Taming nonconvexity: from smooth to nonsmooth problems

Ju Sun
Department of Mathematics
Stanford University

SINE Seminar at Coordiated Science Laboratory, UIUC

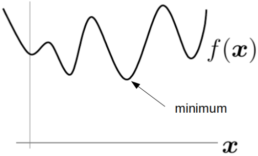
November 5, 2018

Optimization is everywhere

"Nothing takes place in the world whose meaning is not that of some **maximum or minimum**.*"*

Leonhard Euler

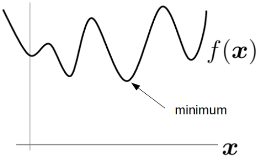
$$\begin{aligned} &\min f(\boldsymbol{x}) \\ &\text{s. t. } \boldsymbol{x} \in \mathcal{S}. \end{aligned}$$



Optimization is everywhere

"Nothing takes place in the world whose meaning is not that of some **maximum or minimum**.*"*

$$\min f(\boldsymbol{x})$$
$$\text{s.t. } \boldsymbol{x} \in \mathcal{S}.$$



Leonhard Euler

HISTORICAL PERSPECTIVES

300 Years of Optimal Control: From The Brachystochrone to the Maximum Principle

Hector J. Sussmann and Jan C. Willems

Optimal control was born in 1697—300 years ago—in Groningen, a university town in the north of The Netherlands, when Johann Bernoulli, professor of mathematics at the local university from 1695 to 1705, published his solution of the *brachystochrone problem*. The year before he had challenged his contemporaries to solve this problem. We will tell the story of the authors. We gladly plead guilty to most of this charge—and state for the record that *we are both control theorists, and one of us is a professor at Groningen*—asking only that the word "merely" be stricken out. Our biases may of course explain how

Historic heroes

Euclid	de Fermat
Newton	Leibniz
Bernoulli's	Euler
Lagrange	Legendre
Gauss	Fourier
Cauchy	Hadamard

...

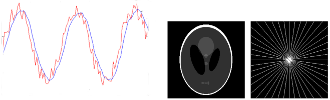
Optimization is everywhere

computer vision

Optimization is everywhere

computer vision

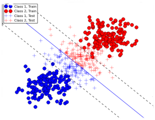
signal processing



Optimization is everywhere

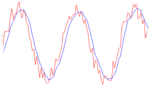
computer vision

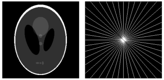
machine learning



$P(A|B) \propto P(B|A)P(A)$

signal processing

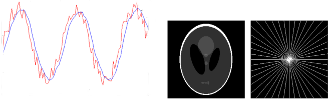




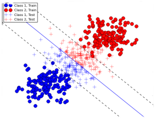
Optimization is everywhere

computer vision

signal processing

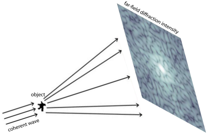


machine learning



$P(A|B) \propto P(B|A)P(A)$

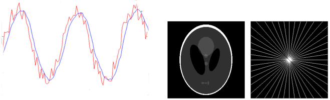
scientific imaging



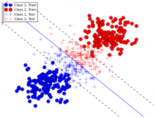
Optimization is everywhere

computer vision

signal processing

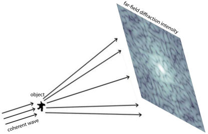


machine learning

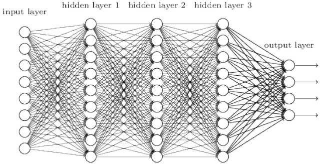


$P(A|B) \propto P(B|A)P(A)$

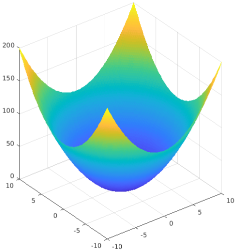
scientific imaging



neural networks



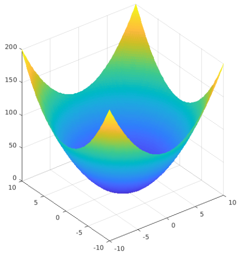
Convex analysis and optimization



All local minimizers are **global**!

(All critical points are **global**!)

Convex analysis and optimization



All local minimizers are **global**!

(All critical points are **global**!)

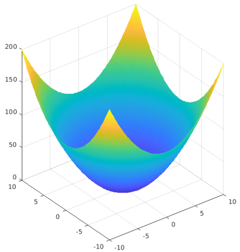
Interior-point method (80's–00's)
–"Most" convex problems can be solved **efficiently**!

Modeling languages (00's–10's)

$$\begin{array}{ll} \text{minimize} & \|Ax-b\|_2 \\ \text{subject to} & Cx = d \\ & \|x\|_\infty \le e \end{array}$$

```
cvx_begin
    variable x(n)
    minimize( norm( A * x - b, 2 ) )
    subject to
        C * x == d
        norm( x, Inf ) <= e
cvx_end
```

Convex analysis and optimization



All local minimizers are **global**!

(All critical points are **global**!)

Interior-point method (80's–00's)

–"Most" convex problems can be solved **efficiently**!

Modeling languages (00's–10's)

$$\begin{array}{ll} \text{minimize} & \|Ax-b\|_2 \\ \text{subject to} & Cx = d \\ & \|x\|_\infty \le e \end{array}$$

```
cvx_begin
    variable x(n)
    minimize( norm( A * x - b, 2 ) )
    subject to
        C * x == d
        norm( x, Inf ) <= e
cvx_end
```

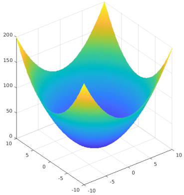
...in fact, the great watershed in optimization isn't between linearity and nonlinearity, but **convexity and nonconvexity**.

— R. Tyrrell Rockafellar [Rockafellar, 1993]

Nonconvex optimization?

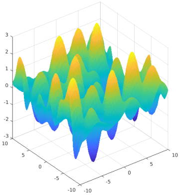
$$
\begin{aligned}
&\min\ f(\boldsymbol{x}) \\
&\mathrm{s.\,t.}\ \boldsymbol{x} \in \mathcal{S}.
\end{aligned}
$$

“easy”



All local mins are global!

“hard”

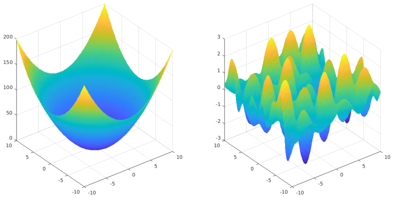


Spurious local mins!

Nonconvex optimization?

$$\min \ f(\boldsymbol{x})$$
$$\text{s. t. } \boldsymbol{x} \in \mathcal{S}.$$

"easy" **"hard"**



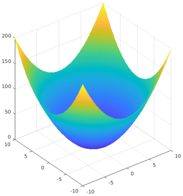
All local mins are global! **Spurious local mins!**

Nonconvex: Even computing a local minimizer is NP-hard! (see, e.g., [Murty and Kabadi, 1987])

Nonconvex optimization?

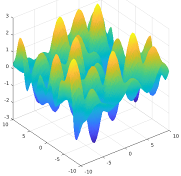
$$\min f(\boldsymbol{x})$$
$$\text{s.t. } \boldsymbol{x} \in \mathcal{S}.$$

"easy"



All local mins are global!

