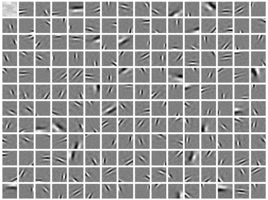
Computational Principles for High-dim Data Analysis

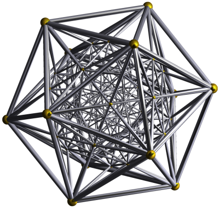
(Lecture Sixteen)

Yi Ma

UC Berkeley

October 26, 2021





Nonconvex Methods for Low-Dimensional Models
Dictionary Learning via ℓ^4 Maximization

Complete Dictionary Learning

Assumes data $\boldsymbol{Y}$ is generated by an orthogonal complete dictionary $\boldsymbol{D}_o$ and sparse coefficients $\boldsymbol{X}_o$:

$$\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o,$$

where $\boldsymbol{X}_o$ follows a Bernoulli Gaussian model:

$$\boldsymbol{X}_o = \boldsymbol{\Omega} \circ \boldsymbol{G}^{1}, \quad \Omega_{i,j} \sim_{iid} \mathrm{Ber}(\theta), G_{i,j} \sim_{iid} \mathcal{N}(0,1).$$

Reduced to find the sparsest direction in a subspace:

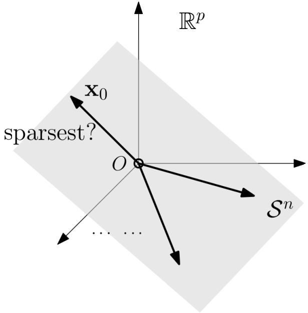
1. $\boldsymbol{D}_o$ is complete $\implies \boxed{\mathrm{row}(\boldsymbol{Y}) = \mathrm{row}(\boldsymbol{X}_o)}$
2. Rows of $\boldsymbol{X}_o$ form *a sparse basis* of $\mathrm{row}(\boldsymbol{Y})$.
3. Find $\boldsymbol{x}_1$, *the sparsest vector* in the subspace $\mathrm{row}(\boldsymbol{Y})$.
4. Find $\boldsymbol{x}_i$, *the sparsest vector* in $\mathrm{row}(\boldsymbol{Y}) \backslash \{\boldsymbol{x}_1, , \dots, \boldsymbol{x}_{i-1}\}$.
5. Recover $\boldsymbol{D}_o$ by: $\boldsymbol{D}_o = \boldsymbol{Y}\boldsymbol{X}_o^*(\boldsymbol{X}_o\boldsymbol{X}_o^*)^{-1}$.

[1] $\circ$ denote element-wise product: $\forall \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times m}$, $\{\boldsymbol{A}\circ\boldsymbol{B}\}_{i,j} = a_{i,j}b_{i,j}$

Complete Dictionary Learning – Prior Arts

Finding the sparsest vector in $\mathrm{row}(\boldsymbol{Y})$ can be naïvely formulated as

$$\min_{\boldsymbol{q}} \|\boldsymbol{q}^*\boldsymbol{Y}\|_0\,, \quad \text{s.t.} \quad \boldsymbol{q} \neq \boldsymbol{0}.$$



Or minimize the ℓ^1 norm on a sphere [SQW17, BJS18]:

$$\min_{\boldsymbol{q}} \|\boldsymbol{q}^*\boldsymbol{Y}\|_1\,, \quad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1.$$

Or maximize the ℓ^4 norm:

$$\max_{\boldsymbol{q}} \|\boldsymbol{q}^*\boldsymbol{Y}\|_4^4\,, \quad \text{s.t.} \quad \|\boldsymbol{q}\|_2 = 1.$$

Solving the same optimization n times (high computation cost)!

Intuition for ℓ^4 Norm Maximization

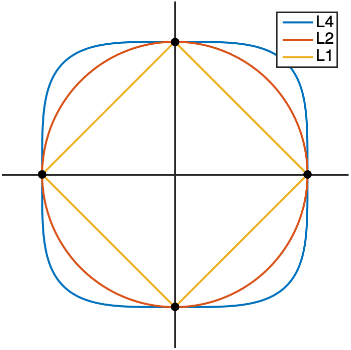


Figure: ℓ^1-,ℓ^2-, and ℓ^4-spheres in $\mathbb{R}^2$

Minimizing ℓ^1 norm or maximizing ℓ^4 norm both promote sparsity or spikiness:

$$\arg\min_{\boldsymbol{q}\in\mathbb{S}^n} \|\boldsymbol{q}\|_1 \quad \Leftrightarrow \quad \arg\min_{\boldsymbol{q}\in\mathbb{S}^n} \|\boldsymbol{q}\|_0 .$$

$$\arg\max_{\boldsymbol{q}\in\mathbb{S}^n} \|\boldsymbol{q}\|_4 \quad \Leftrightarrow \quad \arg\min_{\boldsymbol{q}\in\mathbb{S}^n} \|\boldsymbol{q}\|_0 .$$

Solving the same optimization n times (high computation cost)!

Intuition for ℓ^4 Norm Maximization [ZYL+19]

Consider finding the whole dictionary by the following nonconvex program:

$$\max_{\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})} f(\boldsymbol{A}) = \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \tag{1}$$

which is equivalent to

$$\max_{\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})} \|\boldsymbol{X}\|_4^4, \quad \text{s.t.} \quad \boldsymbol{Y} = \boldsymbol{A}^* \boldsymbol{X}, \tag{2}$$

where maximizing ℓ^4 norm with spherical constraints is promoting "spikiness" [ZKW18].

Related Works of ℓ^4 Norm

- Spherical Harmonic Analysis [SW81, Lu87].
- Independent Component Analysis (ICA) [HO97, HO00]
- Sum of Square (SoS) [BKS15, MSS16, SS17]
- Blind Deconvolution [ZKW18, LB18]

Main Results I

Relation to a Deterministic Objective

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} \mathsf{BG}(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. Then $\forall \boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$, the expectation of $\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$ is determined by function over $\mathsf{O}(n;\mathbb{R})$:

$$\frac{1}{3p\theta}\mathbb{E}_{\boldsymbol{X}_o}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4 = (1-\theta)\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 + \theta n. \tag{3}$$

Main Results I

Relation to a Deterministic Objective

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} \mathsf{BG}(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. Then $\forall \boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$, the expectation of $\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$ is determined by function over $\mathsf{O}(n;\mathbb{R})$:

$$\frac{1}{3p\theta}\mathbb{E}_{\boldsymbol{X}_o}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4 = (1-\theta)\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 + \theta n. \tag{3}$$

Global Maxima of the Deterministic Objective

$$\boldsymbol{W}_\star \in \arg\max_{\boldsymbol{W} \in \mathsf{O}(n;\mathbb{R})} \|\boldsymbol{W}\|_4^4 \iff \boldsymbol{W}_\star \in \mathsf{SP}(n) \tag{4}$$

Main Results I

Relation to a Deterministic Objective

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} \mathsf{BG}(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. Then $\forall \boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$, the expectation of $\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$ is determined by function over $\mathsf{O}(n;\mathbb{R})$:

$$\frac{1}{3p\theta}\mathbb{E}_{\boldsymbol{X}_o}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4 = (1-\theta)\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 + \theta n. \tag{3}$$

Global Maxima of the Deterministic Objective

$$\boldsymbol{W}_\star \in \underset{\boldsymbol{W}\in\mathsf{O}(n;\mathbb{R})}{\arg\max}\|\boldsymbol{W}\|_4^4 \iff \boldsymbol{W}_\star \in \mathsf{SP}(n) \tag{4}$$

Global maxima of $\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4$ are the correct dictionaries (up to signed permutation)!

