OBProx-SG: Orthant-Based Proximal Stochastic Gradient Method for ℓ_1-Regularized Problem

Tianyi Chen

Microsoft

Collaborators:
Tianyu Ding (Johns Hopkins University)
Bo Ji (Zhejiang University)
Guanyi Wang (Georgia Institute of Technology)
Zhihui Zhu (University of Denver)

Outline

Outline

The optimization problem.

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \left\{F(x) \stackrel{\text{def}}{=} \underbrace{\frac{1}{N}\sum_{i=1}^{N} f_i(x)}_{f(x)} + \lambda\|x\|_1\right\}$$

- Finite-sum problem, N is huge, all f_i is continuously differentiable and $\lambda > 0$.
- The solution tends to have high sparsity and low objective function value under proper λ.

Several examples of $f(x)$:

- Logistic Loss:

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} \log(1 + e^{-y_i x^T d_i})$$

- Neural Network:

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} (W_{L+1}\sigma(W_L \cdots \sigma(W_1 x_i + c_1) \cdots + c_L) + c_{L+1} - y_i)^2$$

with $\sigma(\cdot)$ any activation function.

The optimization problem.

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ \frac{1}{N}\sum_{i=1}^{N} f_i(x) + \lambda\|x\|_1$$

- All f_i is continuously differentiable, and $\lambda > 0$

Different approaches:

- Well studied in deterministic optimization, *i.e.*, high sparse solutions with low objective function values, with numerous methods:
  - first-order : steepest descent (minimum norm element of subdifferential)
    proximal full gradient method (Prox-FG), ISTA/FISTA (Beck and Teboulle)
  - second-order :
    - proximal Newton : LIBLINEAR (newGLMNET)
    - orthant-based : FaRSA (Chen, Curtis, and Robinson), OBA (Keskar, Nocedal, Öztoprak, and Wächter)
- Limited studied in stochastic optimization, *i.e.*, solutions with low objective function values but typically with low sparsity with several methods:
  - Prox-SG and its variants, *e.g.*, RDA and Prox-SVRG (Xiao)

Our contributions

- Propose the Orthant Based Proximal Stochastic Gradient Method (OBProx-SG) effectively to achieve solutions of both high sparsity and low objective function value in stochastic settings.
- OBProx-SG utilizes a Prox-SG Step to predict a support cover of the solution to construct an orthant face and an Orthant Step to effectively exploit the sparsity.
- Outperform other state-of-the-art methods comprehensively on sparsity exploration and objective convergence and computational cost.

	OBProx-SG	Prox-SG	RDA	Prox-SVRG
Sparsity Exploration	✓	–	–	–
Objective Convergence	✓	✓	–	✓
Computational Cost	✓	✓	✓	–

Remark: on deep learning experiments, with the same accuracy, the solutions by OBProx-SG usually possess multiple-times higher sparsity than others.

- **Applications:** Feature selection and model compression. (The sparsity can be used as compression ratio. Two heavy AI products on Microsoft AI Cognitive Service has been dramatically compressed via OBProx-SG without accuracy regression and successfully deployed.)

Outline

Target problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \ \frac{1}{N}\sum_{i=1}^{N} f_i(x) + \lambda\|x\|_1 \tag{1}$$

- Under proper λ, its optimal solution x^* is highly sparse (including many zero elements).

How to identify correct zero variables in the solution?

$$\mathcal{I}^0(x) := \{i : [x]_i = 0\}, \mathcal{I}^{\neq 0}(x) := \{i : [x]_i \neq 0\}$$

Two Steps:

Prox-SG Step: Predict a non-zero element cover (support cover) of optimal solutions. $|\mathcal{I}^{\neq 0}(x_k)| \gg |\mathcal{I}^{\neq 0}(x^*)|$ in stochastic setting.

Orthant Step: Exploit the sparsity on the predicted non-zero elements.

A switch: Select **Prox-SG Step** or **Orthant Step**.

