Stochastic optimization and sparse statistical recovery: An optimal algorithm for high dimensions

Alekh Agarwal
Microsoft Research

Joint work with Sahand Negahban and Martin Wainwright

Workshop on Optimization and Statistical Learning 2013, Les Houches, France

Introduction

- Sparse optimization:

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^d} \mathbb{E}_P[\ell(\theta; z)] = \arg\min_{\theta} \bar{\mathcal{L}}(\theta),$$

such that θ^* *is* s*-sparse*

Introduction

- Sparse optimization:

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^d} \mathbb{E}_P[\ell(\theta; z)] = \arg\min_{\theta} \bar{\mathcal{L}}(\theta),$$

such that θ^* *is* s*-sparse*

- Loss function ℓ is convex
- P unknown, can sample from it
- High dimensional setup: $n \ll d$

Introduction

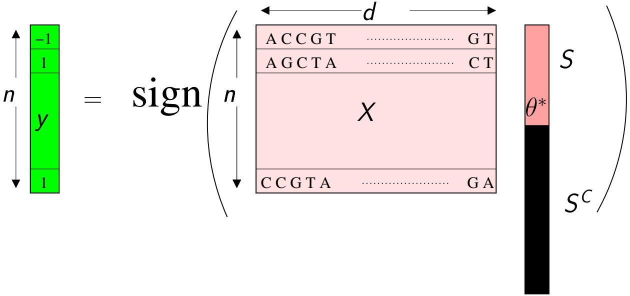
- Sparse optimization:

$$\theta^* = \arg\min_{\theta \in \mathbb{R}^d} \mathbb{E}_P[\ell(\theta; z)] = \arg\min_{\theta} \bar{\mathcal{L}}(\theta),$$

such that θ^* is s-sparse

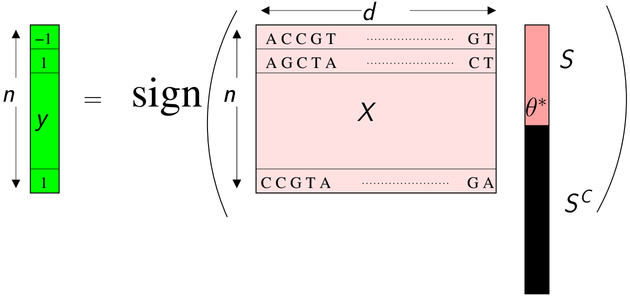
- Loss function ℓ is convex
- P unknown, can sample from it
- High dimensional setup: $n \ll d$
- **Want linear time and statistically (near) optimal algorithm**

Example 1 : Computational genomics



- Predict disease susceptibility from genome
- Depends on very few genes, θ^* is sparse

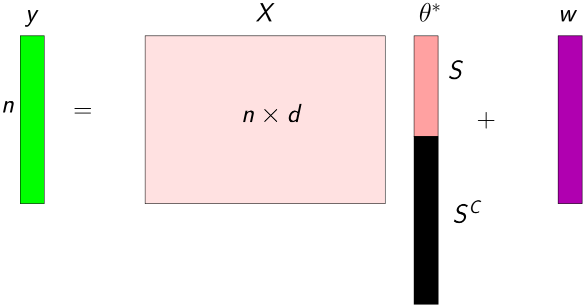
Example 1 : Computational genomics



- Predict disease susceptibility from genome
- Depends on very few genes, θ^* is sparse
- Sparse logistic regression:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_P[\log(1 + \exp(-y\theta^T x))].$$

Example 2 : Compressed sensing



- Recover unknown signal θ^* from noisy measurements
- Sparse linear regression:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_P[(y - \theta^T x)^2].$$

Approach 1: M-estimation (batch optimization)

- Draw n i.i.d. samples
- Obtain $\widehat{\theta}_n$

$$\widehat{\theta}_n = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; z_i) + \lambda_n \|\theta\|_1$$

Approach 1: M-estimation (batch optimization)

- Draw n i.i.d. samples
- Obtain $\widehat{\theta}_n$

$$\widehat{\theta}_n = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \ell(\theta; z_i) + \lambda_n \|\theta\|_1$$