Main Results II

Correctness of Global Optimal

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} \mathsf{BG}(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. Suppose $\hat{\boldsymbol{A}}_\star$ is a global maximizer of optimization:

$$\max_{\boldsymbol{A}} \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \quad \text{s.t.} \quad \boldsymbol{A} \in \mathsf{O}(n;\mathbb{R}), \tag{5}$$

then for any $\varepsilon \in [0,1]$, there exists a signed permutation matrix $\boldsymbol{P} \in \mathsf{SP}(n)$, such that $\frac{1}{n}\left\|\hat{\boldsymbol{A}}_\star^* - \boldsymbol{D}_o\boldsymbol{P}\right\|_F^2 \leq C\varepsilon$, with probability at least $1 - \frac{1}{p}$, when $p = \Omega(\theta n^2 \ln n / \varepsilon^2)$, for a constant $C > \frac{4}{3\theta(1-\theta)}$.

Main Results II

Correctness of Global Optimal

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid}$ $\mathrm{BG}(\theta)$, $\boldsymbol{D}_o \in \mathrm{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. Suppose $\hat{\boldsymbol{A}}_\star$ is a global maximizer of optimization:

$$\max_{\boldsymbol{A}} \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \quad \text{s.t.} \quad \boldsymbol{A} \in \mathrm{O}(n;\mathbb{R}), \tag{5}$$

then for any $\varepsilon \in [0,1]$, there exists a signed permutation matrix $\boldsymbol{P} \in \mathrm{SP}(n)$, such that $\frac{1}{n}\left\|\hat{\boldsymbol{A}}_\star^* - \boldsymbol{D}_o\boldsymbol{P}\right\|_F^2 \le C\varepsilon$, with probability at least $1 - \frac{1}{p}$, when $p = \Omega(\theta n^2 \ln n / \varepsilon^2)$, for a constant $C > \frac{4}{3\theta(1-\theta)}$.

With nearly minimal # samples, w.h.p., global maxima of $\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$ are arbitrarily close to the correct dictionary!

Optimization Algorithm

The program:

$$\max_{\boldsymbol{A}} f(\boldsymbol{A}) \doteq \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \quad \text{s.t.} \quad \boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$$

seems to be the worst case for optimization:

- **concave objective;**
- **geometric constraints;**
- **very high dimensional.**

Try projected (Riemannian) gradient descent anyway:

$$\boldsymbol{A}_{t+1} = \mathcal{P}_{O(n)}[\boldsymbol{A}_t + \alpha \nabla f(\boldsymbol{A}_t)] = \mathcal{P}_{O(n)}[\boldsymbol{A}_t + \alpha \underbrace{4(\boldsymbol{A}_t\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^*}_{\partial \boldsymbol{A}_t}].$$

A happy accident: observed that this converges faster as $\alpha \to \infty$!

(**Why?** something to do with power iteration... later...)

The MSP Algorithm I

We propose a novel algorithm, with Matching, Stretching (or Sparsifying) and Projection (MSP) to maximize $\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$:

Algorithm 1 MSP Algorithm on ℓ^4 Dictionary Learning

1: **Initialize** $\boldsymbol{A}_0 \in \mathsf{O}(n, \mathbb{R})$ ▷ Initialize $\boldsymbol{A}_0$ for iteration
2: **for** $t = 0, 1, ...$
3: $\quad \partial \boldsymbol{A}_t = 4(\boldsymbol{A}_t \boldsymbol{Y})^{\circ 3} \boldsymbol{Y}^*$ ▷ Matching and Stretching[2]
4: $\quad \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* = \mathsf{svd}(\partial \boldsymbol{A}_t)$
5: $\quad \boldsymbol{A}_{t+1} = \boldsymbol{U}\boldsymbol{V}^*$ ▷ Project $\boldsymbol{A}$ onto orthogonal group
6: **end for**
7: **Output** $\boldsymbol{A}_{t+1}, \|\boldsymbol{A}_{t+1}\boldsymbol{Y}\|_4^4 / 3np\theta$, $\|\boldsymbol{A}_{t+1}\boldsymbol{D}_o\|_4^4 / n$

[2] $\nabla_{\boldsymbol{A}} \|\boldsymbol{A}\boldsymbol{Y}\|_4^4 = 4(\boldsymbol{A}\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^*$

A Few Interpretations

NOT Gradient Descent!

"Fixed point" interpretation:

$$\boldsymbol{A}_{t+1} = \mathcal{P}_{O(n)}[\partial \boldsymbol{A}_t] = \mathcal{P}_{O(n)}[(\boldsymbol{A}_t \boldsymbol{Y})^{\circ 3} \boldsymbol{Y}^*].$$

"Deep learning" interpretation: $\delta \boldsymbol{A}_{t+1} = \boldsymbol{A}_{t+1} \boldsymbol{A}_t^*$ and $\boldsymbol{Z}_t = \boldsymbol{A}_t \boldsymbol{Y}$,

$$\delta \boldsymbol{A}_{t+1} = \mathcal{P}_{O(n)}[(\boldsymbol{Z}_t)^{\circ 3} \boldsymbol{Z}_t^*], \qquad \boldsymbol{X} \leftarrow \underbrace{\delta \boldsymbol{A}_{t+1} \delta \boldsymbol{A}_t \dots \delta \boldsymbol{A}_1}_{\text{forward constructed layers!}} \boldsymbol{Y}.$$

"Stochastic batch" variation:

$$\delta \boldsymbol{A}_{t+1} = \mathcal{P}_{O(n)}[(\tilde{\boldsymbol{Z}}_t)^{\circ 3} \tilde{\boldsymbol{Z}}_t^*], \quad \tilde{\boldsymbol{Z}}_t \subseteq \boldsymbol{Z}_t.$$

The MSP Algorithm II

Since $\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4$ has a linear relation with $\frac{1}{np}\mathbb{E}_{\boldsymbol{X}_o}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4$, a similar algorithm also can be applied to maximize $\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4$:

Algorithm 2 MSP Algorithm on ℓ^4 over Orthogonal Group

1: **Initialize** $\boldsymbol{A}_0 \in \mathsf{O}(n, \mathbb{R})$ ▷ Initialize $\boldsymbol{A}_0$ for iteration
2: **for** $t = 0, 1, ...$
3: $\quad \partial \boldsymbol{A}_t = 4(\boldsymbol{A}_t\boldsymbol{D}_o)^{\circ 3}\boldsymbol{D}_o^*$ ▷ Matching and Stretching
4: $\quad \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* = \mathsf{svd}(\partial \boldsymbol{A}_t)$
5: $\quad \boldsymbol{A}_{t+1} = \boldsymbol{U}\boldsymbol{V}^*$ ▷ Project $\boldsymbol{A}$ onto orthogonal group
6: **end for**
7: **Output** $\boldsymbol{A}_{t+1}$, $\|\boldsymbol{A}_{t+1}\boldsymbol{D}_o\|_4^4 / n$