"hard"

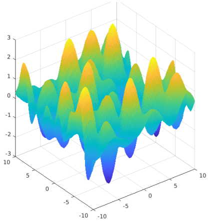


Spurious local mins!

Nonconvex: Even computing a local minimizer is NP-hard! (see, e.g., [Murty and Kabadi, 1987])

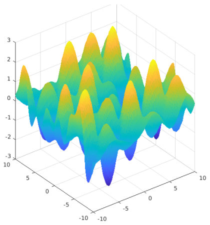
Many problems in modern **signal processing**, **machine learning**, **statistics**, **imaging**, ..., are most naturally formulated as **nonconvex** optimization problems.

Nonscary nonconvex optimization



In theory: Even computing a local minimizer is NP-hard!

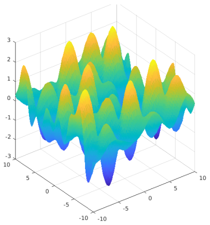
Nonscary nonconvex optimization



In theory: Even computing a local minimizer is NP-hard!

In practice: Heuristic algorithms are often surprisingly successful.

Nonscary nonconvex optimization

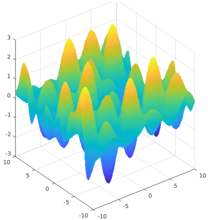


In theory: Even computing a local minimizer is NP-hard!

In practice: Heuristic algorithms are often surprisingly successful.

e.g., training deep neural networks—SGD and variants, plus a few tricks

Nonscary nonconvex optimization



In theory: Even computing a local minimizer is NP-hard!

In practice: Heuristic algorithms are often surprisingly successful.

e.g., training deep neural networks—SGD and variants, plus a few tricks

Which nonconvex optimization problems are easy?

Running app: sparsifying transform learning

Given data $\boldsymbol{Y}$, learn $\boldsymbol{Q}$, st $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small

Running app: sparsifying transform learning

Given data $\boldsymbol{Y}$, learn $\boldsymbol{Q}$, st $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small

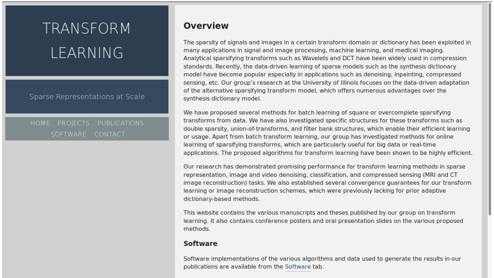


image credit: Professor Yoram Bresler's research website

Running app: sparsifying transform learning

Given data $\boldsymbol{Y}$, learn $\boldsymbol{Q}$, st $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small

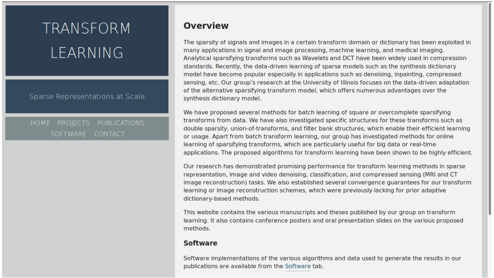


image credit: Professor Yoram Bresler's research website

– **Dictionary learning**: factor $\boldsymbol{Y}$ as $\boldsymbol{Y} \approx \boldsymbol{A}\boldsymbol{X}$ st $\boldsymbol{X}$ is sparse

Running app: sparsifying transform learning

Given data $\boldsymbol{Y}$, learn $\boldsymbol{Q}$, st $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small

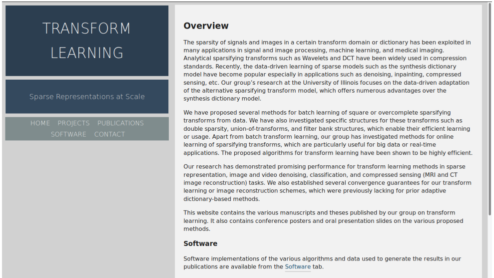


image credit: Professor Yoram Bresler's research website

- **Dictionary learning**: factor $\boldsymbol{Y}$ as $\boldsymbol{Y} \approx \boldsymbol{A}\boldsymbol{X}$ st $\boldsymbol{X}$ is sparse
- Apps: image processing, computer vision, computational imaging [Mairal et al., 2014]

Running app: sparsifying transform learning

Given data $\boldsymbol{Y}$, learn $\boldsymbol{Q}$, st $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small

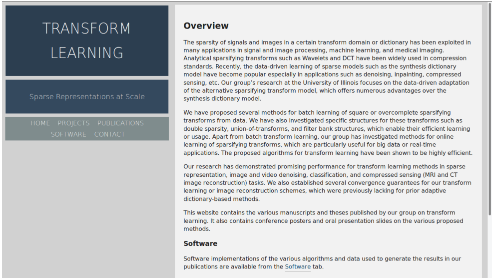


image credit: Professor Yoram Bresler's research website

- **Dictionary learning**: factor $\boldsymbol{Y}$ as $\boldsymbol{Y} \approx \boldsymbol{A}\boldsymbol{X}$ st $\boldsymbol{X}$ is sparse
- Apps: image processing, computer vision, computational imaging [Mairal et al., 2014]
- Cascaded with nonlinearity [Ravishankar and Wohlberg, 2018]

One element each time

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

One element each time

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

A naive formulation:

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

One element each time

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

A naive formulation:

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

Nonconvex "relaxation":

$$\boxed{\min_{\boldsymbol{q}} \quad f(\boldsymbol{q}) \doteq \frac{1}{m}\|\boldsymbol{q}^*\boldsymbol{Y}\|_1 \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

Many precedents, e.g., [Zibulevsky and Pearlmutter, 2001] in blind source separation.

Here, inspired by [Spielman et al., 2012, Sun et al., 2015]

Recovery setting for analysis

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

Recovery setting for analysis

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^* \boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^* \boldsymbol{Y}\|_0$ is small.

To study possibility of **recovery**, given $\boldsymbol{Q}_0$ ortho and $\boldsymbol{X}_0$ sparse,

$$\boldsymbol{Q}_0^* \times \boldsymbol{Y} \quad = \quad \boldsymbol{X}_0,$$

recover $\boldsymbol{Q}_0$ and $\boldsymbol{X}_0$ $\Big($**up to signed permutation**$\Big)$.

Nonconvex "relaxation":

$$\boxed{\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m} \|\boldsymbol{q}^* \boldsymbol{Y}\|_1 \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

Toward geometric intuition

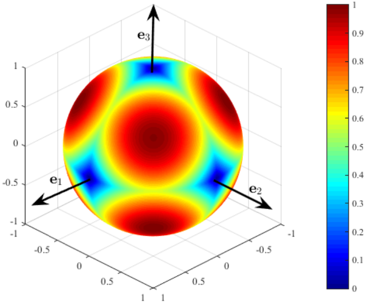
$$\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m} \|\boldsymbol{q}^* \boldsymbol{Y}\|_1 \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

A low-dimensional example ($n = 3$) of the landscape when the target transformation $\boldsymbol{Q}_0 = \boldsymbol{I}$ and $m \to \infty$

Toward geometric intuition

$$\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m}\|\boldsymbol{q}^*\boldsymbol{Y}\|_1 \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

A low-dimensional example ($n = 3$) of the landscape when the target transformation $\boldsymbol{Q}_0 = \boldsymbol{I}$ and $m \to \infty$