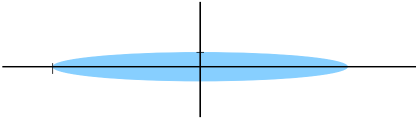
- Statistical arguments for consistency, $\widehat{\theta}_n \to \theta^*$
- Convex optimization to compute $\widehat{\theta}_n$

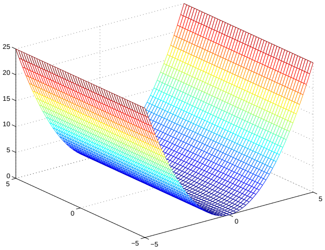
Batch optimization

- Convergence depends on properties of

$$\frac{1}{n}\sum_{i=1}^{n}\ell(\theta; z_i) + \lambda_n \|\theta\|_1$$

- Sample loss not (globally) strongly convex for $n < d$
- Poor smoothness when $n \ll d$



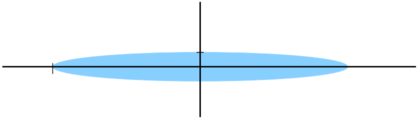


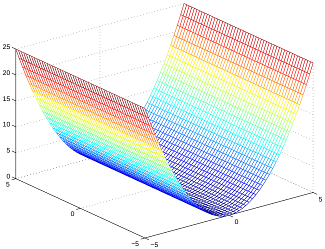
Batch optimization

- Convergence depends on properties of

$$\frac{1}{n}\sum_{i=1}^{n}\ell(\theta; z_i) + \lambda_n\|\theta\|_1$$

- Sample loss not (globally) strongly convex for $n < d$
- Poor smoothness when $n \ll d$

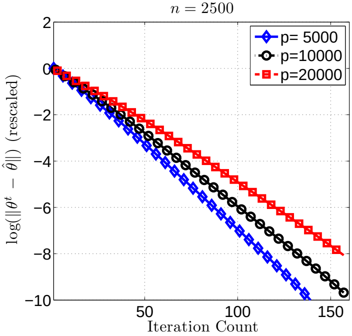




- But, smooth and strongly convex in *sparse directions*
    - Example: Least-squares loss with random design

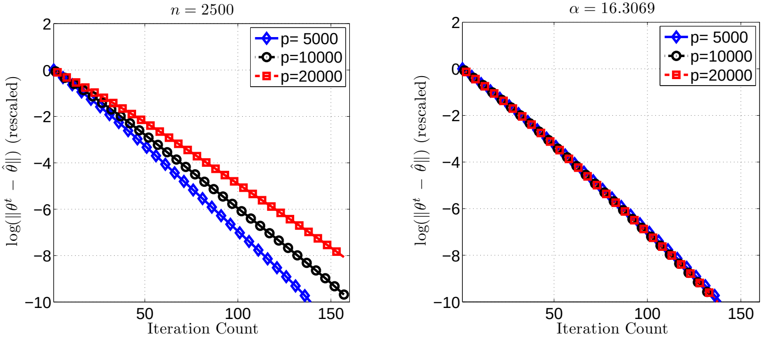
Fast convergence of gradient descent

- We prove (global) linear convergence of gradient descent based on *sparse condition number* of $\frac{1}{n}\sum_{i=1}^{n}\ell(\theta; z_i)$



Fast convergence of gradient descent

- We prove (global) linear convergence of gradient descent based on *sparse condition number* of $\frac{1}{n}\sum_{i=1}^{n} \ell(\theta; z_i)$



Computational complexity of batch optimization

- Convergence rate captures number of iterations
- Each iteration has complexity $\mathcal{O}(nd)$
- One pass over data at each iteration

Computational complexity of batch optimization

- Convergence rate captures number of iterations
- Each iteration has complexity $\mathcal{O}(nd)$
- One pass over data at each iteration
- *But we wanted linear time algorithm!*