One Run of the MSP Algorithm

$$
\begin{array}{rlcl}
 & \boldsymbol{A}_0 = \begin{pmatrix} -0.8249 & 0.3820 & -0.4168 \\ -0.5240 & -0.2398 & 0.8173 \\ -0.2122 & -0.8925 & -0.3979 \end{pmatrix} & \xrightarrow{\textit{stretching}} & \boldsymbol{A}_0^{\circ 3} = \begin{pmatrix} -0.5613 & 0.0557 & -0.0724 \\ -0.1439 & -0.0138 & 0.5459 \\ -0.0096 & -0.7109 & -0.0630 \end{pmatrix} \\
\xrightarrow{\textit{projection}} & \boldsymbol{A}_1 = \begin{pmatrix} -0.9795 & 0.0621 & -0.1917 \\ -0.1953 & -0.0594 & 0.9789 \\ -0.0494 & -0.9963 & -0.0703 \end{pmatrix} & \xrightarrow{\textit{stretching}} & \boldsymbol{A}_1^{\circ 3} = \begin{pmatrix} -0.9397 & 0.0002 & -0.0070 \\ -0.0075 & -0.0002 & 0.9381 \\ -0.0001 & -0.9889 & -0.0003 \end{pmatrix} \\
\xrightarrow{\textit{projection}} & \boldsymbol{A}_2 = \begin{pmatrix} -1.0000 & 0.0002 & -0.0077 \\ -0.0077 & -0.0003 & 1.000 \\ -0.0002 & -1.0000 & -0.0003 \end{pmatrix} & \xrightarrow{\textit{stretching}} & \boldsymbol{A}_2^{\circ 3} = \begin{pmatrix} -0.9999 & 0.0000 & -0.0000 \\ -0.0000 & -0.0000 & 0.9999 \\ -0.0000 & -1.0000 & -0.0000 \end{pmatrix} \\
\xrightarrow{\textit{projection}} & \boldsymbol{A}_3 = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} & \xrightarrow{\textit{output}} & \boldsymbol{A}_3^{\circ 3} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}.
\end{array}
$$

Figure: One run of the MSP algorithm for maximizing $\|\boldsymbol{A}\boldsymbol{D}_o\|_4^4$ over orthogonal group $\mathsf{O}(3)$ with $\boldsymbol{D}_o = \boldsymbol{I}$.

Convergence Guarantee of the MSP Algorithm

Theorem (Local Convergence of the MSP Algorithm)

Given an orthogonal matrix $\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$, let $\boldsymbol{A}'$ denote the output of the MSP Algorithm 2 after one iteration: $\boldsymbol{A}' = \boldsymbol{U}\boldsymbol{V}^*$, where $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* =$ SVD$(\boldsymbol{A}^{\circ 3})$. If $\|\boldsymbol{A}-\boldsymbol{I}\|_F^2 = \varepsilon$, for $\varepsilon < 0.579$, then we have $\|\boldsymbol{A}'-\boldsymbol{I}\|_F^2 < \|\boldsymbol{A}-\boldsymbol{I}\|_F^2$ and $\|\boldsymbol{A}'-\boldsymbol{I}\|_F^2 < O(\varepsilon^3)$.

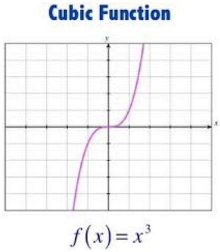


Figure: Cubic Function from ℓ^4.

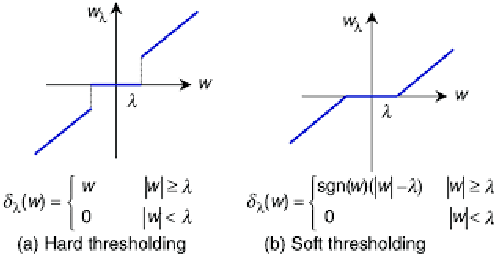


Figure: Thresholding from ℓ^1.

Convergence Guarantee of the MSP Algorithm

Theorem (Local Convergence of the MSP Algorithm)

Given an orthogonal matrix $\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$*, let* $\boldsymbol{A}'$ *denote the output of the MSP Algorithm 2 after one iteration:* $\boldsymbol{A}' = \boldsymbol{U}\boldsymbol{V}^*$*, where* $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* =$ *SVD*$(\boldsymbol{A}^{\circ 3})$*. If* $\|\boldsymbol{A}-\boldsymbol{I}\|_F^2 = \varepsilon$*, for* $\varepsilon < 0.579$*, then we have* $\|\boldsymbol{A}'-\boldsymbol{I}\|_F^2 < \|\boldsymbol{A}-\boldsymbol{I}\|_F^2$ *and* $\|\boldsymbol{A}'-\boldsymbol{I}\|_F^2 < O(\varepsilon^3)$*.*

Generalization to all Signed Permutation Matrices

The Identity can be generalized to any signed permutation matrix!

MSP algorithm in Maximizing $\|\boldsymbol{AY}\|_4^4$

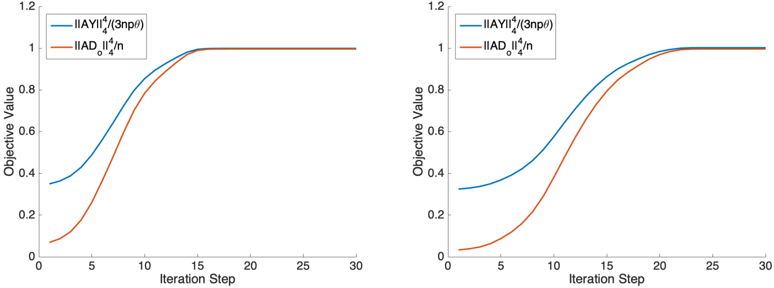


Figure: The value of $\frac{1}{3np\theta}\|\boldsymbol{AY}\|_4^4$ and $\frac{1}{n}\|\boldsymbol{AD}_o\|_4^4$ in two experiments with different settings: left: $n = 50, p = 20000, \theta = 0.3$, right: $n = 100, p = 40000, \theta = 0.3$. **The MSP algorithm converges quickly and smoothly with dozens of iterations.**

MSP algorithm in Maximizing $\|\boldsymbol{AY}\|_4^4$

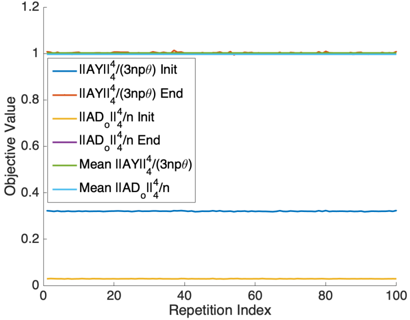


Figure: Initial value and final value of $\frac{1}{3np\theta}\|\boldsymbol{AY}\|_4^4$ and $\frac{1}{n}\|\boldsymbol{AD}_o\|_4^4$ for dictionary learning, with $n = 100, p = 40000$, $\theta = 0.3$, left: with initial values; right: without initial values. **All 100 trials converge to the global optima within statistical errors.**

Phase Transition of the MSP algorithm

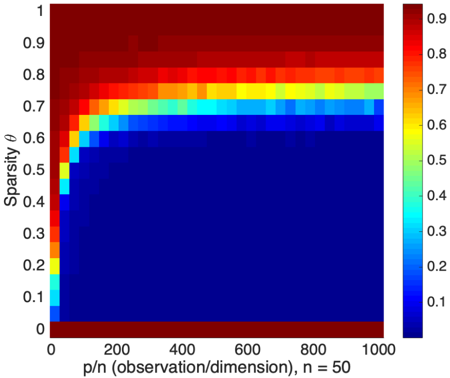
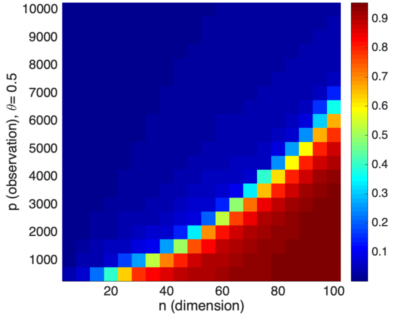


Figure: Phase transition plot of average normalized error $\left|1 - \|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 / n\right|$ for 10 trials of MSP algorithm 1 with $n = 50$. Red area indicates large error and blue area small error. Plot shows results for varying p versus θ. **The algorithm successes even when θ is up to 0.6!**