- global mins
- saddles

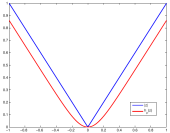
Toward geometric analysis

Smoothed model problem

$$\boxed{\min\ f_1(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{j=1}^{m} |\boldsymbol{q}^*\boldsymbol{y}_j| \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

$\downarrow\downarrow$ smoothing $\downarrow\downarrow$

$$\boxed{\min\ f(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{i=1}^{m} h_\mu\left(\boldsymbol{q}^*\boldsymbol{y}_i\right) \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

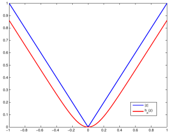


$$h_\mu\left(z\right) = \mu \log\cosh \frac{z}{\mu}$$

Toward geometric analysis

Smoothed model problem

$$\boxed{\min\ f_1(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{j=1}^{m} |\boldsymbol{q}^*\boldsymbol{y}_j| \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$



$$h_\mu(z) = \mu \log\cosh \frac{z}{\mu}$$

$\downarrow\downarrow$ smoothing $\downarrow\downarrow$

$$\boxed{\min\ f(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{i=1}^{m} h_\mu(\boldsymbol{q}^*\boldsymbol{y}_i) \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

For analysis: Bernoulli-Gaussian model $\boldsymbol{X}_0 = \boldsymbol{\Omega}_0 \circ \boldsymbol{V}_0$, $\boldsymbol{\Omega}_0 \sim_{\text{iid}} \text{Ber}(\theta)$, $\boldsymbol{V}_0 \sim_{\text{iid}} \mathcal{N}(0,1)$.

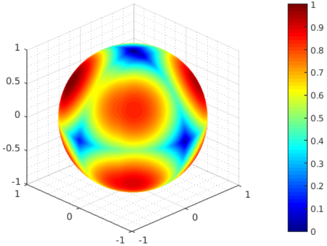
Sparsity parameter θ; average number of nonzeros per column is θn.

The geometric result

$$\boxed{\min \ f(\boldsymbol{q}) \doteq \frac{1}{m} \sum_{i=1}^{m} h_\mu \left(\boldsymbol{q}^* \boldsymbol{y}_i \right) \ \text{s.t.} \ \|\boldsymbol{q}\|_2^2 = 1.}$$

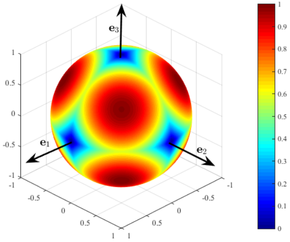
The geometric result

$$\min\ f(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{i=1}^{m} h_\mu\left(\boldsymbol{q}^*\boldsymbol{y}_i\right) \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$



$m = 100$

$\Longrightarrow$



$m \to \infty$

Theorem (Informal, [Sun et al., 2015])

When p is reasonably large, and $\theta \leq 1/3$, with high probability,

All local minimizers are "global".

Comparison with prior results

Efficient algorithms with performance guarantees

[Spielman et al., 2012]	$\boldsymbol{Q} \in \mathbb{R}^{n \times n}$, $\theta = \tilde{O}\left(1/\sqrt{n}\right)$
[Agarwal et al., 2013b]	$\boldsymbol{Q} \in \mathbb{R}^{m \times n}$ $(m \le n)$, $\theta = \tilde{O}\left(1/\sqrt{n}\right)$
[Arora et al., 2013]	$\boldsymbol{Q} \in \mathbb{R}^{m \times n}$ $(m \le n)$, $\theta = \tilde{O}\left(1/\sqrt{n}\right)$
[Arora et al., 2015]	$\boldsymbol{Q} \in \mathbb{R}^{m \times n}$ $(m \le n)$, $\theta = \tilde{O}\left(1/\sqrt{n}\right)$

Quasipolynomial algorithms with better guarantees

[Arora et al., 2014]	different model, $\theta = O\left(1/\mathrm{polylog}\left(n\right)\right)$
[Barak et al., 2014]	sum-of-squares, $\theta = \tilde{O}\left(1\right)$
	polytime for $\theta = O(n^{-\varepsilon})$.

Other theoretical work on **local geometry**:

[Gribonval and Schnass, 2010], [Geng and Wright, 2011], [Schnass, 2014], etc

> This work: the **first** polynomial-time algorithm for learning complete $\boldsymbol{Q}$ with $\theta = \Omega(1)$.

See also recent refined SOS analysis [Ma et al., 2016a] with similar guarantees.

Back to the question

$$\boxed{\textbf{Which nonconvex optimization problems are easy?}}$$

... two types of partial answers

A1: Problems with nice global landscapes

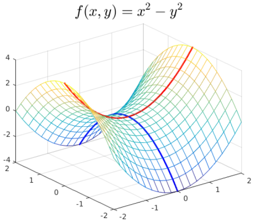
(P-1) All local minimizers are **global**

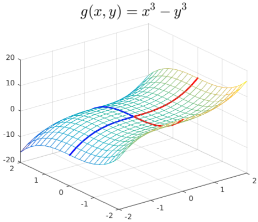
(P-2) All saddle points (and local maximizers) have a **directional negative curvature**, i.e., $\lambda_{\min}(\mathrm{Hess}) < 0$

A1: Problems with nice global landscapes

(P-1) All local minimizers are **global**

(P-2) All saddle points (and local maximizers) have a **directional negative curvature**, i.e., $\lambda_{\min}(\text{Hess}) < 0$





$\nabla^2 f = \text{diag}(2, -2)$

Ridable/strict saddle (also [Ge et al., 2015])

$\nabla^2 f = \text{diag}(6x, -6y)$

locally shaped by high-order derivatives at $\mathbf{0}$

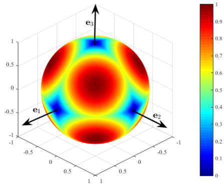
A1: Problems with nice global landscapes

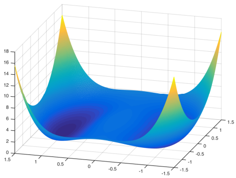
All local mins are global, all saddles are strict

A1: Problems with nice global landscapes

All local mins are global, all saddles are strict

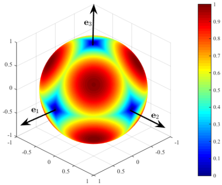
Eigenvalue problems (folklore!)
Sparsifying dictionary learning [Sun et al., 2015]
Generalized phase retrieval [Sun et al., 2016]
Orthogonal tensor decomposition [Ge et al., 2015]
Low-rank matrix recovery and completion
[Ge et al., 2016, Bhojanapalli et al., 2016]
Phase synchronization [Boumal, 2016]
Community detection [Bandeira et al., 2016]
Deep/shallow networks [Kawaguchi, 2016,
Lu and Kawaguchi, 2017, Soltanolkotabi et al., 2017]
Sparse blind deconvolution [Zhang et al., 2017]





A1: Problems with nice global landscapes

All local mins are global, all saddles are strict



Eigenvalue problems (folklore!)

Sparsifying dictionary learning [Sun et al., 2015]

Generalized phase retrieval [Sun et al., 2016]

Orthogonal tensor decomposition [Ge et al., 2015]

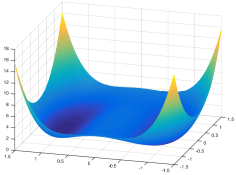
Low-rank matrix recovery and completion [Ge et al., 2016, Bhojanapalli et al., 2016]

Phase synchronization [Boumal, 2016]

Community detection [Bandeira et al., 2016]

Deep/shallow networks [Kawaguchi, 2016, Lu and Kawaguchi, 2017, Soltanolkotabi et al., 2017]

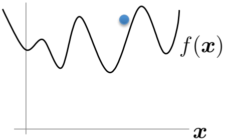
Sparse blind deconvolution [Zhang et al., 2017]



Algorithms: virtually everything reasonable works!