Outline of OBProx-SG

Algorithm 1 OBProx-SG

Input: $x_0 \in \mathbb{R}^n$, $\alpha_0 \in (0, 1)$.
for $k = 0, 1, 2, \ldots$ **do**
  Switch Prox-SG Step or Orthant Step by Algorithm 4.
  if Prox-SG Step is selected **then**
    Compute the Prox-SG Step update:
    $x_{k+1} \leftarrow \text{Prox-SG}(x_k, \alpha_k)$ by Algorithm 2.
  else if Orthant Step is selected **then**
    Compute the Orthant Step update:
    $x_{k+1} \leftarrow \text{Orthant}(x_k, \alpha_k)$ by Algorithm 3.
  Update α_{k+1} given α_k according to some rule.

Prox-SG Step : Predict support cover (non-zero elements).

Algorithm 2 Prox-SG Step

1: **Input:** Current iterate x_k, and step size α_k.
2: Compute the stochastic gradient of f on $\mathcal{B}_k$

$$\nabla f_{\mathcal{B}_k}(x_k) \leftarrow \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \nabla f_i(x_k). \tag{2}$$

3: **Return** $x_{k+1} \leftarrow \text{Prox}_{\alpha_k \lambda \|\cdot\|_1} \left(x_k - \alpha_k \nabla f_{\mathcal{B}_k}(x_k) \right)$.

$$x_{k+1} = \text{Prox}_{\alpha_k \lambda \|\cdot\|_1}(x_k - \alpha_k \nabla f_{\mathcal{B}_k}(x_k)) = \underset{x \in \mathbb{R}^n}{\text{argmin}} \ \frac{1}{2\alpha_k} \|x - (x_k - \alpha_k \nabla f_{\mathcal{B}_k}(x_k))\|_2^2 + \lambda \|x\| \tag{3}$$

Denote the trial iterate $\widehat{x}_{k+1} := x_k - \alpha_k \nabla f_{\mathcal{B}_k}(x_k)$, then x_{k+1} is computed efficiently as:

$$[x_{k+1}]_i = \begin{cases} [\widehat{x}_{k+1}]_i - \alpha_k \lambda, & \text{if } [\widehat{x}_{k+1}]_i > \alpha_k \lambda \\ [\widehat{x}_{k+1}]_i + \alpha_k \lambda, & \text{if } [\widehat{x}_{k+1}]_i < -\alpha_k \lambda \\ 0, & \text{otherwise} \end{cases} . \tag{4}$$

Prox-SG Step : Predict support cover (non-zero elements).

Comments:

- The Prox-SG step has a sparsity promition mechanism to project variables to zero if trial iterates falls into an interval $[-\alpha_k\lambda, \alpha_k\lambda]$ (**Projection region**).
- Due to stochastic nature and the small α_k selection in stochastic problems, rare variables are projected to zero.
- The predicted non-zero elements typically much more than the exact support of the solutions:

$$|\mathcal{I}^{\neq 0}(x_k)| \gg |\mathcal{I}^{\neq 0}(x^*)|.$$

Orthant Step : We define the orthant face $\mathcal{O}_k$ that x_k lies in to be

$$\mathcal{O}_k := \{x \in \mathbb{R}^n : \text{sign}([x]_i) = \text{sign}([x_k]_i) \text{ or } [x]_i = 0, 1 \leq i \leq n\} \tag{5}$$

$F(x)$ can be written precisely as a smooth function $\widetilde{F}(x)$ on $\mathcal{O}_k$ in the form

$$F(x) \equiv \widetilde{F}(x) := f(x) + \lambda \text{sign}(x_k)^T x, \tag{6}$$

$$\underset{x \in \mathcal{O}_k}{\text{minimize}} \, \tilde{F}(x)$$

Algorithm 3 Orthant Step.

1: **Input:** Current iterate x_k, and step size α_k.
2: Compute the stochastic gradient of $\widetilde{F}$ on $\mathcal{B}_k$

$$\nabla \widetilde{F}_{\mathcal{B}_k}(x_k) \leftarrow \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \nabla \widetilde{F}_i(x_k) \tag{7}$$

3: **Return** $x_{k+1} \leftarrow \text{Proj}_{\mathcal{O}_k}(x_k - \alpha_k \nabla \widetilde{F}_{\mathcal{B}_k}(x_k))$.

$\text{Proj}_{\mathcal{O}_k}(\cdot)$ defined as $\left[\text{Proj}_{\mathcal{O}_k}(z)\right]_i := \begin{cases} [z]_i & \text{if sign}([z]_i) = \text{sign}([x_k]_i) \\ 0 & \text{otherwise} \end{cases}$.