Phase Transition of the MSP algorithm



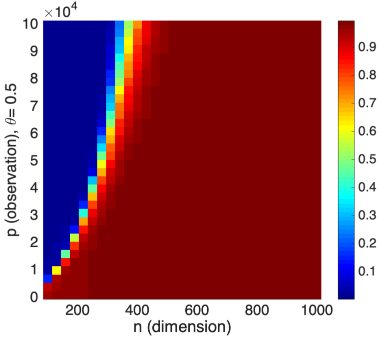


Figure: Phase transition plot of average normalized error $\left|1 - \|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 / n\right|$ for 10 trials of MSP algorithm 1 with $\theta = 0.5$. Red area indicates large error and blue area small error, left: n from 10 to 100 and p from 10^3 to 10^4, right: changing n from 100 to 10^3 and p from 10^4 to 10^5. **The number of samples p needed is quadratic in n.**

Optimal Choice of ℓ^{2k} Norm

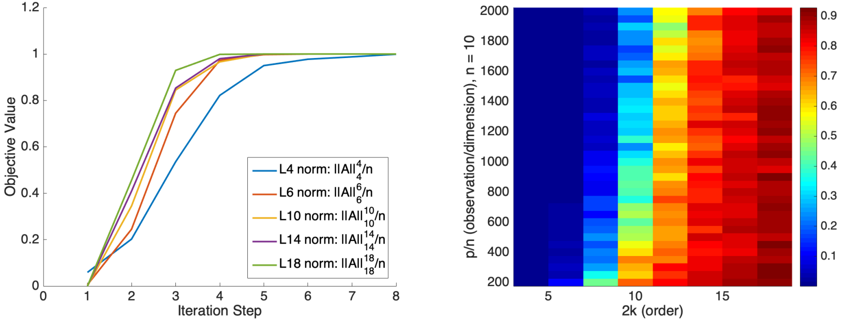


Figure: Experiments with different ℓ^{2k} norm. Left: Maximizing $\|\boldsymbol{A}\|_{2k}^{2k}$ for different order k. Right: Average normalized error of $\left|1 - \|\boldsymbol{A}\boldsymbol{D}_o\|_{2k}^{2k}/n\right|$ for maximizing $\|\boldsymbol{A}\boldsymbol{Y}\|_{2k}^{2k}$ for 20 trials, with $n = 10$, varying k and p. ℓ^4 **strikes a good balance between convergence and concentration.**

Comparison with the State of the Art

	KSVD		Subgradient		MSP (Ours)	
Trials	Error	Time	Error	Time	Error	Time
(a)	12.35%	51.2s	0.27%	35.6s	**0.34%**	**0.4s**
(b)	8.63%	244.4s	0.28%	354.9s	**0.34%**	**1.5s**
(c)	6.15%	684.9s	1.28%	6924.6s	**0.35%**	**7.6s**
(d)	8.61%	1042.3s	N/A	> 12h	**0.35%**	**48.0s**
(e)	13.07%	5401.9s	N/A	> 12h	**0.35%**	**374.2s**

Table: Comparison experiments with KSVD [AEB+06] and Subgradient method [BJS18] in different trials of dictionary learning: (a) $n = 25, p = 1 \times 10^4, \theta = 0.3$; (b) $n = 50, p = 2 \times 10^4, \theta = 0.3$; (c) $n = 100, p = 4 \times 10^4, \theta = 0.3$; (d) $n = 200, p = 4 \times 10^4, \theta = 0.3$; (e) $n = 400, p = 16 \times 10^4, \theta = 0.3$. Recovery error is measured as $\left|1 - \|\boldsymbol{A}\boldsymbol{D}_o\|_4^4 / n\right|$. All experiments are conducted on a 2.7 GHz Intel Core i5 processor (CPU of a 13-inch Mac Pro 2015).

MSP on the MNIST Dataset [LBB+98]

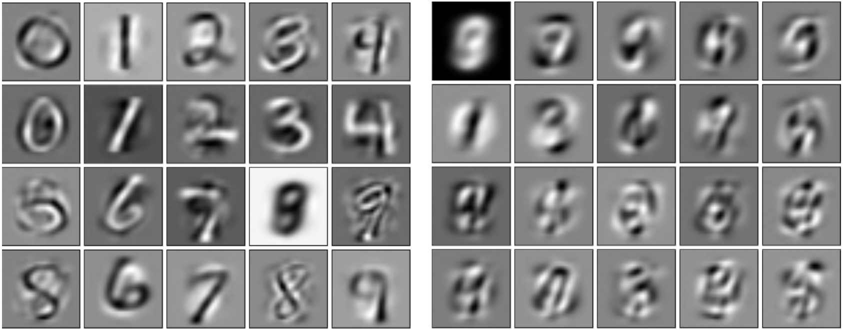


Figure: Bases learned from the MNIST dataset. Left: Some selected "meaningful" bases learned through MSP; Right: Top bases learned through PCA.

MSP on the MNIST Dataset [LBB$^+$98]

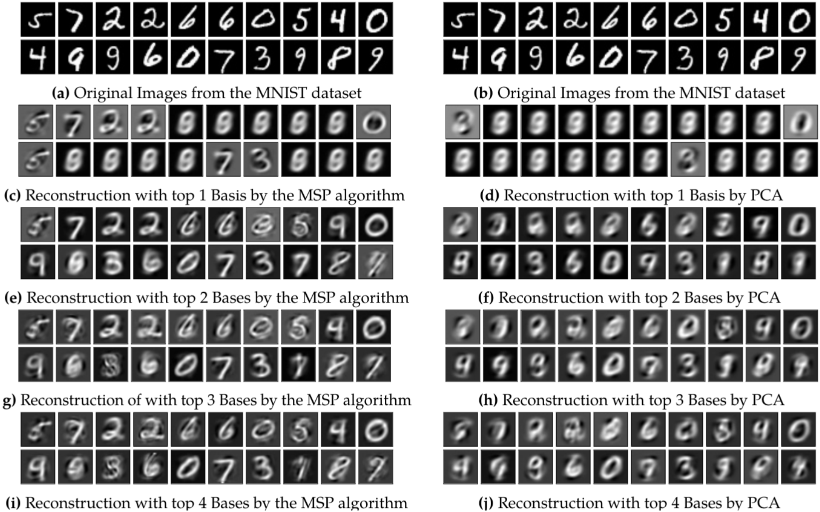
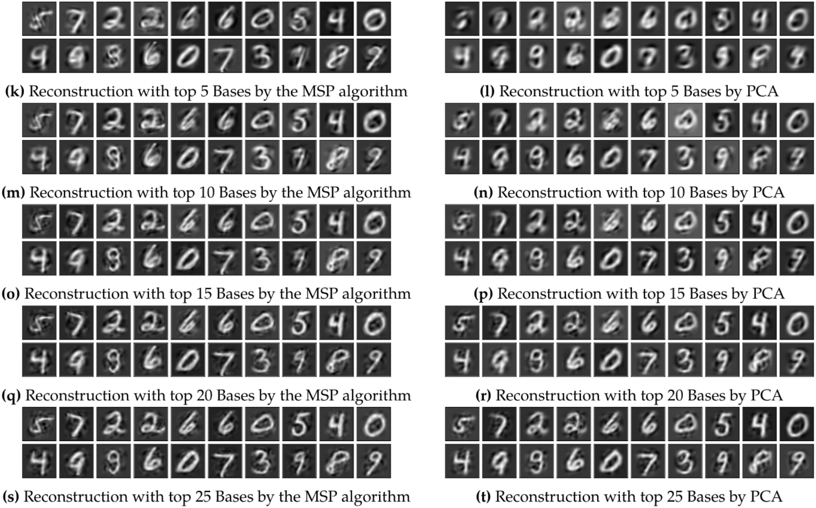


Figure: Reconstruction result comparison between MSP and PCA using different number of bases.