[Conn et al., 2000, Nesterov and Polyak, 2006, Goldfarb, 1980, Jin et al., 2017]

A2: Problems with nice local landscapes



Use problem structure to find a **clever (sometimes random) initial guess.**

Analyze local search algorithms **in the vicinity of the optimum.**

- **Matrix completion/recovery**: [Keshavan et al., 2010], [Jain et al., 2013], [Hardt, 2014], [Hardt and Wootters, 2014], [Netrapalli et al., 2014], [Jain and Netrapalli, 2014], [Sun and Luo, 2014], [Zheng and Lafferty, 2015], [Tu et al., 2015], [Chen and Wainwright, 2015], [Sa et al., 2015], [Wei et al., 2015]. Also [Jain et al., 2010]
- **Dictionary learning**: [Agarwal et al., 2013a], [Arora et al., 2013], [Agarwal et al., 2013b], [Arora et al., 2015], [Chatterji and Bartlett, 2017], [Gilboa et al., 2018]
- **Tensor recovery**: [Jain and Oh, 2014], [Anandkumar et al., 2014b], [Anandkumar et al., 2014a], [Anandkumar et al., 2015]
- **Phase retrieval**: [Netrapalli et al., 2013], [Candès et al., 2015], [Chen and Candès, 2015], [White et al., 2015], [Wang et al., 2016], [Chen et al., 2018]

Nonscary nonconvex problems

Problems with nice global/local landscapes

- My webpage: `http://sunju.org/research/nonconvex/`
- Sun, Ju and Qu, Qing and Wright, John. **When are nonconvex problems not scary?**. arXiv preprint arXiv:1510.06096 (2015).
- Jain, Prateek and Kar, Purushottam. **Non-convex optimization for machine learning**. Foundations and Trends® in Machine Learning 10.3–4 (2017): 142–336.
- Chen, Yudong and Chi, Yuejie. **Harnessing structures in big data via guaranteed low-rank matrix estimation**. arXiv preprint arXiv:1802.08397 (2018).
- Chi, Yuejie and Lu, Yue M., and Chen, Yuxin. **Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview**. arXiv preprint arXiv:1809.09573 (2018).

Common ingredients in analysis

For **smooth** problems,

1st order geometry: ∇f or $v^\top \nabla f$ (directional derivatives)
2nd order geometry: $\nabla^2 f$ or $v^\top \nabla^2 f v$ (directional curvatures)

What about nonsmooth, nonconvex problems?

nonsmooth: may be non-differentiable

Nonsmooth problems are everywhere

Optimization: exact penalty functions

$$\min\ f(\boldsymbol{x})\ \text{s.t.}\ g_i(x) \le 0, h_j(x) = 0$$

$$\longrightarrow P(\boldsymbol{x}, c) = f(\boldsymbol{x}) + c\left(\sum_i g_i(x)_+ + \sum_j |h_j(\boldsymbol{x})|\right)$$

Nonsmooth problems are everywhere

Optimization: exact penalty functions

$$\min \ f(\boldsymbol{x}) \ \text{s.t.} \ g_i(x) \le 0, h_j(x) = 0$$

$$\longrightarrow P(\boldsymbol{x}, c) = f(\boldsymbol{x}) + c\left(\sum_i g_i(x)_+ + \sum_j |h_j(\boldsymbol{x})|\right)$$

Robust estimation:

$$\min_{\boldsymbol{x}} \ \left\|f\left(\boldsymbol{x}\right) - \boldsymbol{y}\right\|_p \qquad f \text{ nonlinear}$$

Nonsmooth problems are everywhere

Optimization: exact penalty functions

$$\min \; f(\boldsymbol{x}) \; \text{s.t.} \; g_i(x) \le 0, h_j(x) = 0$$

$$\longrightarrow P(\boldsymbol{x}, c) = f(\boldsymbol{x}) + c \left(\sum_i g_i(x)_+ + \sum_j |h_j(\boldsymbol{x})| \right)$$

Robust estimation:

$$\min_{\boldsymbol{x}} \; \|f(\boldsymbol{x}) - \boldsymbol{y}\|_p \qquad f \text{ nonlinear}$$

Promoting structures:

Sparse phase retrieval

Sparse principal component analysis (SPCA)

Sparse blind deconvolution

Neural networks with nonsmooth activations (e.g., ReLU)

Nonsmooth problems are everywhere

Optimization: exact penalty functions

$$\min \; f(\boldsymbol{x}) \;\text{s. t.}\; g_i(x) \le 0, h_j(x) = 0$$

$$\longrightarrow P(\boldsymbol{x}, c) = f(\boldsymbol{x}) + c \left(\sum_i g_i(x)_+ + \sum_j |h_j(\boldsymbol{x})| \right)$$

Robust estimation:

$$\min_{\boldsymbol{x}} \; \left\| f\left(\boldsymbol{x}\right) - \boldsymbol{y} \right\|_p \qquad f \text{ nonlinear}$$

Promoting structures:

- Sparse phase retrieval
- Sparse principal component analysis (SPCA)
- Sparse blind deconvolution
- Neural networks with nonsmooth activations (e.g., ReLU)

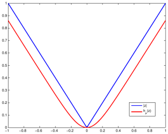
Others [Bagirov et al., 2014, Absil and Hosseini, 2017]

Smoothing?

$$\min \; f_1(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{j=1}^{m} |\boldsymbol{q}^*\boldsymbol{y}_j| \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

$\downarrow\downarrow$ smoothing $\downarrow\downarrow$

$$\min \; f(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{i=1}^{m} h_\mu\left(\boldsymbol{q}^*\boldsymbol{y}_i\right) \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$



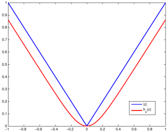
$$h_\mu\left(z\right) = \mu \log\cosh \frac{z}{\mu}$$

Smoothing?

$$\min\ f_1(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{j=1}^{m} |\boldsymbol{q}^*\boldsymbol{y}_j| \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

$\downarrow\downarrow$ smoothing $\downarrow\downarrow$

$$\min\ f(\boldsymbol{q}) \doteq \frac{1}{m}\sum_{i=1}^{m} h_\mu\left(\boldsymbol{q}^*\boldsymbol{y}_i\right) \text{ s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$



$$h_\mu(z) = \mu \log\cosh\frac{z}{\mu}$$

... at your own risk

Dear Ju Sun,

I regret to inform you that the editorial board has decided that your paper

"Complete dictionary recovery over the sphere"

By Ju Sun, Qing Qu, and John Wright

to Foundations of Computational Mathematics.

which you recently submitted to Foundations of Computational Mathematics cannot be accepted for publication in the journal, based on the statement of the handling editor which I attach below.

The paper will not be sent out for further refereeing.