Illustration of Orthant Step

Assume $x \in \mathbb{R}^3$.

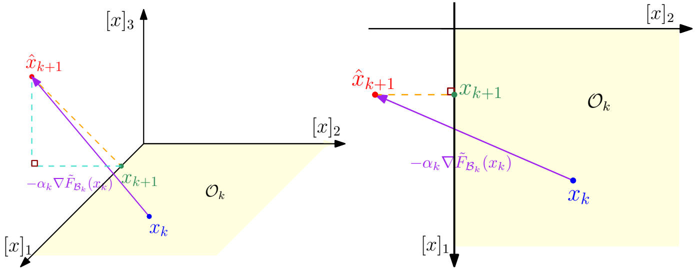


Figure: Illustration of Orthant Step with projection, where $\mathcal{O}_k = \{x \in \mathbb{R}^3 : [x]_1 \geq 0, [x]_2 \geq 0, [x]_3 = 0\}$. (L): 3D view. (R): top view.

- x_{k+1} is more sparser than x_k due to $[x_{k+1}]_2 = 0$.

Projection Region Comparison

- Orthant Step is a more aggressive sparsity promotion mechanism than SOTA.
- It enjoys a much large projection region than others while still maintains convergence characteristic.

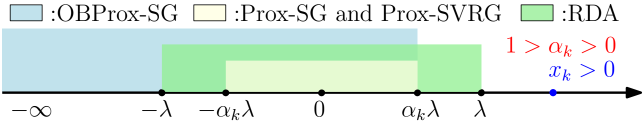


Figure: Projection regions of different methods for 1D case at $x_k > 0$.

Projection region: the region that projects trial iterate to zero if it falls in.

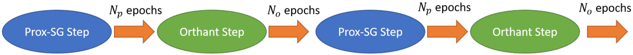
Switching Mechanism

Algorithm 4 Switching Mechanism.

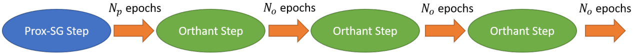
Input: $k, N_{\mathcal{P}}, N_{\mathcal{O}}$.
if $\mathrm{mod}(k, N_{\mathcal{P}} + N_{\mathcal{O}}) < N_{\mathcal{P}}$ **then**
  Return Prox-SG Step is selected.
else
  Return Orthant Step is selected.

Convergence analysis supports: either

- Alternatively employ Prox-SG Step and Orthant Step; or



- Employ Prox-SG Step sufficiently many time, then stick on Orthant Step until the end. Referred as OBProx-SG+.



OBProx-SG+ is recommended due to its attractive property of maintaining sparsity exploration.

Outline

Convergence under alternating schema

We define the gradient mapping as follows

$$\mathcal{G}_\eta(x) = \frac{1}{\eta}\left(x - \text{Prox}_{\eta\lambda\|\cdot\|_1}(x - \eta\nabla f(x))\right). \tag{8}$$

Theorem 1

Suppose $N_{\mathcal{P}} < \infty$ and $N_{\mathcal{O}} < \infty$.

1. the step size $\{\alpha_k\}$ is $\mathcal{O}(1/k)$, then $\liminf_{k\to\infty} \mathbb{E}\|\mathcal{G}_{\alpha_k}(x_k)\|_2^2 = 0$.
2. f is μ-strongly convex, and $\alpha_k \equiv \alpha$ for any $\alpha < \min\{\frac{1}{2\mu}, \frac{1}{L}\}$, then

$$\mathbb{E}[F(x_{k+1}) - F^*] \leq (1 - 2\alpha\mu)^{\kappa_{\mathcal{P}}}[F(x_0) - F^*] + \frac{LC^2}{2\mu}\alpha, \tag{9}$$

where $\kappa_{\mathcal{P}}$ is the number of Prox-SG Steps employed until k-th iteration.

Convergence under practical plus schema

In practice:

- Repeatedly switch back to Prox-SG Step since most likely it is going to ruin the sparsity from the previous iterates by Orthant Step due to the stochastic nature.
- OBProx-SG+ is preferred, *i.e.*, $N_{\mathcal{P}} < \infty$, $N_{\mathcal{O}} = \infty$.