MSP on the MNIST Dataset [LBB+98]



(k) Reconstruction with top 5 Bases by the MSP algorithm

(l) Reconstruction with top 5 Bases by PCA

(m) Reconstruction with top 10 Bases by the MSP algorithm

(n) Reconstruction with top 10 Bases by PCA

(o) Reconstruction with top 15 Bases by the MSP algorithm

(p) Reconstruction with top 15 Bases by PCA

(q) Reconstruction with top 20 Bases by the MSP algorithm

(r) Reconstruction with top 20 Bases by PCA

(s) Reconstruction with top 25 Bases by the MSP algorithm

(t) Reconstruction with top 25 Bases by PCA

Figure: Reconstruction result comparison between MSP and PCA using different number of bases.

Generalization to Stiefel Manifold [ZMZM20]

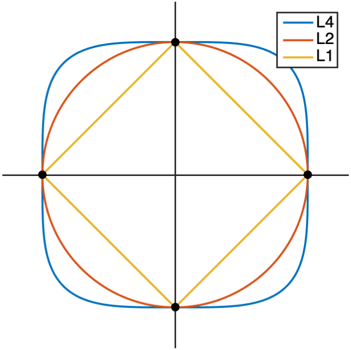


Figure: ℓ^1-,ℓ^2-, and ℓ^4-spheres in $\mathbb{R}^2$

Given data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$, recall the ℓ^4 dictionary learning

$$\max_{\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})} \frac{1}{4} \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \tag{6}$$

where the orthogonality constraint $\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$ can be viewed as *enforcing orthogonality constraint of* n *unit vectors*.

Generalization to Stiefel Manifold [ZMZM20]

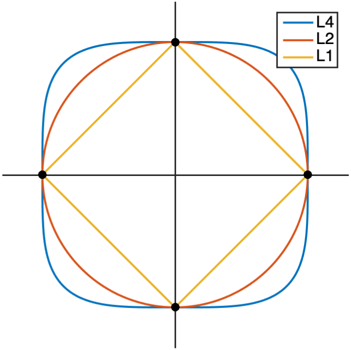


Figure: ℓ^1-,ℓ^2-, and ℓ^4-spheres in $\mathbb{R}^2$

Given data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$, recall the ℓ^4 dictionary learning

$$\max_{\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})} \frac{1}{4} \|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \qquad (6)$$

where the orthogonality constraint $\boldsymbol{A} \in \mathsf{O}(n;\mathbb{R})$ can be viewed as *enforcing orthogonality constraint of n unit vectors.*

Can we further reduce computation complexity if we are only interested in the top $k(1 \leq k \leq n)$ bases?

Generalization to Stiefel Manifold

Consider generalized Dictionary Learning from orthogonal group to Stiefel manifold $\mathsf{St}(k, n; \mathbb{R})$:[3]

$$\max_{\boldsymbol{W}} \frac{1}{4} \|\boldsymbol{W}^*\boldsymbol{Y}\|_4^4 \quad \text{s. t.}\, \boldsymbol{W} \in \mathsf{St}(k, n; \mathbb{R}) \subset \mathbb{R}^{n \times k}. \tag{7}$$

The MSP Algorithm can also be generalized to finding the top k bases:

$$\boldsymbol{W}_{t+1} = \mathcal{P}_{\mathsf{St}(k,n;\mathbb{R})} \left[\nabla_{\boldsymbol{W}} \phi(\boldsymbol{W}_t)\right] = \boldsymbol{U}_t \boldsymbol{V}_t^*, \tag{8}$$

where $\boldsymbol{U}_t \boldsymbol{\Sigma}_t \boldsymbol{V}_t^* = \mathsf{SVD}[\boldsymbol{Y}(\boldsymbol{Y}^*\boldsymbol{W}_t)^{\circ 3}]$.

[3] For any $1 \leq k \leq n$, $\mathsf{St}(k, n; \mathbb{R}) \doteq \{\boldsymbol{W} \in \mathbb{R}^{n \times k} : \boldsymbol{W}^*\boldsymbol{W} = \boldsymbol{I}_k\}$.

Relation with Geometric Interpretation of PCA

For data matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$:

- PCA aims at finding the top (k) left singular vector(s) of $\boldsymbol{Y}$:

$$\max_{\boldsymbol{W}} \frac{1}{2} \|\boldsymbol{W}^* \boldsymbol{Y}\|_F^2 \quad \mathrm{s.\,t.} \quad \boldsymbol{W} \in \mathsf{St}(k, n; \mathbb{R})$$

  can be considered as finding a direction (a k-dimensional subspace) in row$(\boldsymbol{Y})$ where $\boldsymbol{Y}$ has the largest ℓ^2 (Frobenius) norm.

- ℓ^4-Norm maximization

$$\max_{\boldsymbol{W}} \frac{1}{4} \|\boldsymbol{W}^* \boldsymbol{Y}\|_4^4 \quad \mathrm{s.\,t.} \quad \boldsymbol{W} \in \mathsf{St}(k, n; \mathbb{R})$$

  aims at finding a direction (a k-dimensional subspace) in row$(\boldsymbol{Y})$ where the projection of $\boldsymbol{Y}$ has the largest ℓ^4-norm.

Relation with Statistical Interpretation of PCA

View each column $\boldsymbol{y}_j, j \in [p]$ of data matrix $\boldsymbol{Y}$ as an n dimensional random vector that are i.i.d. drawn from a distribution of random variable $\boldsymbol{y}$. Let $\boldsymbol{Y}_c$ denote the centered $\boldsymbol{Y}$:$\boldsymbol{Y}_c \doteq \boldsymbol{Y}\left[\boldsymbol{I} - \frac{1}{p}\mathbf{1}\mathbf{1}^*\right]$. Then:

- $\max_{\boldsymbol{W} \in \mathsf{St}(k,n;\mathbb{R})} \frac{1}{2}\left\|\boldsymbol{W}^*\boldsymbol{Y}_c\right\|_F^2$ finds the top k uncorrelated projections of $\boldsymbol{y}$ with largest sample variance.
- $\max_{\boldsymbol{W} \in \mathsf{St}(k,n;\mathbb{R})} \frac{1}{4}\left\|\boldsymbol{W}^*\boldsymbol{Y}_c\right\|_4^4$ finds the top k uncorrelated projections of $\boldsymbol{y}$ with largest 4^{th} order moments.

Relation with ICA and 4^{th} Order Moment

In Independent Component Analysis (ICA) [HO97, HO00], finding maximizer or minimizer of *kurtosis*:

$$\operatorname{kurt}(\boldsymbol{w}^*\boldsymbol{y}) = \mathbb{E}[\boldsymbol{w}^*\boldsymbol{y}]^4 - 3\,\|\boldsymbol{w}\|_2^4 \tag{9}$$

can identify one independent component of $\boldsymbol{y}$.

Relation with ICA and 4^{th} Order Moment

In Independent Component Analysis (ICA) [HO97, HO00], finding maximizer or minimizer of *kurtosis*:

$$\operatorname{kurt}(\boldsymbol{w}^*\boldsymbol{y}) = \mathbb{E}[\boldsymbol{w}^*\boldsymbol{y}]^4 - 3\,\|\boldsymbol{w}\|_2^4 \tag{9}$$

can identify one independent component of $\boldsymbol{y}$.

Importance of 4^{th} Order Statistics

- The 4^{th} order statistics carries more "abnormal" information regarding nonnormality [Hub85, DeC97, CZY17]
- The distributions of real data (images) are usually not Gaussian [LPM03, HHH09].