Sincerely yours,

[redacted]

Handling editor statement: $\boxed{\text{The paper is unusually long (more than 100 pages) for JoFoCM}}$ which very rarely publish papers longer than 40-50 pages. The results are strong and there are a number of useful ideas in the paper for further research. I have no doubt that it would be accepted (modulo correctness of the proof, which I did not check in detail) by FOCM if it were of regular length. I do not see a good way to reduce it to FOCM-length without making the paper hard to read. It is long, but well written. That being said, I do not feel if it is that groundbreaking to overwrite the length restriction, more likely not.

Language for nonsmooth functions?

What about nonsmooth, nonconvex problems?

1st order geometry: ∇f or $v^\top \nabla f$ (directional derivatives)

$$\Longrightarrow \textbf{?}$$

2nd order geometry: $\nabla^2 f$ or $v^\top \nabla^2 f v$ (directional curvatures)

$$\Longrightarrow \textbf{?}$$

Locally Lipschitz (continuous) functions

... functions that are Lipschitz **locally**:

- Continuous convex and concave functions
- Continuously differentiable functions
- Distance function to a set
- Sum of two locally Lipschitz functions: e.g., weakly convex functions ($f(\boldsymbol{x})$ so that $f(\boldsymbol{x}) + \rho \|\boldsymbol{x}\|_2^2$ is convex)
- Products/Quotients of two locally Lipschitz functions
- Compositions of two locally Lipschitz functions: e.g., $h(g(\boldsymbol{x}))$ with h convex and $g \in \mathcal{C}^1$
- ...

Clarke subdifferentials

We restrict to **finite-dimensional** functions, i.e.,
$f : X \mapsto \mathbb{R}$ with $X \subset \mathbb{R}^n$.

Clarke subdifferentials

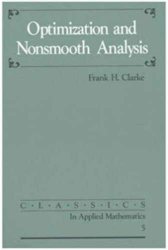
We restrict to **finite-dimensional** functions, i.e., $f : X \mapsto \mathbb{R}$ with $X \subset \mathbb{R}^n$.

Rademacher's theorem: If f is locally Lipschitz, f is differentiable almost everywhere in X.

Clarke subdifferentials

We restrict to **finite-dimensional** functions, i.e., $f : X \mapsto \mathbb{R}$ with $X \subset \mathbb{R}^n$.

Rademacher's theorem: If f is locally Lipschitz, f is differentiable almost everywhere in X.



Definition (Clarke subdifferential [Clarke, 1990])

$$\partial f(\boldsymbol{x}) \doteq \operatorname{conv}\{\boldsymbol{v} :\ \boldsymbol{x}_k \to \boldsymbol{x}, \nabla f(\boldsymbol{x}_k) \to \boldsymbol{v}, f \text{ diff. at } \boldsymbol{x}_k\}$$

... due to **Frank H. Clarke**. Well known in optimal control and economics

Clarke's subdifferential

For locally Lipschitz functions

Definition (Clarke subdifferential [Clarke, 1990])

$$\partial f\left(\boldsymbol{x}\right) \doteq \operatorname{conv}\left\{\boldsymbol{v}:\ \boldsymbol{x}_k \to \boldsymbol{x}, \nabla f\left(\boldsymbol{x}_k\right) \to \boldsymbol{v}, f \text{ diff. at } \boldsymbol{x}_k\right\}$$

- $\partial f\left(\boldsymbol{x}\right)$ is always nonempty, convex, and compact
- $f \in \mathcal{C}^1$, $\partial f\left(\boldsymbol{x}\right) = \left\{\nabla f\left(\boldsymbol{x}\right)\right\}$
- f convex: the usual subdifferential in convex analysis
- Most natural calculus rules hold (under regularity conditions, Chap 2 of [Clarke, 1990])
- Optimality: $\boldsymbol{x}_0$ is local min $\Longrightarrow \mathbf{0} \in \partial f\left(\boldsymbol{x}_0\right)$

Language for nonsmooth functions?

What about nonsmooth, nonconvex problems?

Language for nonsmooth functions?

What about nonsmooth, nonconvex problems?

1st order geometry: ∇f or $\boldsymbol{v}^\top \nabla f$ (directional derivatives)
$\implies \partial f$ or $\boldsymbol{v}^\top \partial f$

Language for nonsmooth functions?

What about nonsmooth, nonconvex problems?

1st order geometry: ∇f or $\boldsymbol{v}^\top \nabla f$ (directional derivatives)
$\Longrightarrow$ ∂f or $\boldsymbol{v}^\top \partial f$

2nd order geometry: $\nabla^2 f$ or $\boldsymbol{v}^\top \nabla^2 f \boldsymbol{v}$ (directional curvatures)
$\Longrightarrow$ motononicity of ∂f: f is convex **iff**

$\langle \boldsymbol{u_x} - \boldsymbol{u_y}, \boldsymbol{x} - \boldsymbol{y} \rangle \geq 0 \quad \forall\, \boldsymbol{x}, \boldsymbol{y}$ and $\forall\, \boldsymbol{u_x} \in \partial f\left(\boldsymbol{x}\right), \boldsymbol{u_y} \in \partial f\left(\boldsymbol{y}\right).$

Back to sparsifying transform learning

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

$$\boxed{\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m}\|\boldsymbol{q}^*\boldsymbol{Y}\|_1 = \frac{1}{m}\sum_i |\boldsymbol{q}^*\boldsymbol{y}_i| \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.}$$

Back to sparsifying transform learning

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

$$\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m}\|\boldsymbol{q}^*\boldsymbol{Y}\|_1 = \frac{1}{m}\sum_i |\boldsymbol{q}^*\boldsymbol{y}_i| \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

Riemannian language: $\partial_R f(\boldsymbol{q}) = (\boldsymbol{I} - \boldsymbol{q}\boldsymbol{q}^*)\,\partial f(\boldsymbol{q})$

[Hosseini and Uschmajew, 2017]

Back to sparsifying transform learning

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^* \boldsymbol{Y}$ is sparse, i.e., $\left\| \boldsymbol{Q}^* \boldsymbol{Y} \right\|_0$ is small.

$$\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m} \left\| \boldsymbol{q}^* \boldsymbol{Y} \right\|_1 = \frac{1}{m} \sum_i \left| \boldsymbol{q}^* \boldsymbol{y}_i \right| \quad \text{s.t.} \ \|\boldsymbol{q}\|_2^2 = 1.$$

Riemannian language: $\partial_R f(\boldsymbol{q}) = (\boldsymbol{I} - \boldsymbol{q}\boldsymbol{q}^*) \, \partial f(\boldsymbol{q})$

[Hosseini and Uschmajew, 2017]

For analysis: Bernoulli-Gaussian model $\boldsymbol{X}_0 = \boldsymbol{\Omega}_0 \circ \boldsymbol{V}_0$, $\boldsymbol{\Omega}_0 \sim_{\text{iid}} \mathrm{Ber}(\theta)$, $\boldsymbol{V}_0 \sim_{\text{iid}} \mathcal{N}(0,1)$. Sparsity parameter θ

Back to sparsifying transform learning

Given $\boldsymbol{Y}$, learn ortho $\boldsymbol{Q}$ s.t. $\boldsymbol{Q}^*\boldsymbol{Y}$ is sparse, i.e., $\|\boldsymbol{Q}^*\boldsymbol{Y}\|_0$ is small.

$$\min \quad f(\boldsymbol{q}) \doteq \frac{1}{m}\|\boldsymbol{q}^*\boldsymbol{Y}\|_1 = \frac{1}{m}\sum_i |\boldsymbol{q}^*\boldsymbol{y}_i| \quad \text{s.t. } \|\boldsymbol{q}\|_2^2 = 1.$$

