directly?

Manifold Approach

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

- The problem is defined on the set of rank-one matrices (denoted by $\mathbb{C}_1^{K\times N}$), neither $\mathbb{C}^{K\times N}$ nor $\mathbb{C}^K \times \mathbb{C}^N$; Why not work on the manifold directly?

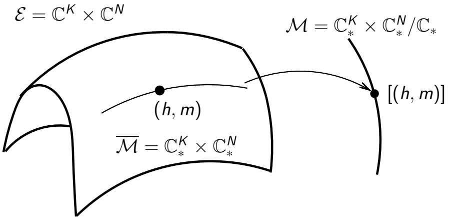
- Optimization on manifolds
  - A representation of $\mathbb{C}_1^{K\times N}$;
  - Representation of directions;
  - A Riemannian metric;
  - Riemannian gradient;
  - A Riemannian steepest descent method;

A Representation of $\mathbb{C}_1^{K \times N}$: $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_*$

- Given $X \in \mathbb{C}_1^{K \times N}$, there exist (h, m) such that $X = hm^*$;
- (h, m) is not unique;
- The equivalent class: $[(h, m)] = \{(ha, ma^{-*}) \mid a \neq 0\}$;
- Quotient manifold: $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* = \{[(h, m)] \mid (h, m) \in \mathbb{C}_*^K \times \mathbb{C}_*^N\}$

A Representation of $\mathbb{C}_1^{K\times N}$: $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_*$

- Given $X \in \mathbb{C}_1^{K\times N}$, there exist (h, m) such that $X = hm^*$;
- (h, m) is not unique;
- The equivalent class: $[(h, m)] = \{(ha, ma^{-*}) \mid a \neq 0\}$;
- Quotient manifold: $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* = \{[(h, m)] \mid (h, m) \in \mathbb{C}_*^K \times \mathbb{C}_*^N\}$



$$\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* \simeq \mathbb{C}_1^{K\times N}$$

A Representation of $\mathbb{C}_1^{K \times N}$: $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_*$

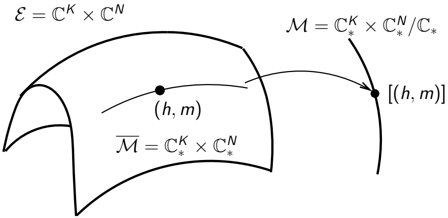
Cost function[4]

- Riemannian approach:

  $$f : \mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* \to \mathbb{R} : [(h, m)] \mapsto \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2.$$

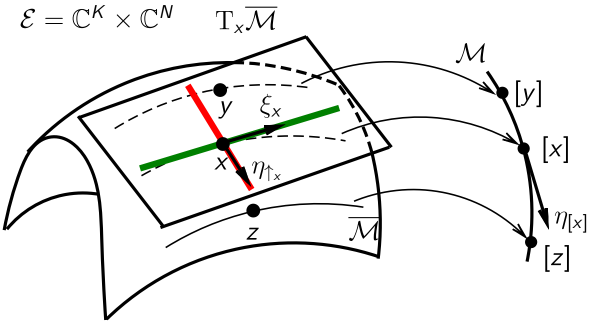
- Approach in [LLSW16]:

  $$\mathfrak{f} : \mathbb{C}^K \times \mathbb{C}^N \to \mathbb{R} : (h, m) \mapsto \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2.$$



[4] The penalty in the cost function is not added for simplicity.

Representation of directions on $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_*$



- x denotes (h, m);
- Green line: the tangent space of $[x]$;
- Red line (horizontal space at x): orthogonal to the green line;
- Horizontal space at x: a representation of the tangent space of $\mathcal{M}$ at $[x]$;

A Riemannian metric

Riemannian metric:

- Inner product on tangent spaces
- Define angles and lengths

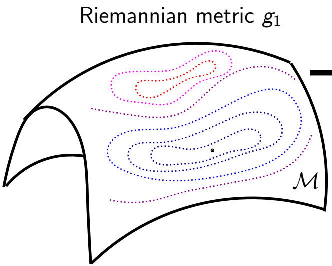


Figure: Changing metric may influence the difficulty of a problem.

A Riemannian metric

$$\min_{[(h,m)]} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$$

Idea for choosing a Riemannian metric

The block diagonal terms in the Euclidean Hessian are used to choose the Riemannian metric.