Approach 2: Stochastic optimization

- Directly minimize $\mathbb{E}_P[\ell(\theta; z)]$
- Use samples to obtain gradient estimates

$$\theta^{t+1} = \theta^t - \alpha_t \nabla \ell(\theta^t; z_t)$$

Approach 2: Stochastic optimization

- Directly minimize $\mathbb{E}_P[\ell(\theta; z)]$
- Use samples to obtain gradient estimates

$$\theta^{t+1} = \theta^t - \alpha_t \nabla \ell(\theta^t; z_t)$$

- Stop after one pass over data
- Statistically, often competitive with batch (that is, $\|\theta^n - \theta^*\|^2 \approx \|\widehat{\theta}_n - \theta^*\|^2$)
- Precise rates depend on the problem structure

Structural assumptions

- θ^* is s-sparse
- Make additional structural assumptions on $\bar{\mathcal{L}}(\theta) = \mathbb{E}_P[\ell(\theta; z)]$
  - $\bar{\mathcal{L}}$ is Locally Lipschitz
  - $\bar{\mathcal{L}}$ is Locally strongly convex (LSC)

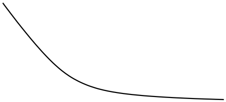
Locally Lipschitz functions

Definition (Locally Lipschitz function)

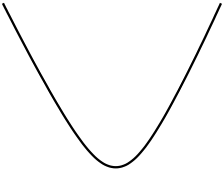
$\bar{\mathcal{L}}$ is locally G-Lipschitz in ℓ_1-norm, meaning that

$$|\bar{\mathcal{L}}(\theta) - \bar{\mathcal{L}}(\tilde{\theta})| \leq G\|\theta - \tilde{\theta}\|_1,$$

if $\|\theta - \theta^*\|_1 \leq R$ and $\|\tilde{\theta} - \theta^*\|_1 \leq R$.



Globally Lipschitz



Locally Lipschitz

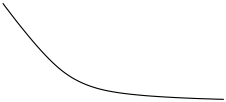
Locally strongly convex functions

Definition (Locally strongly convex function)

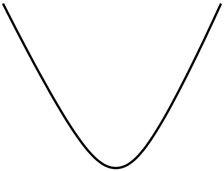
There is a constant $\gamma > 0$ such that

$$\bar{\mathcal{L}}(\tilde{\theta}) \geq \bar{\mathcal{L}}(\theta) + \langle \nabla \bar{\mathcal{L}}(\theta), \tilde{\theta} - \theta \rangle + \frac{\gamma}{2} \|\theta - \tilde{\theta}\|_2^2,$$

if $\|\theta\|_1 \leq R$ and $\|\tilde{\theta}\|_1 \leq R$



Locally Strongly convex



Globally strongly convex

Stochastic optimization and structural conditions

Method	Sparsity	LSC	Convergence
SGD	×	✓	$\mathcal{O}\left(\frac{d}{T}\right)$
Mirror descent/RDA/FOBOS/COMID	✓	×	$\mathcal{O}\left(\sqrt{\frac{s^2 \log d}{T}}\right)$
Our Method	✓	✓	$\mathcal{O}\left(\frac{s \log d}{T}\right)$

Some previous methods

- All methods based on observing g^t such that $\mathbb{E}[g^t] \in \partial \bar{\mathcal{L}}(\theta^t)$
- **Stochastic gradient descent:** based on ℓ_2 distances, *exploits LSC*

$$\theta^{t+1} = \arg\min_{\theta} \langle g^t, \theta \rangle + \frac{1}{2\alpha_t} \|\theta - \theta^t\|_2^2$$

Some previous methods

- All methods based on observing g^t such that $\mathbb{E}[g^t] \in \partial\bar{\mathcal{L}}(\theta^t)$
- **Stochastic gradient descent:** based on ℓ_2 distances, *exploits LSC*

$$\theta^{t+1} = \arg\min_{\theta} \langle g^t, \theta \rangle + \frac{1}{2\alpha_t} \|\theta - \theta^t\|_2^2$$

- **Stochastic dual averaging:** based on ℓ_p distances, *exploits sparstity when* $p \approx 1$

$$\theta^{t+1} = \arg\min_{\theta} \sum_{s=1}^{t} \langle g^s, \theta \rangle + \frac{1}{2\alpha_t} \|\theta\|_p^2$$