Fixed-Point Style Algorithms

- **PCA**
  - Optimization: $$\max_{w \in \mathbb{S}^{n-1}} \varphi(w) \doteq \frac{1}{2} \|\boldsymbol{w}^* \boldsymbol{Y}\|_2^2$$
  - Algorithm: $$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}[\nabla_{\boldsymbol{w}} \varphi(\boldsymbol{w}_t)] = \frac{\boldsymbol{Y}\boldsymbol{Y}^* \boldsymbol{w}_t}{\|\boldsymbol{Y}\boldsymbol{Y}^* \boldsymbol{w}_t\|_2}$$
- **ICA**
  - Optimization: $$\max_{\boldsymbol{w} \in \mathbb{S}^{n-1}} \psi(\boldsymbol{w}) \doteq \frac{1}{4}\mathrm{kurt}[\boldsymbol{w}^* \boldsymbol{y}] = \frac{1}{4}\mathbb{E}\left[\boldsymbol{w}^* \boldsymbol{y}\right]^4 - \frac{3}{4}\|\boldsymbol{w}\|_2^4$$
  - Algorithm: $$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}\left[\nabla_{\boldsymbol{w}} \psi(\boldsymbol{w}_t)\right] = \frac{\mathbb{E}\left[\boldsymbol{y}\left(\boldsymbol{y}^* \boldsymbol{w}_t\right)^3\right] - 3\|\boldsymbol{w}_t\|_2^2 \boldsymbol{w}_t}{\left\|\mathbb{E}\left[\boldsymbol{y}\left(\boldsymbol{y}^* \boldsymbol{w}_t\right)^3\right] - 3\|\boldsymbol{w}_t\|_2^2 \boldsymbol{w}_t\right\|_2}$$
- **DL**
  - Optimization: $$\max_{\boldsymbol{W} \in \mathsf{St}(k,n;\mathbb{R})} \phi(\boldsymbol{W}) \doteq \frac{1}{4}\|\boldsymbol{W}^* \boldsymbol{Y}\|_4^4$$
  - Algorithm: $$\boldsymbol{W}_{t+1} = \mathcal{P}_{\mathsf{St}(k,n;\mathbb{R})}\left[\nabla_{\boldsymbol{W}} \phi(\boldsymbol{W}_t)\right] = \boldsymbol{U}_t \boldsymbol{V}_t^*,$$

  where $\boldsymbol{U}_t \boldsymbol{\Sigma}_t \boldsymbol{V}_t^* = \mathsf{SVD}[\boldsymbol{Y}(\boldsymbol{Y}^* \boldsymbol{W})^{\circ 3}]$.

Relations to PCA, ICA, and MSP

	Objectives	Constraint Sets	Algorithms
Power Iter.	$\varphi(\boldsymbol{w}) \doteq \frac{1}{2}\left\|\boldsymbol{w}^*\boldsymbol{Y}\right\|_2^2$	$\boldsymbol{w} \in \mathbb{S}^{n-1}$	$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}\left[\nabla_{\boldsymbol{w}}\varphi(\boldsymbol{w}_t)\right]$
FastICA	$\psi(\boldsymbol{w}) \doteq \frac{1}{4}\mathrm{kurt}[\boldsymbol{w}^*\boldsymbol{y}]$	$\boldsymbol{w} \in \mathbb{S}^{n-1}$	$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}\left[\nabla_{\boldsymbol{w}}\psi(\boldsymbol{w}_t)\right]$
MSP	$\phi(\boldsymbol{W}) \doteq \frac{1}{4}\left\|\boldsymbol{W}^*\boldsymbol{Y}\right\|_4^4$	$\boldsymbol{W} \in \mathsf{St}(k, n; \mathbb{R})$	$\boldsymbol{W}_{t+1} = \mathcal{P}_{\mathsf{St}(k,n;\mathbb{R})}\left[\nabla_{\boldsymbol{W}}\phi(\boldsymbol{W}_t)\right]$

Table: Similarities among fixed-point algorithms for: PCA (Power iteration), ICA (FastICA), and DL (MSP).

Different Type of Imperfect Measurements I

Noisy Measurements: $\boldsymbol{Y}_N := \boldsymbol{Y} + \boldsymbol{G}$, $\boldsymbol{G} \in \mathbb{R}^{n \times p}$ is matrix with $g_{i,j} \sim_{iid} \mathcal{N}(0, \eta^2)$ and $\eta > 0$ the variance of the noise.

Different Type of Imperfect Measurements I

Noisy Measurements: $\boldsymbol{Y}_N := \boldsymbol{Y} + \boldsymbol{G}$, $\boldsymbol{G} \in \mathbb{R}^{n \times p}$ is matrix with $g_{i,j} \sim_{iid} \mathcal{N}(0, \eta^2)$ and $\eta > 0$ the variance of the noise.

Proposition (Objective with Small Noise)

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid}$ BG(θ), $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix, and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. For any orthogonal matrix $\boldsymbol{W} \in \mathsf{O}(n;\mathbb{R})$ and any random Gaussian matrix $\boldsymbol{G} \in \mathbb{R}^{n \times p}, g_{i,j} \sim_{iid} \mathcal{N}(0, \eta^2)$ independent of $\boldsymbol{X}_o$, let $\boldsymbol{Y}_N = \boldsymbol{Y} + \boldsymbol{G}$ denote the data with noise. Then the expectation of $\|\boldsymbol{W}^* \boldsymbol{Y}_N\|_4^4$ is:

$$\frac{1}{np} \mathbb{E}_{\boldsymbol{X}_o, \boldsymbol{G}} \|\boldsymbol{W}^* \boldsymbol{Y}_N\|_4^4 = 3\theta(1-\theta) \frac{\|\boldsymbol{W}^* \boldsymbol{D}_o\|_4^4}{n} + C_{\theta,\eta},$$

where $C_{\theta,\eta}$ is a constant depending on θ and η.

Different Type of Imperfect Measurements II

Measurements with Outliers: $\boldsymbol{Y}_O := [\boldsymbol{Y}, \boldsymbol{G}']$, where $\boldsymbol{Y}_O$ contains extra columns ($\boldsymbol{G}' \in \mathbb{R}^{n \times \tau p}$)[4] that is generated from an independent Gaussian process $g'_{i,j} \sim_{iid} \mathcal{N}(0, 1)$, and τ controls the portion of the outliers, w.r.t. the clean data size p.

[4] When τp is not an integer, τp is rounded to the closest integer.

Different Type of Imperfect Measurements II

Measurements with Outliers: $\boldsymbol{Y}_O := [\boldsymbol{Y}, \boldsymbol{G}']$, where $\boldsymbol{Y}_O$ contains extra columns ($\boldsymbol{G}' \in \mathbb{R}^{n \times \tau p}$)[4] that is generated from an independent Gaussian process $g'_{i,j} \sim_{iid} \mathcal{N}(0,1)$, and τ controls the portion of the outliers, w.r.t. the clean data size p.

Proposition (Objective with Outliers)

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} BG(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. For any orthogonal matrix $\boldsymbol{W} \in \mathsf{O}(n;\mathbb{R})$ and any random Gaussian matrix $\boldsymbol{G}' \in \mathbb{R}^{n \times \tau p}, g'_{i,j} \sim_{iid} \mathcal{N}(0,1)$ independent of $\boldsymbol{X}_o$, let $\boldsymbol{Y}_O = [\boldsymbol{Y}, \boldsymbol{G}']$ denote the data with outliers $\boldsymbol{G}'$. Then the expectation of $\|\boldsymbol{W}^* \boldsymbol{Y}_O\|_4^4$ is:

$$\frac{1}{np} \mathbb{E}_{\boldsymbol{X}_o, \boldsymbol{G}'} \|\boldsymbol{W}^* \boldsymbol{Y}_O\|_4^4 = 3\theta(1-\theta) \frac{\|\boldsymbol{W}^* \boldsymbol{D}_o\|_4^4}{n} + C_\theta,$$

where C_θ is a constant depending on θ.

[4] When τp is not an integer, τp is rounded to the closest integer.