A Riemannian metric

$\min_{[(h,m)]} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$

Idea for choosing a Riemannian metric

The block diagonal terms in the Euclidean Hessian are used to choose the Riemannian metric.

- Let $\langle u, v \rangle_2 = \mathrm{Re}(\mathrm{trace}(u^* v))$:

$$\langle \eta_h, \mathrm{Hess}_h f[\xi_h] \rangle_2 = 2\langle \mathrm{diag}(B\eta_h m^* C^*), \mathrm{diag}(B\xi_h m^* C^*) \rangle_2 \approx 2\langle \eta_h m^*, \xi_h m^* \rangle_2$$
$$\langle \eta_m, \mathrm{Hess}_m f[\xi_m] \rangle_2 = 2\langle \mathrm{diag}(Bh\eta_m^* C^*), \mathrm{diag}(Bh\xi_m^* C^*) \rangle_2 \approx 2\langle h\eta_m^*, h\xi_m^* \rangle_2;$$

A Riemannian metric

$$\min_{[(h,m)]} \|y - \mathrm{diag}(Bhm^* C^*)\|_2^2$$

Idea for choosing a Riemannian metric

The block diagonal terms in the Euclidean Hessian are used to choose the Riemannian metric.

- Let $\langle u, v \rangle_2 = \mathrm{Re}(\mathrm{trace}(u^* v))$:

$$\langle \eta_h, \mathrm{Hess}_h f[\xi_h] \rangle_2 = 2 \langle \mathrm{diag}(B \eta_h m^* C^*), \mathrm{diag}(B \xi_h m^* C^*) \rangle_2 \approx 2 \langle \eta_h m^*, \xi_h m^* \rangle_2$$
$$\langle \eta_m, \mathrm{Hess}_m f[\xi_m] \rangle_2 = 2 \langle \mathrm{diag}(B h \eta_m^* C^*), \mathrm{diag}(B h \xi_m^* C^*) \rangle_2 \approx 2 \langle h \eta_m^*, h \xi_m^* \rangle_2;$$

- The Riemannian metric:

$$g\left(\eta_{[x]}, \xi_{[x]}\right) = \langle \eta_h, \xi_h m^* m \rangle_2 + \langle \eta_m^*, \xi_m^* h^* h \rangle_2;$$

Riemannian gradient

- $\mathrm{grad}\, \mathbf{f}([x]) \in \mathrm{T}_{[x]}\, \mathcal{M}$;
- $\mathrm{D}f([x])[\eta_{[x]}] = g(\mathrm{grad}\, f([x]), \eta_{[x]}), \quad \forall \eta_{[x]} \in \mathrm{T}_{[x]}\, \mathcal{M}$;
- A vector in horizontal space;
- Riemannian gradient:

$$(\mathrm{grad}\, f([(h, m)]))_{\uparrow_{(h,m)}} = Proj\left(\nabla_h f(h, m)(m^* m)^{-1}, \nabla_m f(h, m)(h^* h)^{-1}\right);$$

A Riemannian steepest descent method (RSD)

An implementation of a Riemannian steepest descent method[5]

0. Given (h_0, m_0), step size $\alpha > 0$, and set $k = 0$
1. $d_k = \|h_k\|_2 \|m_k\|_2$, $h_k \leftarrow \sqrt{d_k} \frac{h_k}{\|h_k\|_2}$; $m_k \leftarrow \sqrt{d_k} \frac{m_k}{\|m_k\|_2}$;
2. $(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha \left(\frac{\nabla_{h_k} f(h_k, m_k)}{d_k}, \frac{\nabla_{m_k} f(h_k, m_k)}{d_k} \right)$;
3. If not converge, goto Step 2.

[5] The penalty in the cost function is not added for simplicity

A Riemannian steepest descent method (RSD)

An implementation of a Riemannian steepest descent method[5]

0. Given (h_0, m_0), step size $\alpha > 0$, and set $k = 0$
1. $d_k = \|h_k\|_2 \|m_k\|_2$, $h_k \leftarrow \sqrt{d_k} \frac{h_k}{\|h_k\|_2}$; $m_k \leftarrow \sqrt{d_k} \frac{m_k}{\|m_k\|_2}$;
2. $(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha \left(\frac{\nabla_{h_k} f(h_k, m_k)}{d_k}, \frac{\nabla_{m_k} f(h_k, m_k)}{d_k} \right)$;
3. If not converge, goto Step 2.