Theorem 2

Suppose $N_{\mathcal{P}} < \infty, N_{\mathcal{O}} = \infty, f$ is convex on $\{x : \|x - x^*\|_2 \leq \delta_1\}$ and $\|x_{N_{\mathcal{P}}} - x^*\|_2 \leq \frac{\delta_1}{2}$. Set $k := N_{\mathcal{P}} + t$, $(t \in \mathbb{Z}^+)$, step size $\alpha_k = \mathcal{O}(\frac{1}{\sqrt{Nt}})$, and mini-batch size $|\mathcal{B}_k| = \mathcal{O}(t)$. Then for any $\tau \in (0, 1)$, we have $\{x_k\}$ converges to some stationary point in expectation with probability at least $1 - \tau$, *i.e.*,

$$\mathbb{P}(\liminf_{k\to\infty} \mathbb{E}\|\mathcal{G}_{\alpha_k}(x_k)\|_2^2 = 0) \geq 1 - \tau.$$

Outline

Convex experiments

Focus on the convex ℓ_1-regularized logistic regression with the form

$$\underset{(x;b)\in\mathbb{R}^{n+1}}{\text{minimize}} \ \frac{1}{N}\sum_{i=1}^{N}\log(1+e^{-l_i(x^T d_i+b)})+\lambda\|x\|_1,$$

for binary classification.

Dataset	N	n	Attribute	Dataset	N	n	Attribute
a9a	32561	123	binary {0, 1}	real-sim	72309	20958	real [0, 1]
higgs	11000000	28	real $[-3, 41]$	rcv1	20242	47236	real [0, 1]
kdda	8407752	20216830	real $[-1, 4]$	url_combined	2396130	3231961	real $[-4, 9]$
news20	19996	1355191	unit-length	w8a	49749	300	binary {0, 1}

Table: Summary of datasets

- Best value is marked as **bold**.
- OBProx-SG(+) performs competitively on objective function values.
- OBProx-SG(+) performs much better on sparsity exploration (lowest density).

Table: Objective function values F/f for tested algorithms on convex problems

Dataset	Prox-SG	RDA	Prox-SVRG	OBProx-SG	OBProx-SG+
a9a	0.332 / 0.330	0.330 / 0.329	0.330 / 0.329	**0.327 / 0.326**	0.329 / 0.328
higgs	**0.326 / 0.326**	**0.326 / 0.326**	**0.326 / 0.326**	**0.326 / 0.326**	**0.326 / 0.326**
kdda	**0.102 / 0.102**	0.103 / 0.103	0.105 / 0.105	**0.102 / 0.102**	**0.102 / 0.102**
news20	**0.413 / 0.355**	0.625 / 0.617	**0.413 / 0.355**	**0.413 / 0.355**	**0.413 / 0.355**
real-sim	**0.164 / 0.125**	0.428 / 0.421	**0.164 / 0.125**	**0.164 / 0.125**	**0.164 / 0.125**
rcv1	**0.242 / 0.179**	0.521 / 0.508	**0.242 / 0.179**	**0.242 / 0.179**	**0.242 / 0.179**
url_combined	0.050 / 0.049	0.634 / 0.634	0.078 / 0.077	0.050 / 0.049	**0.047 / 0.046**
w8a	**0.052 / 0.048**	0.080 / 0.079	**0.052 / 0.048**	**0.052 / 0.048**	**0.052 / 0.048**

Table: Density (%) of solutions for tested algorithms on convex problems

Dataset	Prox-SG	RDA	Prox-SVRG	OBProx-SG	OBProx-SG+
a9a	96.37	86.69	61.29	62.10	**59.68**
higgs	89.66	96.55	93.10	**70.69**	**70.69**
kdda	0.09	18.62	3.35	0.08	**0.06**
news20	4.24	0.44	0.20	0.20	**0.19**
real-sim	53.93	52.71	22.44	22.44	**22.15**
rcv1	16.95	9.61	4.36	4.36	**4.33**
url_combined	7.73	41.71	6.06	3.26	**3.00**
w8a	99.00	99.83	78.07	78.03	**74.75**

Runtime Comparison:

- Prox-SG, RDA and OBProx-SG(+) are almost as efficient as each other,
- Prox-SVRG takes much more time due to the computation of full gradient.