Different Type of Imperfect Measurements III

Measurements with Sparse Corruptions: $\boldsymbol{Y}_C := \boldsymbol{Y} + \sigma \boldsymbol{B} \circ \boldsymbol{S}$, where $\sigma > 0$ controls the scale of corrupting entries, $\boldsymbol{B} \in \mathbb{R}^{n \times p}$ is a Bernoulli matrix with $b_{i,j} \sim_{iid} \mathrm{Ber}(\beta)$, where $\beta \in (0,1)$ controls the ratio of the sparse corruptions, and entries $s_{i,j}$ of $\boldsymbol{S} \in \mathbb{R}^{n \times p}$ are i.i.d. drawn from a *Rademacher* distribution:

$$s_{i,j} = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}.$$

Different Type of Imperfect Measurements III

Measurements with Sparse Corruptions: $\boldsymbol{Y}_C := \boldsymbol{Y} + \sigma \boldsymbol{B} \circ \boldsymbol{S}$, where $\sigma > 0$ controls the scale of corrupting entries, $\boldsymbol{B} \in \mathbb{R}^{n \times p}$ is a Bernoulli matrix with $b_{i,j} \sim_{iid} \text{Ber}(\beta)$, where $\beta \in (0,1)$ controls the ratio of the sparse corruptions, and entries $s_{i,j}$ of $\boldsymbol{S} \in \mathbb{R}^{n \times p}$ are i.i.d. drawn from a *Rademacher* distribution:

$$s_{i,j} = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}.$$

Proposition (Objective with Sparse Corruptions)

$\forall \theta \in (0,1)$, let $\boldsymbol{X}_o \in \mathbb{R}^{n \times p}$, $x_{i,j} \sim_{iid} BG(\theta)$, $\boldsymbol{D}_o \in \mathsf{O}(n;\mathbb{R})$ is any orthogonal matrix and $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$. For any orthogonal matrix $\boldsymbol{W} \in \mathsf{O}(n;\mathbb{R})$ and any random Bernoulli matrix $\boldsymbol{B} \in \mathbb{R}^{n \times p}, b_{i,j} \sim_{iid} Ber(\beta)$ independent of $\boldsymbol{X}_o$, let $\boldsymbol{Y}_C = \boldsymbol{Y} + \sigma \boldsymbol{B} \circ \boldsymbol{S}$ denote the data with sparse corruptions, and $\boldsymbol{S} \in \mathbb{R}^{n \times p}$ is defined as (35). Then the expectation of $\|\boldsymbol{W}^* \boldsymbol{Y}_C\|_4^4$ is:

$$\frac{1}{np} \mathbb{E}_{\boldsymbol{X}_o, \boldsymbol{B}, \boldsymbol{S}} \|\boldsymbol{W}^* \boldsymbol{Y}_C\|_4^4 = 3\theta(1-\theta) \frac{\|\boldsymbol{W}^* \boldsymbol{D}_o\|_4^4}{n} + \sigma^4 \beta (1 - 3\beta) \frac{\|\boldsymbol{W}\|_4^4}{n} + C_{\theta,\sigma,\beta},$$

where $C_{\theta,\sigma,\beta}$ is a constant depending on θ, σ and β.

Numerical Experiments I

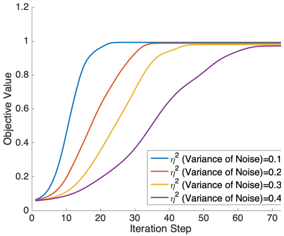


Figure: $n = 50, p = 20,000, \theta = 0.3$, varying η^2 from 0.1 to 0.4

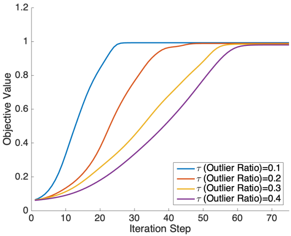


Figure: $n = 50, p = 20,000, \theta = 0.3$, varying τ from 0.1 to 0.4

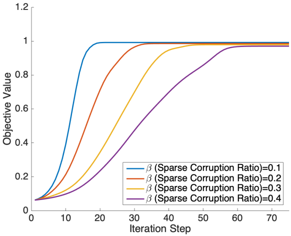


Figure: $n = 50, p = 20,000, \theta = 0.3$, $\sigma = 1$, varying β from 0.1 to 0.4

Figure: Normalized $\|\boldsymbol{W}^*\boldsymbol{D}_o\|_4^4 / n$ of the MSP algorithm for dictionary learning, using imperfect measurements $\boldsymbol{Y}_N, \boldsymbol{Y}_O, \boldsymbol{Y}_C$, respectively.

Numerical Experiments II

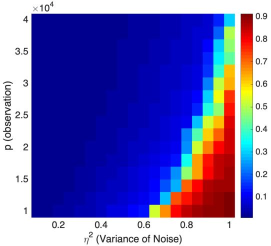


Figure: Noise: $n = 50, \theta = 0.3$

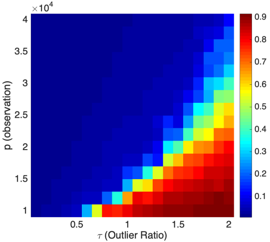


Figure: Outliers: $n = 50, \theta = 0.3$

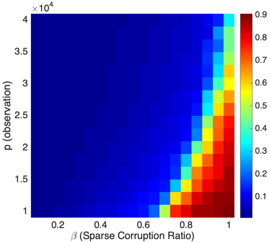


Figure: Corruptions: $n = 50, \theta = 0.3$

Figure: Average normalized error $\left|1 - \|\boldsymbol{W}^* \boldsymbol{D}_o\|_4^4 / n\right|$ of 10 random trials for the MSP Algorithm: **(a)** Varying sample size p and variance of noise η^2; **(b)** Varying sample size p and Gaussian Outlier ratio τ; **(c)** Varying sample size p and sparse corruption ratio β, with fixed $\sigma = 1$.

Real Image Data: MNIST

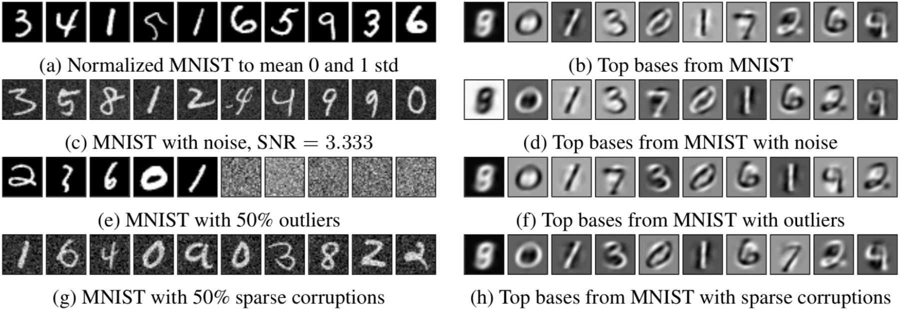


Figure: Top Bases learned from imperfect measurements of MNIST.

Real Image Data: Single Image

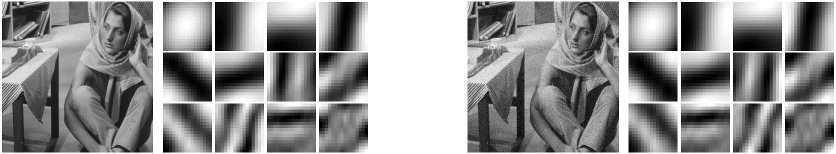


Figure: The top 12 bases learned from all 16×16 patches of Barbara, both with (right) and without (left) Gaussian noise. The noisy image is produced by adding Gaussian noise to the clean image, resulting in SNR of 5.87.