Wirtinger flow Method in [LLSW16]

0. Given (h_0, m_0), step size $\alpha > 0$, and set $k = 0$
1. $(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha (\nabla_{h_k} f(h_k, m_k), \nabla_{m_k} f(h_k, m_k))$;
2. If not converge, goto Step 2.

[5] The penalty in the cost function is not added for simplicity

A Riemannian steepest descent method (RSD)

An implementation of a Riemannian steepest descent method[5]

0 Given (h_0, m_0), step size $\alpha > 0$, and set $k = 0$

1 $d_k = \|h_k\|_2 \|m_k\|_2$, $h_k \leftarrow \sqrt{d_k} \frac{h_k}{\|h_k\|_2}$; $m_k \leftarrow \sqrt{d_k} \frac{m_k}{\|m_k\|_2}$;

2 $(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha \left(\frac{\nabla_{h_k} f(h_k, m_k)}{d_k}, \frac{\nabla_{m_k} f(h_k, m_k)}{d_k} \right)$;

3 If not converge, goto Step 2.

Wirtinger flow Method in [LLSW16]

0 Given (h_0, m_0), step size $\alpha > 0$, and set $k = 0$

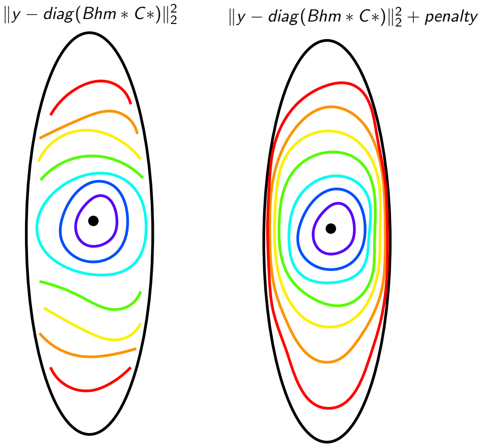
1 $(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha (\nabla_{h_k} f(h_k, m_k), \nabla_{m_k} f(h_k, m_k))$;

2 If not converge, goto Step 2.

[5] The penalty in the cost function is not added for simplicity

Penalty

- Ω: ellipsoid;
- Unique minimizer in Ω;
- Initial iterate in Ω;
- Importance of the penalty;



Penalty

- Riemannian approach:

$$\rho \sum_{i=1}^{L} G_0 \left(\frac{L|b_i^* h|^2 \|m\|_2^2}{8d^2\mu^2} \right)$$

- [LLSW16]:

$$\rho \left[G_0 \left(\frac{\|h\|_2^2}{2d} \right) + G_0 \left(\frac{\|m\|_2^2}{2d} \right) + \sum_{i=1}^{L} G_0 \left(\frac{L|b_i^* h|^2}{8d\mu^2} \right) \right]$$

- $G_0(t) = \max(t-1, 0)^2$, $[b_1 b_2 \dots b_L]^* = B$;

Penalty

- Riemannian approach:

$$\rho \sum_{i=1}^{L} G_0 \left(\frac{L |b_i^* h|^2 \|m\|_2^2}{8 d^2 \mu^2} \right)$$

- [LLSW16]:

$$\rho \left[G_0 \left(\frac{\|h\|_2^2}{2d} \right) + G_0 \left(\frac{\|m\|_2^2}{2d} \right) + \sum_{i=1}^{L} G_0 \left(\frac{L |b_i^* h|^2}{8 d \mu^2} \right) \right]$$

- $G_0(t) = \max(t-1, 0)^2$, $[b_1 b_2 \dots b_L]^* = B$;

Riemannian approach avoids the two terms.