- Need to reconcile the geometries for exploiting both structures

RADAR algorithm: outline

- Based on Juditsky and Nesterov (2011)
- Recall the minimization problem: $\min_\theta \mathbb{E}[\ell(\theta; z)]$
- Algorithm proceeds over K epochs
- At epoch i, solve the regularized problem:

$$\min_{\theta \in \Omega_i} \mathbb{E}[\ell(\theta; z)] + \lambda_i \|\theta\|_1$$

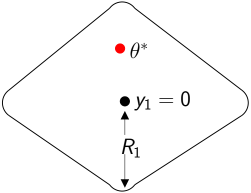
- where $\Omega_i = \theta \in \mathbb{R}^d \;\; : \;\; \|\theta - y_i\|_p^2 \leq R_i^2$

RADAR algorithm: First epoch

- Require: R_1 such that $\|\theta^*\|_1 \leq R_1$
- Perform stochastic dual averaging with $p = \frac{2\log d}{2\log d - 1} \approx 1$

  - Initialize $\theta^1 = 0$, $y_1 = 0$
  - Observe g^t where $\mathbb{E}[g^t] \in \partial\bar{\mathcal{L}}(\theta^t)$ and $\nu^t \in \partial\|\theta^t\|_1$
  - Update

$$
\begin{aligned}
\mu^{t+1} &= \mu^t + g^t + \lambda_1 \nu^t \\
\theta^{t+1} &= \arg\min_{\|\theta\|_p \leq R_1} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t}\|\theta\|_p^2
\end{aligned}
$$

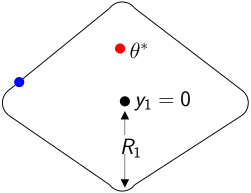


RADAR algorithm: First epoch

- Require: R_1 such that $\|\theta^*\|_1 \leq R_1$
- Perform stochastic dual averaging with $p = \frac{2\log d}{2\log d - 1} \approx 1$

- Initialize $\theta^1 = 0,\ y_1 = 0$
- Observe g^t where $\mathbb{E}[g^t] \in \partial\bar{\mathcal{L}}(\theta^t)$ and $\nu^t \in \partial\|\theta^t\|_1$
- Update

$$\begin{aligned} \mu^{t+1} &= \mu^t + g^t + \lambda_1 \nu^t \\ \theta^{t+1} &= \arg\min_{\|\theta\|_p \leq R_1} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t}\|\theta\|_p^2 \end{aligned}$$

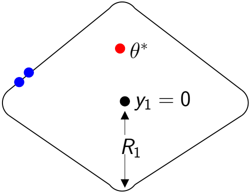


RADAR algorithm: First epoch

- Require: R_1 such that $\|\theta^*\|_1 \le R_1$
- Perform stochastic dual averaging with $p = \frac{2\log d}{2\log d - 1} \approx 1$

  - Initialize $\theta^1 = 0$, $y_1 = 0$
  - Observe g^t where $\mathbb{E}[g^t] \in \partial \bar{\mathcal{L}}(\theta^t)$ and $\nu^t \in \partial \|\theta^t\|_1$
  - Update

$$
\begin{aligned}
\mu^{t+1} &= \mu^t + g^t + \lambda_1 \nu^t \\
\theta^{t+1} &= \arg \min_{\|\theta\|_p \le R_1} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t} \|\theta\|_p^2
\end{aligned}
$$

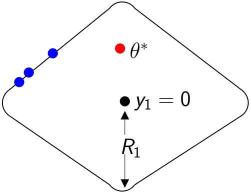


RADAR algorithm: First epoch

- Require: R_1 such that $\|\theta^*\|_1 \leq R_1$
- Perform stochastic dual averaging with $p = \frac{2\log d}{2\log d - 1} \approx 1$

- Initialize $\theta^1 = 0$, $y_1 = 0$
- Observe g^t where $\mathbb{E}[g^t] \in \partial \bar{\mathcal{L}}(\theta^t)$ and $\nu^t \in \partial\|\theta^t\|_1$
- Update

$$
\begin{aligned}
\mu^{t+1} &= \mu^t + g^t + \lambda_1 \nu^t \\
\theta^{t+1} &= \arg\min_{\|\theta\|_p \leq R_1} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t}\|\theta\|_p^2
\end{aligned}
$$