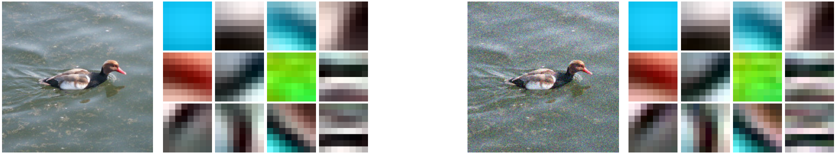


Figure: The top 12 bases learned from all $8 \times 8 \times 3$ color patches of the clean and noisy image, respectively. Here, the SNR of the noisy image is 6.56.

Real Image Data: Single Image

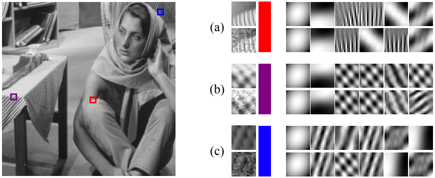


Figure: Representations of three 16×16 patches from Barbara w/ and w/o noise. Each selected patch is visualized, both w/ and w/o noise, and the top 6 corresponding bases are shown.

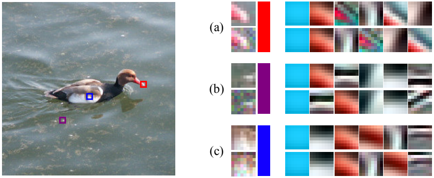
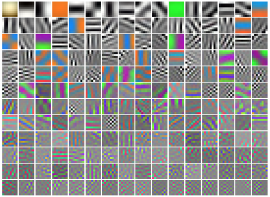
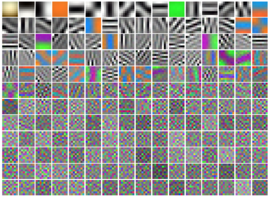


Figure: Representations of three $8 \times 8 \times 3$ patches from duck w/ and w/o noise. Each selected patch is visualized, both w/ and w/o noise, and the top 6 corresponding bases are shown.

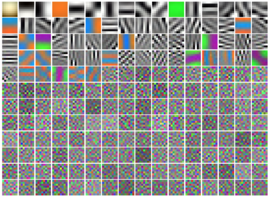
Real Image Data: CIFAR-10



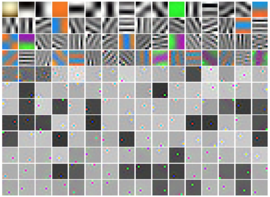
(a) Clean Patches



(b) Patches with Noise (SNR $= 6.23$)



(c) Patches with Outliers



(d) Patches with Sparse Corruptions

Figure: All $8 \times 8 \times 3 = 192$ bases learned from $100,000$ random 8×8 colored patches sampled from the CIFAR-10 data-set. (a) Learned Bases from clean CIFAR-10; (b) Learned Bases from CIFAR-10 with Gaussian noise, SNR $= 6.23$; (c) Learned Bases from CIFAR-10 with 20% of Gaussian outliers; (d) Learned Bases from CIFAR-10 with 50% of sparse corruptions.

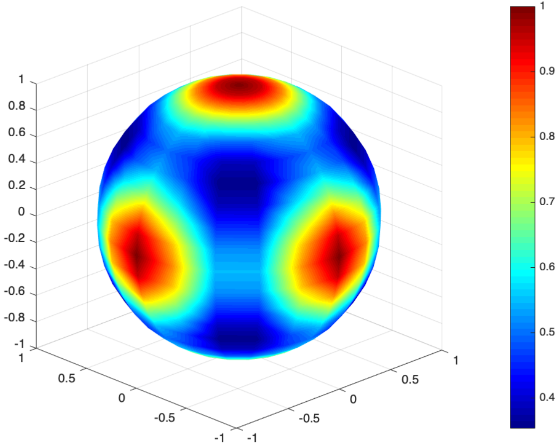
Summary

[ZYL$^+$19]:

- The MSP algorithm solves complete dictionary learning **holistically**.
- The sample complexity $\Omega(n^2 \ln n)$ corroborates with experiments.
- Special **symmetries** help nonconvex optimization.

[ZMZM20]:

- The MSP algorithm is a **fixed-point** type algorithm just like Power-iteration [Jol11] and FastICA [HO97].
- The MSP algorithm is robust to stable to noise, robust to outliers and resilient to sparse corruptions.



Thanks! & Questions?

References I

Michal Aharon, Michael Elad, Alfred Bruckstein, et al.
K-svd: An algorithm for designing overcomplete dictionaries for sparse representation.
IEEE Transactions on signal processing, 54(11):4311, 2006.

Yu Bai, Qijia Jiang, and Ju Sun.
Subgradient descent learns orthogonal dictionaries.
arXiv preprint arXiv:1810.10702, 2018.

Boaz Barak, Jonathan Kelner, and David Steurer.
Dictionary learning and tensor decomposition via the sum-of-squares method.
In *STOC*, 2015.

Meghan K Cain, Zhiyong Zhang, and Ke-Hai Yuan.
Univariate and multivariate skewness and kurtosis for measuring nonnormality: Prevalence, influence and estimation.
Behavior Research Methods, 49(5):1716–1735, 2017.

Lawrence T DeCarlo.
On the meaning and use of kurtosis.
Psychological methods, 2(3):292, 1997.

Aapo Hyvärinen, Jarmo Hurri, and Patrick O Hoyer.
Natural image statistics: A probabilistic approach to early computational vision., volume 39.
Springer Science & Business Media, 2009.

Aapo Hyvärinen and Erkki Oja.
A fast fixed-point algorithm for independent component analysis.
Neural Computation, 9:1483–1492, 1997.

References II

Aapo Hyvärinen and Erkki Oja.
Independent component analysis: algorithms and applications.
Neural networks, 13(4-5):411–430, 2000.

Peter J Huber.
Projection pursuit.
The annals of Statistics, pages 435–475, 1985.

Ian Jolliffe.
Principal component analysis.
Springer, 2011.

Yanjun Li and Yoram Bresler.
Global geometry of multichannel sparse blind deconvolution on the sphere.
In *Advances in Neural Information Processing Systems*, pages 1132–1143, 2018.

Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al.
Gradient-based learning applied to document recognition.
Proceedings of the IEEE, 86(11):2278–2324, 1998.

Ann B Lee, Kim S Pedersen, and David Mumford.
The nonlinear statistics of high-contrast patches in natural images.
International Journal of Computer Vision, 54(1-3):83–103, 2003.

Jiang-Hua Lu.
A note on a theorem of Stanton-Weinstein on the L4-norm of spherical harmonics.
In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 102, page 561, 1987.

References III

Tengyu Ma, Jonathan Shi, and David Steurer.
Polynomial-time tensor decompositions with sum-of-squares.
In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 438–446. IEEE, 2016.

Ju Sun, Qing Qu, and John Wright.
Complete dictionary recovery over the sphere i: Overview and the geometric picture.
IEEE Transactions on Information Theory, 63(2):853–884, 2017.

Tselil Schramm and David Steurer.
Fast and robust tensor decomposition with applications to dictionary learning.
arXiv preprint arXiv:1706.08672, 2017.

Robert J Stanton and Alan Weinstein.
On the L4 norm of spherical harmonics.
In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 89, page 343, 1981.

Yuqian Zhang, Han-Wen Kuo, and John Wright.
Structured local optima in sparse blind deconvolution.
arXiv preprint arXiv:1806.00338, 2018.

Yuexiang Zhai, Hermish Mehta, Zhengyuan Zhou, and Yi Ma.
Understanding l4-based dictionary learning: Interpretation, stability, and robustness.
In *International Conference on Learning Representations (ICLR)*, 2020.

Yuexiang Zhai, Zitong Yang, Zhenyu Liao, John Wright, and Yi Ma.
Complete dictionary learning via ℓ^4-norm maximization over the orthogonal group.
arXiv preprint arXiv:1906.02435, 2019.