Initialization

Initialization method [LLSW16]

- $(d, \tilde{h}_0, \tilde{m}_0)$: SVD of $B^* \operatorname{diag}(y) C$;
- Project $(\tilde{h}_0, \tilde{m}_0)$ to a neighborhood of the true solution;
- Initial iterate $[(h_0, m_0)]$;

Numerical Results

- Synthetic tests
  - Efficiency
  - Probability of successful recovery
- Image deblurring

Efficiency

Table: Comparisons of efficiency

	$L = 400, K = N = 50$			$L = 600, K = N = 50$		
Algorithms	[LLSW16]	[LWB13]	R-SD	[LLSW16]	[LWB13]	R-SD
nBh/nCm	351	718	208	162	294	122
$nFFT$	870	1436	518	401	588	303
$RMSE$	2.22_{-8}	3.67_{-8}	2.20_{-8}	1.48_{-8}	2.34_{-8}	1.42_{-8}

- An average of 100 random runs
- nBh/nCm: the numbers of Bh and Cm multiplication operations respectively
- nFFT: the number of Fourier transform
- RMSE: the relative error $\frac{\|hm^* - h_\sharp m_\sharp^*\|_F}{\|h_\sharp\|_2 \|m_\sharp\|_2}$

[LLSW16]: X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016
[LWB13]: K. Lee et. al., Near Optimal Compressed Sensing of a Class of Sparse Low-Rank Matrices via Sparse Power Factorization *preprint arXiv:1312.0525*, 2013

Probability of successful recovery

- Success if $\frac{\|hm^* - h_\sharp m_\sharp^*\|_F}{\|h_\sharp\|_2 \|m_\sharp\|_2} \leq 10^{-2}$

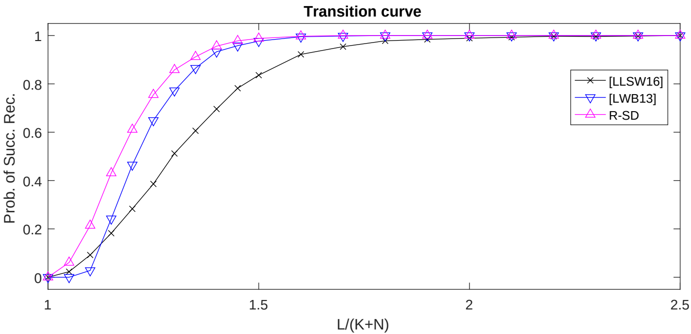
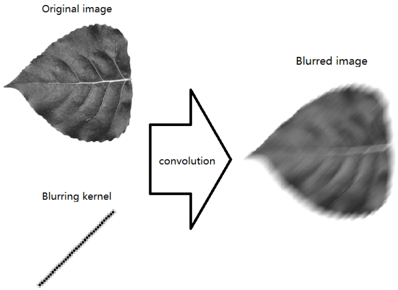


Figure: Empirical phase transition curves for 1000 random runs.

[LLSW16]: X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016
[LWB13]: K. Lee et. al., Near Optimal Compressed Sensing of a Class of Sparse Low-Rank Matrices via Sparse Power Factorization *preprint arXiv:1312.0525*, 2013

Image deblurring



- Original image [WBX^{+}07]: 1024-by-1024 pixels
- Motion blurring kernel (Matlab: fspecial('motion', 50, 45))

Image deblurring

What subspaces are the two unknown signals in?

- Image is approximately sparse in the Haar wavelet basis
- Support of the blurring kernel is learned from the blurred image

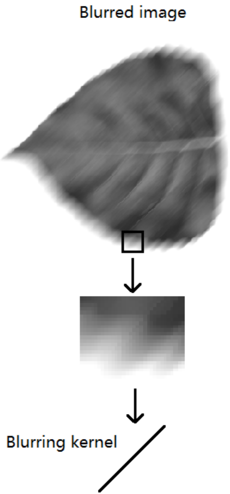


Image deblurring

What subspaces are the two unknown signals in?

- Image is approximately sparse in the Haar wavelet basis

  Use the blurred image to learn the dominated basis: **C**.

- Support of the blurring kernel is learned from the blurred image

  Suppose the support of the blurring kernel is known: **B**.