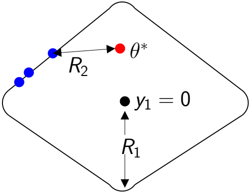


RADAR algorithm: First epoch

- Require: R_1 such that $\|\theta^*\|_1 \leq R_1$
- Perform stochastic dual averaging with $p = \frac{2\log d}{2\log d - 1} \approx 1$

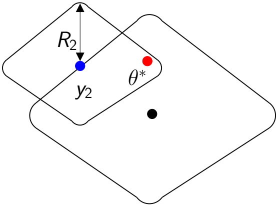
- Initialize $\theta^1 = 0,\ y_1 = 0$
- Observe g^t where $\mathbb{E}[g^t] \in \partial\bar{\mathcal{L}}(\theta^t)$ and $\nu^t \in \partial\|\theta^t\|_1$
- Update

$$\begin{aligned} \mu^{t+1} &= \mu^t + g^t + \lambda_1 \nu^t \\ \theta^{t+1} &= \arg\min_{\|\theta\|_p \leq R_1} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t}\|\theta\|_p^2 \end{aligned}$$



Initializing next epoch

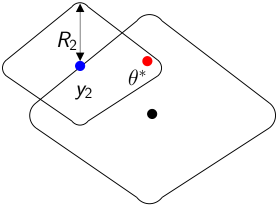
- Update $y_2 = \bar{\theta}_T$
- Update $R_2^2 = R_1^2/2$
- Update $\lambda_2 = \lambda_1/\sqrt{2}$
- Initialize $\theta^1 = y_2$ for next epoch



Initializing next epoch

- Update $y_2 = \bar{\theta}_T$
- Update $R_2^2 = R_1^2/2$
- Update $\lambda_2 = \lambda_1/\sqrt{2}$
- Initialize $\theta^1 = y_2$ for next epoch
- Now use updates

$$\begin{aligned} \mu^{t+1} &= \mu^t + g^t + \lambda_2 \nu^t \\ \theta^{t+1} &= \arg \min_{\|\theta - y_2\|_p \le R_2} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t} \|\theta - y_2\|_p^2 \end{aligned}$$

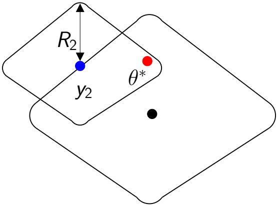


Initializing next epoch

- Update $y_2 = \bar{\theta}_T$
- Update $R_2^2 = R_1^2/2$
- Update $\lambda_2 = \lambda_1/\sqrt{2}$
- Initialize $\theta^1 = y_2$ for next epoch
- Now use updates

$$\begin{aligned} \mu^{t+1} &= \mu^t + g^t + \lambda_2 \nu^t \\ \theta^{t+1} &= \arg\min_{\|\theta - y_2\|_p \leq R_2} \langle \theta, \mu^{t+1} \rangle + \frac{1}{2\alpha_t} \|\theta - y_2\|_p^2 \end{aligned}$$

Each step still $\mathcal{O}(d)$



Convergence rate for exact sparsity

Theorem

Suppose the expected loss is G-Lipschitz and γ-strongly convex. Suppose θ^* has at most s non-zero entries. With probability at least $1 - 6\exp(-\delta \log d/12)$

$$\|\bar{\theta}_T - \theta^*\|_2^2 \leq c\, \frac{G^2 + \sigma^2(1+\delta)}{\gamma^2}\, \frac{s \log d}{T}.$$

- Logarithmic scaling in d
- Error decays as $1/T$
- Results extend to approximately sparse problems

Convergence rate for exact sparsity

Theorem

Suppose the expected loss is G-Lipschitz and γ-strongly convex. Suppose θ^* has at most s non-zero entries. With probability at least $1 - 6\exp(-\delta \log d/12)$

$$\|\bar{\theta}_T - \theta^*\|_2^2 \leq c\, \frac{G^2 + \sigma^2(1+\delta)}{\gamma^2}\, \frac{s \log d}{T}.$$

- Logarithmic scaling in d
- Error decays as $1/T$
- Results extend to approximately sparse problems
- Similar result for the method of Juditsky and Nesterov (2011) applied with a fixed λ