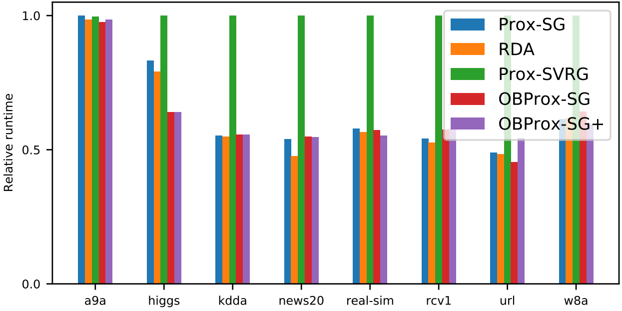


Figure: Relative runtime for tested algorithms on convex problems

Nonconvex experiments

- Model Architectures: ResNet18 and MobileNetV1.
- Datasets: CIFAR10 and Fashion-MNIST.

Table: Final objective values F/f for tested algorithms on non-convex problems

Backbone	Dataset	Prox-SG	RDA	Prox-SVRG	OBProx-SG	OBProx-SG+
MobileNetV1	CIFAR10	1.473 / 0.049	4.129 / 0.302	1.921 / 0.079	1.619 / **0.048**	**1.453** / 0.063
	Fashion-MNIST	1.314 / **0.089**	4.901 / 0.197	1.645 / 0.103	2.119 / **0.089**	**1.310** / 0.099
ResNet18	CIFAR10	0.781 / 0.034	1.494 / 0.051	0.815 / 0.031	**0.746 / 0.021**	0.755 / 0.044
	Fashion-MNIST	0.688 / 0.103	1.886 / 0.081	0.683 / **0.074**	**0.682 / 0.074**	0.689 / 0.116

Table: Density/testing accuracy (%) for tested algorithms on non-convex problems

Backbone	Dataset	Prox-SG	RDA	Prox-SVRG	OBProx-SG	OBProx-SG+
MobileNetV1	CIFAR10	14.17 / **90.98**	74.05 / 81.48	92.26 / 87.85	9.15 / 90.54	**2.90** / 90.91
	Fashion-MNIST	5.28 / 94.23	74.67 / 92.12	75.40 / 93.66	4.15 / 94.28	**1.23 / 94.39**
ResNet18	CIFAR10	11.60 / 92.43	41.01 / 90.74	37.92 / 92.48	2.12 / **92.81**	**0.88** / 92.45
	Fashion-MNIST	6.34 / 94.28	42.46 / 93.66	35.07 / 94.24	5.44 / **94.39**	**0.29** / 93.97

- OBProx-SG(+) performs competitively among the methods with respect to the final objective function values;
- OBProx-SG(+) computes much sparser solutions. Particularly, OBProx-SG+ achieves the highest sparse (lowest dense) solutions on all non-convex tests, of which the solutions are **4.24 to 21.86 times sparser** than those of Prox-SG.
- The density of OBProx-SG+ drops dramatically after switching to Orthant Step.

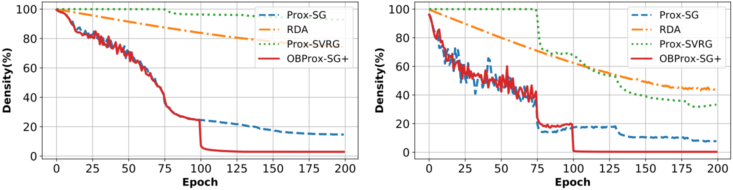


Figure: Density Evolution. (L): MobileNetV1 on CIFAR10. (R): ResNet18 on Fashion-MNIST

Outline

Conclusion

OBProx-SG is designed to

- explore the sparsity of solution effectively in stochastic settings.
- maintain the convergence property.
- sacrifice no generalization performance
- work in both convex and nonconvex settings

for effectively solving ℓ_1-regularized convex optimization problems.

Current OBProx-SG's optimizer:

- has been implemented as a Pytorch optimizer instance.
- is avaliable at `https://github.com/tianyic/obproxsg`.

Thank you!

Q & A