Riemannian language: $\partial_R f(\boldsymbol{q}) = (\boldsymbol{I} - \boldsymbol{q}\boldsymbol{q}^*)\,\partial f(\boldsymbol{q})$

[Hosseini and Uschmajew, 2017]

For analysis: Bernoulli-Gaussian model $\boldsymbol{X}_0 = \boldsymbol{\Omega}_0 \circ \boldsymbol{V}_0$, $\boldsymbol{\Omega}_0 \sim_{\text{iid}} \mathrm{Ber}(\theta)$, $\boldsymbol{V}_0 \sim_{\text{iid}} \mathcal{N}(0,1)$. Sparsity parameter θ

When m is large, w.h.p., in a "reasonably large" region of $\boldsymbol{e}_n$:

$$\inf \langle \partial_R f(\boldsymbol{q}), \boldsymbol{q} - \boldsymbol{e}_n \rangle \geq \gamma \|\boldsymbol{q} - \boldsymbol{e}_n\|$$

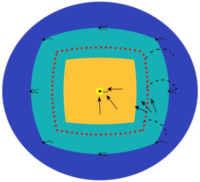


image credit: [Gilboa et al., 2018]

Subgradient descent learns orthogonal dictionaries!

Starting from a $\boldsymbol{q}^{(0)}$ **uniformly random** on $\mathbb{S}^{n-1}$, for $k = 0, 1, 2, \ldots$:

$$\boldsymbol{q}^{(k+1)} = \frac{\boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)}}{\left\| \boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)} \right\|} \quad \text{for any } \boldsymbol{v} \in \partial_R f\left(\boldsymbol{q}^{(k)}\right)$$

Subgradient descent learns orthogonal dictionaries!

Starting from a $\boldsymbol{q}^{(0)}$ **uniformly random** on $\mathbb{S}^{n-1}$, for $k = 0, 1, 2, \ldots$:

$$\boldsymbol{q}^{(k+1)} = \frac{\boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)}}{\left\| \boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)} \right\|} \quad \text{for any } \boldsymbol{v} \in \partial_R f\left(\boldsymbol{q}^{(k)}\right)$$

Ideas:

- Each runs finds an $\boldsymbol{e}_i$ with constant probability
- All basis vectors found in $O(n \log n)$ independent runs

Subgradient descent learns orthogonal dictionaries!

Starting from a $\boldsymbol{q}^{(0)}$ **uniformly random** on $\mathbb{S}^{n-1}$, for $k = 0, 1, 2, \ldots$:

$$\boldsymbol{q}^{(k+1)} = \frac{\boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)}}{\left\| \boldsymbol{q}^{(k)} - \eta^{(k)} \boldsymbol{v}^{(k)} \right\|} \quad \text{for any } \boldsymbol{v} \in \partial_R f \left(\boldsymbol{q}^{(k)} \right)$$

Ideas:

- Each runs finds an $\boldsymbol{e}_i$ with constant probability
- All basis vectors found in $O(n \log n)$ independent runs

Theorem (Informal)

Assume $\theta \in [1/n, 1/2]$. When $m \geq \Omega\left(\theta^{-2} n^4 \log^3 n\right)$, whp, the proposed algorithm recovers all basis vectors in polynomial time.

Comparison with the DL literature

Algorithms working in the constant sparsity regime, i.e., $\theta \in \Theta(1)$

– **Convex relaxation based on Sum-of-Squares (SOS)**:

[Barak et al., 2015, Ma et al., 2016b, Schramm and Steurer, 2017]

solving huge SDP's or tensor decompositions

Comparison with the DL literature

Algorithms working in the constant sparsity regime, i.e., $\theta \in \Theta(1)$

- **Convex relaxation based on Sum-of-Squares (SOS)**: [Barak et al., 2015, Ma et al., 2016b, Schramm and Steurer, 2017] solving huge SDP's or tensor decompositions
- **Nonconvex relaxation based on smoothed ℓ_1**: 2nd order method [Sun et al., 2015] or 1st order method [Gilboa et al., 2018], still expensive in computation and involved for analysis

Comparison with the DL literature

Algorithms working in the constant sparsity regime, i.e., $\theta \in \Theta(1)$

- **Convex relaxation based on Sum-of-Squares (SOS)**: [Barak et al., 2015, Ma et al., 2016b, Schramm and Steurer, 2017] solving huge SDP's or tensor decompositions
- **Nonconvex relaxation based on smoothed ℓ_1**: 2nd order method [Sun et al., 2015] or 1st order method [Gilboa et al., 2018], still expensive in computation and involved for analysis
- **This work: nonconvex relaxation based directly on ℓ_1**: lightweight computation and neater analysis — compress the smoothed ℓ_1 analysis by $1/2$!

A word on technicalities

Subdifferentials are (convex) sets in general, and randomness in data leads to **random sets**.

- Measure set difference: Hausdorff distance
- Expectation of random sets: selection integrals and support functions [Aubin and Frankowska, 2009, Molchanov, 2013]
- Concentration of Minkowski sum of random sets: support functions and concentration of empirical processes [Molchanov, 2017, Molchanov, 2013]

A word on technicalities

Subdifferentials are (convex) sets in general, and randomness in data leads to **random sets**.

- Measure set difference: Hausdorff distance
- Expectation of random sets: selection integrals and support functions [Aubin and Frankowska, 2009, Molchanov, 2013]
- Concentration of Minkowski sum of random sets: support functions and concentration of empirical processes [Molchanov, 2017, Molchanov, 2013]

The $\operatorname{sign}(\cdot)$ function is not Lipschitz in the usual sense

- Careful construction of the ε-net for covering in showing uniform convergence of the subdifferential

Easier ways around nonsmoothness?

Ignore non-differentiable points

- Pathological examples well known
- Performance on "generic" cases not understood

Easier ways around nonsmoothness?

Ignore non-differentiable points

- Pathological examples well known
- Performance on "generic" cases not understood

Smooth out and continue

- Relatively mature for convex problems [Nesterov, 2004]
- Lack in theory for nonconvex problems [Mobahi, 2013]

Easier ways around nonsmoothness?

Ignore non-differentiable points

- Pathological examples well known
- Performance on “generic” cases not understood

Smooth out and continue

- Relatively mature for convex problems [Nesterov, 2004]
- Lack in theory for nonconvex problems [Mobahi, 2013]

Tweak around subdifferential sets

- Intuitive chain rules [Kakade and Lee, 2018]
- Setting a predefined rule—might not be reliable in computation

So far ...

Tame nonconvexity = Live with and understand nonconvexity

So far ...

Tame nonconvexity = Live with and understand nonconvexity

Which nonconvex optimization problems are easy?

A1: problems with nice **global** landscapes

A2: problems with nice **local** landscapes

So far ...

Tame nonconvexity = Live with and understand nonconvexity

Which nonconvex optimization problems are easy?

A1: problems with nice **global** landscapes

A2: problems with nice **local** landscapes

What about nonsmooth, nonconvex problems?

we're still picking up the right language...

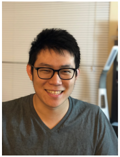
Thanks to ...

John Wright

Columbia

Qing Qu

NYU



Yu Bai

Stanford

Qijia Jiang

Stanford

E. Candès

Stanford

Block-Reference Coherent Diffraction Imaging, Barmherzig, **S.**, Lane, and Li, '18.