Optimality of results

- Error of $\mathcal{O}\left(\frac{s \log d}{\gamma^2 T}\right)$ after T iterations
- Stochastic gradients computed with one sample
- T iterations $\equiv$ T samples
- Information-theoretic limit: Error $\Omega\left(\frac{s \log d}{\gamma^2 T}\right)$ after observing T samples for *any possible method*

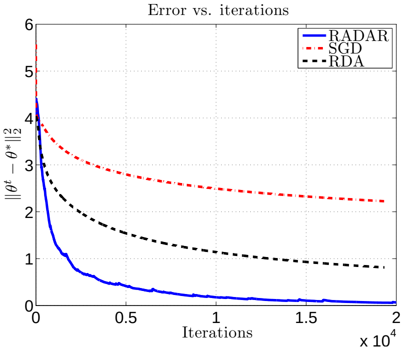
Optimality of results

- Error of $\mathcal{O}\left(\frac{s \log d}{\gamma^2 T}\right)$ after T iterations
- Stochastic gradients computed with one sample
- T iterations $\equiv$ T samples
- Information-theoretic limit: Error $\Omega\left(\frac{s \log d}{\gamma^2 T}\right)$ after observing T samples for *any possible method*
- **We obtain the best possible error in linear time**

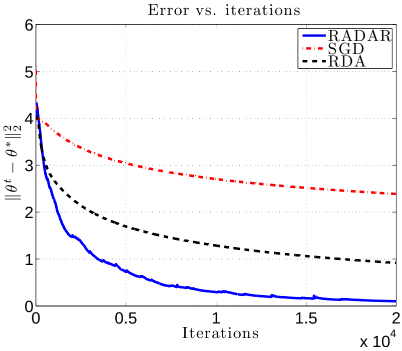
Simulation results

- Performed simulations for sparse linear regression
- Compared to classical benchmarks: RDA, SGD
- Evaluated several versions: RADAR, EDA, RADAR-Const
- Results averaged over 5 random trials

Simulation results

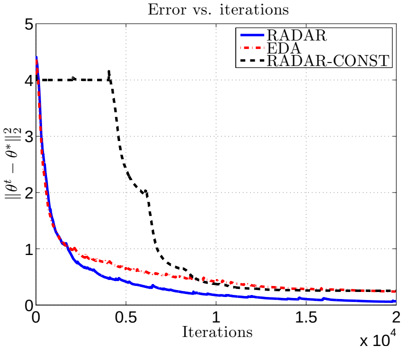


$d = 20000$

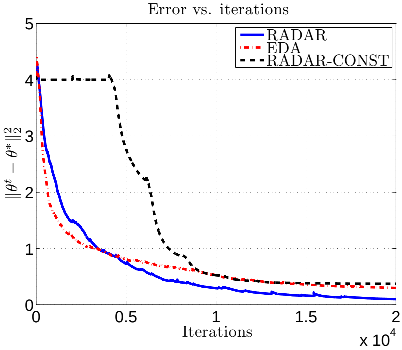


$d = 40000$

Simulation results



$d = 20000$



$d = 40000$

Intuition

- Convergence rate of $1/\sqrt{t}$ within each epoch
- Re-centering and shrinking of set boosts convergence speed at each epoch
- Error halved after each epoch
- Epoch lengths double— initial epochs negligible
- Fast convergence at later epochs due to small set
- High regularization initially, little at the end leads to (aprpox.) sparsity all along

Conclusions

- Stochastic optimization algorithm for sparse, high-dimensional problems
- Simultaneously exploits sparsity and strong convexity of the problem
- Optimal rate of convergence
- Updates computed in closed form for common problems
- Extends to group sparsity, low-rank etc.
- Similar extensions for mirror descent, accelerated methods (Hazan and Kale (2011), Ghadimi and Lan (2012))
- Possible extensions to distributed settings

More details can be found in

- Fast global convergence of gradient methods for high dimensional statistical recovery, A., Negahban and Wainwright, `http://arxiv.org/abs/1104.4824`.
- Stochastic optimization and sparse statistical recovery: An optimal algorithm for high dimensions, A., Negahban and Wainwright, `http://arxiv.org/abs/1207.4421`.

Thank You