- $L = 1048576, K = 109,$
  $N = 5000, 20000, 80000$

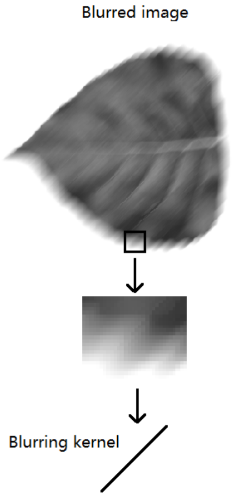


Image deblurring

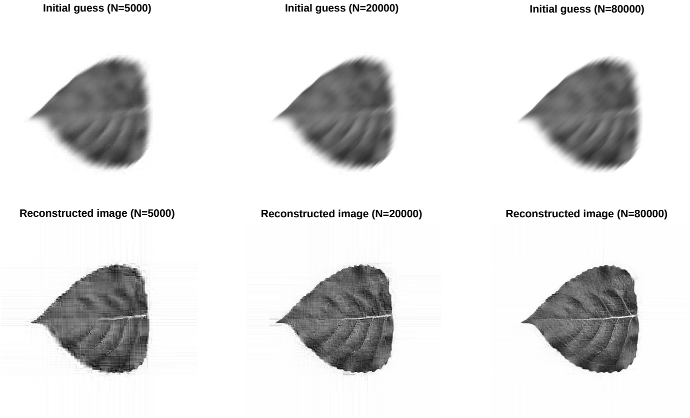


Figure: Initial guess by running power method for 50 iterations and the reconstructed image for $N = 5000, 20000$, and 80000.

Image deblurring

Table: Computational costs for multiple values of N on the image deblurring

N	nBh/nCm	$nFFT$	$relres$	$relerr$	t
5000	535	1330	4.8_{-3}	5.7_{-2}	170
20000	546	1358	2.1_{-3}	5.3_{-2}	173
80000	452	1124	8.0_{-4}	5.0_{-2}	144

- *relres*: $\|y - \mathrm{diag}(Bhm^* C^*)\|_2 / \|y\|_2$;
- *relerr*: $\left\| \mathbf{y}_o - \frac{\|\mathbf{y}\|}{\|\mathbf{y}_f\|} \mathbf{y}_f \right\| / \|\mathbf{y}_o\|$
- $\mathbf{y}_f$: the vector by reshaping the reconstructed image
- $\mathbf{y}_o$: the vector by reshaping the original image

Theoretical Results

Mathematical model:

- C is a complex Gaussian distribution; and
- B satisfies $B^*B = I_K$ and $\|b_i\|_2^2 \leq \phi \frac{K}{L}, i = 1, \ldots, L$ for some constant ϕ.

Theoretical Results

Initialization [LLSW16][6]

If $L \geq C_\gamma(\mu_h^2 + \sigma^2)\max(K, N)\log^2(L)/\varepsilon$, then with high probability, it holds that

$$[(h_0, m_0)] \in \Omega_{\frac{1}{2}\mu} \cap \Omega_{\frac{2}{5}\varepsilon},$$

where $\Omega_\mu = \{[(h, m)] \mid \sqrt{L}\|Bh\|_\infty \|m\|_2 \leq 4d_*\mu\}$, $\Omega_\varepsilon = \{[(h, m)] \mid \|hm^* - h_\sharp m_\sharp^*\|_F \leq \varepsilon d_*\}$, and $(h_\sharp, m_\sharp)$ is the true solution.

[6] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016

Theoretical Results

Initialization [LLSW16][6]

If $L \geq C_\gamma(\mu_h^2 + \sigma^2)\max(K, N)\log^2(L)/\varepsilon$, then with high probability, it holds that

$$[(h_0, m_0)] \in \Omega_{\frac{1}{2}\mu} \cap \Omega_{\frac{2}{5}\varepsilon},$$

where $\Omega_\mu = \{[(h, m)] \mid \sqrt{L}\|Bh\|_\infty \|m\|_2 \leq 4 d_* \mu\}$,
$\Omega_\varepsilon = \{[(h, m)] \mid \|hm^* - h_\sharp m_\sharp^*\|_F \leq \varepsilon d_*\}$, and $(h_\sharp, m_\sharp)$ is the true solution.

Large enough number of measurements $\Longrightarrow$ the initial point in a small neighborhood of the true solution.

[6] X. Li et. al., Rapid, robust, and reliable blind deconvolution via nonconvex optimization, *preprint arXiv:1606.04933*, 2016

Theoretical Results

Convergence analysis

Suppose $L \geq C_\gamma(\mu^2 + \sigma^2)\max(K, N)\log^2(L)/\varepsilon^2$ and the initialization $[(h_0, m_0)] \in \Omega_{\frac{1}{2}\mu} \cap \Omega_{\frac{2}{5}\varepsilon}$. Then with high probability, it holds that

$$\|h_k m_k^* - h_\sharp m_\sharp^*\|_F \leq \frac{2}{3}\left(1 - \frac{\alpha}{3000}\right)^{k/2} \varepsilon d_*,$$

where α is a small enough fixed step size.