Landscape Analysis of Nonsmooth Functions, **S.** and Candès, '18.

Subgradient descent learns orthogonal dictionaries, Bai, Jiang, **S.** and Candès, '18.

Dictionary Learning in Fourier Transform Scanning Tunneling Spectroscopy, Cheung, Shin, Lau, Chen, **S.**, Zhang, Wright, and Pasupathy, '18.

A Geometric Analysis of Phase Retrieval, **S.**, Qu, Wright, '16

Complete Dictionary Recovery over the Sphere, **S.**, Qu, Wright, '15

When are Nonconvex Optimization Problems Not Scary?, **S.**, Qu, Wright, NIPS Workshop, '15

Finding a Sparse Vector in a Subspace: Linear Sparsity Using Alternating Directions, Qu, **S.**, Wright, '15

My webpage on provable nonconvex heuristics: `http://sunju.org/research/nonconvex/`

Thank you!

References i

[Absil and Hosseini, 2017] Absil, P. and Hosseini, S. (2017). **A collection of nonsmooth riemannian optimization problems.**

[Agarwal et al., 2013a] Agarwal, A., Anandkumar, A., Jain, P., Netrapalli, P., and Tandon, R. (2013a). **Learning sparsely used overcomplete dictionaries via alternating minimization.** *arXiv preprint arXiv:1310.7991*.

[Agarwal et al., 2013b] Agarwal, A., Anandkumar, A., and Netrapalli, P. (2013b). **Exact recovery of sparsely used overcomplete dictionaries.** *arXiv preprint arXiv:1309.1952*.

[Anandkumar et al., 2014a] Anandkumar, A., Ge, R., and Janzamin, M. (2014a). **Analyzing tensor power method dynamics: Applications to learning overcomplete latent variable models.** *arXiv preprint arXiv:1411.1488*.

[Anandkumar et al., 2014b] Anandkumar, A., Ge, R., and Janzamin, M. (2014b). **Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates.** *arXiv preprint arXiv:1402.5180*.

[Anandkumar et al., 2015] Anandkumar, A., Jain, P., Shi, Y., and Niranjan, U. N. (2015). **Tensor vs matrix methods: Robust tensor decomposition under block sparse perturbations.** *arXiv preprint arXiv:1510.04747*.

References ii

[Arora et al., 2014] Arora, S., Bhaskara, A., Ge, R., and Ma, T. (2014). **More algorithms for provable dictionary learning.** *arXiv preprint arXiv:1401.0579*.

[Arora et al., 2015] Arora, S., Ge, R., Ma, T., and Moitra, A. (2015). **Simple, efficient, and neural algorithms for sparse coding.** *arXiv preprint arXiv:1503.00778*.

[Arora et al., 2013] Arora, S., Ge, R., and Moitra, A. (2013). **New algorithms for learning incoherent and overcomplete dictionaries.** *arXiv preprint arXiv:1308.6273*.

[Aubin and Frankowska, 2009] Aubin, J.-P. and Frankowska, H. (2009). **Set-Valued Analysis.** Modern Birkhäuser Classics. Birkhäuser Basel.

[Bagirov et al., 2014] Bagirov, A., Karmitsa, N., and Mäkelä, M. M. (2014). **Introduction to Nonsmooth Optimization: theory, practice and software.** Springer.

[Bandeira et al., 2016] Bandeira, A. S., Boumal, N., and Voroninski, V. (2016). **On the low-rank approach for semidefinite programs arising in synchronization and community detection.** *arXiv preprint arXiv:1602.04426*.

[Barak et al., 2014] Barak, B., Kelner, J. A., and Steurer, D. (2014). **Dictionary learning and tensor decomposition via the sum-of-squares method.** *arXiv preprint arXiv:1407.1543*.

[Barak et al., 2015] Barak, B., Kelner, J. A., and Steurer, D. (2015). **Dictionary learning and tensor decomposition via the sum-of-squares method.** In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 143–151. ACM.

[Bhojanapalli et al., 2016] Bhojanapalli, S., Neyshabur, B., and Srebro, N. (2016). **Global optimality of local search for low rank matrix recovery.** *arXiv preprint arXiv:1605.07221*.

[Boumal, 2016] Boumal, N. (2016). **Nonconvex phase synchronization.** *arXiv preprint arXiv:1601.06114*.

[Candès et al., 2015] Candès, E. J., Li, X., and Soltanolkotabi, M. (2015). **Phase retrieval via wirtinger flow: Theory and algorithms.** *Information Theory, IEEE Transactions on*, 61(4):1985–2007.

[Chatterji and Bartlett, 2017] Chatterji, N. S. and Bartlett, P. L. (2017). **Alternating minimization for dictionary learning with random initialization.** *arxiv:1711.03634*.

[Chen and Candès, 2015] Chen, Y. and Candès, E. J. (2015). **Solving random quadratic systems of equations is nearly as easy as solving linear systems.** *arXiv preprint arXiv:1505.05114*.

References iv

[Chen et al., 2018] Chen, Y., Chi, Y., Fan, J., and Ma, C. (2018). **Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval.** *arXiv:1803.07726*.

[Chen and Wainwright, 2015] Chen, Y. and Wainwright, M. J. (2015). **Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees.** *arXiv preprint arXiv:1509.03025*.

[Clarke, 1990] Clarke, F. H. (1990). **Optimization and nonsmooth analysis, volume 5.** Siam.

[Conn et al., 2000] Conn, A. R., Gould, N. I. M., and Toint, P. L. (2000). **Trust-region Methods.** Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

[Ge et al., 2015] Ge, R., Huang, F., Jin, C., and Yuan, Y. (2015). **Escaping from saddle points—online stochastic gradient for tensor decomposition.** In *Proceedings of The 28th Conference on Learning Theory*, pages 797–842.

[Ge et al., 2016] Ge, R., Lee, J. D., and Ma, T. (2016). **Matrix completion has no spurious local minimum.** *arXiv preprint arXiv:1605.07272*.

[Geng and Wright, 2011] Geng, Q. and Wright, J. (2011). **On the local correctness of ℓ^1-minimization for dictionary learning.** Submitted to *IEEE Transactions on Information Theory*. Preprint: `http://www.columbia.edu/~jw2966`.

[Gilboa et al., 2018] Gilboa, D., Buchanan, S., and Wright, J. (2018). **Efficient dictionary learning with gradient descent.** *arXiv:1809.10313*.

[Goldfarb, 1980] Goldfarb, D. (1980). **Curvilinear path steplength algorithms for minimization which use directions of negative curvature.** *Mathematical programming*, 18(1):31–40.

[Gribonval and Schnass, 2010] Gribonval, R. and Schnass, K. (2010). **Dictionary identification - sparse matrix-factorization via ℓ^1-minimization.** *IEEE Transactions on Information Theory*, 56(7):3523–3539.

[Hardt, 2014] Hardt, M. (2014). **Understanding alternating minimization for matrix completion.** In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 651–660. IEEE.

[Hardt and Wootters, 2014] Hardt, M. and Wootters, M. (2014). **Fast matrix completion without the condition number.** In *Proceedings of The 27th Conference on Learning Theory*, pages 638–678.

References vi

[Hosseini and Uschmajew, 2017] Hosseini, S. and Uschmajew, A. (2017). **A Riemannian gradient sampling algorithm for nonsmooth optimization on manifolds.** *SIAM Journal on Optimization*, 27(1):173–189.

[Jain et al., 2010] Jain, P., Meka, R., and Dhillon, I. S. (2010). **Guaranteed rank minimization via singular value projection.** In *Advances in Neural Information Processing Systems*, pages 937–945.

[Jain and Netrapalli, 2014] Jain, P. and Netrapalli, P. (2014). **Fast exact matrix completion with finite samples.** *arXiv preprint arXiv:1411.1087*.

[Jain et al., 2013] Jain, P., Netrapalli, P., and Sanghavi, S. (2013). **Low-rank matrix completion using alternating minimization.** In *Proceedings of the forty-fifth annual ACM symposium on Theory of Computing*, pages 665–674. ACM.

[Jain and Oh, 2014] Jain, P. and Oh, S. (2014). **Provable tensor factorization with missing data.** In *Advances in Neural Information Processing Systems*, pages 1431–1439.

[Jin et al., 2017] Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. (2017). **How to escape saddle points efficiently.** *arXiv preprint arXiv:1703.00887*.

[Kakade and Lee, 2018] Kakade, S. and Lee, J. D. (2018). **Provably correct automatic subdifferentiation for qualified programs.** *arXiv:1809.08530*.

[Kawaguchi, 2016] Kawaguchi, K. (2016). **Deep learning without poor local minima.** *arXiv preprint arXiv:1605.07110*.

[Keshavan et al., 2010] Keshavan, R. H., Montanari, A., and Oh, S. (2010). **Matrix completion from a few entries.** *Information Theory, IEEE Transactions on*, 56(6):2980–2998.

[Lu and Kawaguchi, 2017] Lu, H. and Kawaguchi, K. (2017). **Depth creates no bad local minima.** *arXiv preprint arXiv:1702.08580*.

[Ma et al., 2016a] Ma, T., Shi, J., and Steurer, D. (2016a). **Polynomial-time tensor decompositions with sum-of-squares.** In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 438–446. IEEE.

[Ma et al., 2016b] Ma, T., Shi, J., and Steurer, D. (2016b). **Polynomial-time tensor decompositions with sum-of-squares.**

[Mairal et al., 2014] Mairal, J., Bach, F., and Ponce, J. (2014). **Sparse modeling for image and vision processing.** *Foundations and Trends in Computer Graphics and Vision*, 8(2-3):85–283.

[Mobahi, 2013] Mobahi, H. (2013). **Optimization by Gaussian smoothing with application to geometric alignment.** PhD thesis, University of Illinois at Urbana-Champaign.

[Molchanov, 2013] Molchanov, I. (2013). **Foundations of stochastic geometry and theory of random sets.** In *Stochastic Geometry, Spatial Statistics and Random Fields*, pages 1–20. Springer.

[Molchanov, 2017] Molchanov, I. (2017). **Theory of random sets.** Springer-Verlag London, 2nd edition.

[Murty and Kabadi, 1987] Murty, K. G. and Kabadi, S. N. (1987). **Some NP-complete problems in quadratic and nonlinear programming.** *Mathematical programming*, 39(2):117–129.

[Nesterov, 2004] Nesterov, Y. (2004). **Smooth minimization of non-smooth functions.** *Mathematical Programming*, 103(1):127–152.

[Nesterov and Polyak, 2006] Nesterov, Y. and Polyak, B. T. (2006). **Cubic regularization of newton method and its global performance.** *Mathematical Programming*, 108(1):177–205.

[Netrapalli et al., 2013] Netrapalli, P., Jain, P., and Sanghavi, S. (2013). **Phase retrieval using alternating minimization.** In *Advances in Neural Information Processing Systems*, pages 2796–2804.

[Netrapalli et al., 2014] Netrapalli, P., Niranjan, U. N., Sanghavi, S., Anandkumar, A., and Jain, P. (2014). **Non-convex robust PCA.** In *Advances in Neural Information Processing Systems*, pages 1107–1115.

[Ravishankar and Wohlberg, 2018] Ravishankar, S. and Wohlberg, B. (2018). **Learning multi-layer transform models.** *arXiv:1810.08323*.

[Rockafellar, 1993] Rockafellar, R. T. (1993). **Lagrange multipliers and optimality.** *SIAM review*, 35(2):183–238.

[Sa et al., 2015] Sa, C. D., Re, C., and Olukotun, K. (2015). **Global convergence of stochastic gradient descent for some non-convex matrix problems.** In *The 32nd International Conference on Machine Learning*, volume 37, pages 2332–2341.

[Schnass, 2014] Schnass, K. (2014). **Local identification of overcomplete dictionaries.** *arXiv preprint arXiv:1401.6354*.

[Schramm and Steurer, 2017] Schramm, T. and Steurer, D. (2017). **Fast and robust tensor decomposition with applications to dictionary learning.** *arXiv:1706.08672*.

[Soltanolkotabi et al., 2017] Soltanolkotabi, M., Javanmard, A., and Lee, J. D. (2017). **Theoretical insights into the optimization landscape of over-parameterized shallow neural networks.** *arXiv preprint arXiv:1707.04926*.

[Spielman et al., 2012] Spielman, D. A., Wang, H., and Wright, J. (2012). **Exact recovery of sparsely-used dictionaries.** In *Proceedings of the 25th Annual Conference on Learning Theory*.

[Sun et al., 2015] Sun, J., Qu, Q., and Wright, J. (2015). **Complete dictionary recovery over the sphere.** *arXiv preprint arXiv:1504.06785*.

[Sun et al., 2016] Sun, J., Qu, Q., and Wright, J. (2016). **A geometric analysis of phase retreival.** *arXiv preprint arXiv:1602.06664*.

[Sun and Luo, 2014] Sun, R. and Luo, Z.-Q. (2014). **Guaranteed matrix completion via non-convex factorization.** *arXiv preprint arXiv:1411.8003*.

[Tu et al., 2015] Tu, S., Boczar, R., Soltanolkotabi, M., and Recht, B. (2015). **Low-rank solutions of linear matrix equations via procrustes flow.** *arXiv preprint arXiv:1507.03566*.

[Wang et al., 2016] Wang, G., Giannakis, G. B., and Eldar, Y. C. (2016). **Solving systems of random quadratic equations via truncated amplitude flow.** *arxiv:1605.08285*.

[Wei et al., 2015] Wei, K., Cai, J.-F., Chan, T. F., and Leung, S. (2015). **Guarantees of Riemannian optimization for low rank matrix recovery.** *arXiv preprint arXiv:1511.01562*.

[White et al., 2015] White, C. D., Ward, R., and Sanghavi, S. (2015). **The local convexity of solving quadratic equations.** *arXiv preprint arXiv:1506.07868*.

[Zhang et al., 2017] Zhang, Y., Lau, Y., Kuo, H.-w., Cheung, S., Pasupathy, A., and Wright, J. (2017). **On the global geometry of sphere-constrained sparse blind deconvolution.** In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Zheng and Lafferty, 2015] Zheng, Q. and Lafferty, J. (2015). **A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements.** *arXiv preprint arXiv:1506.06081*.

[Zibulevsky and Pearlmutter, 2001] Zibulevsky, M. and Pearlmutter, B. (2001). **Blind source separation by sparse decomposition in a signal dictionary.** *Neural computation*, 13(4):863–882.