Theoretical Results

Convergence analysis

Suppose $L \geq C_\gamma(\mu^2 + \sigma^2) \max(K, N) \log^2(L)/\varepsilon^2$ and the initialization $[(h_0, m_0)] \in \Omega_{\frac{1}{2}\mu} \cap \Omega_{\frac{2}{5}\varepsilon}$. Then with high probability, it holds that

$$\|h_k m_k^* - h_\sharp m_\sharp^*\|_F \leq \frac{2}{3}\left(1 - \frac{\alpha}{3000}\right)^{k/2} \varepsilon d_*,$$

where α is a small enough fixed step size.

i) Large enough number of measurements; ii) the initial point in a small neighborhood of the true solution $\Longrightarrow$ the Riemannian method converges linearly to the true solution.

Theoretical Results

Riemannian Hessian

Suppose $L \geq C_\gamma \max(K, \mu_h^2 N) \log^2(L)$. Then with high probability, it holds that

$$\frac{9d_*^2}{5} \leq \lambda_i \leq \frac{22d_*^2}{5}$$

for all i, where λ_i are eigenvalues of the Riemannian Hessian $\mathrm{Hess}\, f$ at the true solution.

Theoretical Results

Riemannian Hessian

Suppose $L \geq C_\gamma \max(K, \mu_h^2 N) \log^2(L)$. Then with high probability, it holds that

$$\frac{9d_*^2}{5} \leq \lambda_i \leq \frac{22d_*^2}{5}$$

for all i, where λ_i are eigenvalues of the Riemannian Hessian $\mathrm{Hess}\, f$ at the true solution.

The Riemannian Hessian $f \circ R$ is well-conditioned near the true solution.

Conclusion

Blind deconvolution by optimizing over a quotient manifold

- A Riemannian steepest descent method
- Simple implementation
- Recovery guarantee
- Superior numerical performance

Thank you

Thank you!

References I

A. Ahmed, B. Recht, and J. Romberg.
Blind deconvolution using convex programming.
IEEE Transactions on Information Theory, 60(3):1711–1732, March 2014.

A. Cherian and S. Sra.
Riemannian dictionary learning and sparse coding for positive definite matrices.
CoRR, abs/1507.02772, 2015.

H. Drira, B. Ben Amor, A. Srivastava, M. Daoudi, and R. Slama.
3D face recognition under expressions, occlusions, and pose variations.
Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(9):2270–2283, 2013.

Wen Huang, K. A. Gallivan, Anuj Srivastava, and P.-A. Absil.
Riemannian optimization for registration of curves in elastic shape analysis.
Journal of Mathematical Imaging and Vision, 54(3):320–343, 2015.
DOI:10.1007/s10851-015-0606-8.

H. Laga, S. Kurtek, A. Srivastava, M. Golzarian, and S. J. Miklavcic.
A Riemannian elastic metric for shape-based plant leaf classification.
2012 International Conference on Digital Image Computing Techniques and Applications (DICTA), pages 1–7, December 2012.
doi:10.1109/DICTA.2012.6411702.

Xiaodong Li, Shuyang Ling, Thomas Strohmer, and Ke Wei.
Rapid, robust, and reliable blind deconvolution via nonconvex optimization.
CoRR, abs/1606.04933, 2016.

References II

K. Lee, Y. Wu, and Y. Bresler.
Near Optimal Compressed Sensing of a Class of Sparse Low-Rank Matrices via Sparse Power Factorization.
pages 1–80, 2013.

Melissa Marchand, Wen Huang, Arnaud Browet, Paul Van Dooren, and Kyle A. Gallivan.
A riemannian optimization approach for role model extraction.
In *Proceedings of the 22nd International Symposium on Mathematical Theory of Networks and Systems*, pages 58–64, 2016.

S. G. Wu, F. S. Bao, E. Y. Xu, Y.-X. Wang, Y.-F. Chang, and Q.-L. Xiang.
A leaf recognition algorithm for plant classification using probabilistic neural network.
2007 IEEE International Symposium on Signal Processing and Information Technology, pages 11–16, 2007.
arXiv:0707.4289v1.

Xinru Yuan, Wen Huang, P.-A. Absil, and K. A. Gallivan.
A Riemannian quasi-newton method for computing the Karcher mean of symmetric positive definite matrices.
Technical Report FSU17-02, Florida State University, 2017.