The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)

Faster Gradient-Free Proximal Stochastic Methods for Nonconvex Nonsmooth Optimization

Feihu Huang,^{1,2} **Bin Gu,**^{3} **Zhouyuan Huo,**^{2} **Songcan Chen,**^{1*} **Heng Huang**^{2,3}
^{1}College of Computer Science & Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China
^{2}Department of Electrical & Computer Engineering, University of Pittsburgh, PA 15261, USA ^{3}JDDGlobal.com
huangfeihu@nuaa.edu.cn, gubin3@jd.com, zhouyuan.huo@pitt.edu, s.chen@nuaa.edu.cn, heng.huang@pitt.edu

Abstract

Proximal gradient method has been playing an important role to solve many machine learning tasks, especially for the nonsmooth problems. However, in some machine learning problems such as the bandit model and the black-box learning problem, proximal gradient method could fail because the explicit gradients of these problems are difficult or infeasible to obtain. The gradient-free (zeroth-order) method can address these problems because only the objective function values are required in the optimization. Recently, the first zeroth-order proximal stochastic algorithm was proposed to solve the nonconvex nonsmooth problems. However, its convergence rate is $O(\frac{1}{\sqrt{T}})$ for the nonconvex problems, which is significantly slower than the best convergence rate $O(\frac{1}{T})$ of the zeroth-order stochastic algorithm, where T is the iteration number. To fill this gap, in the paper, we propose a class of faster zeroth-order proximal stochastic methods with the variance reduction techniques of SVRG and SAGA, which are denoted as ZO-ProxSVRG and ZO-ProxSAGA, respectively. In theoretical analysis, we address the main challenge that an unbiased estimate of the true gradient does not hold in the zeroth-order case, which was required in previous theoretical analysis of both SVRG and SAGA. Moreover, we prove that both ZO-ProxSVRG and ZO-ProxSAGA algorithms have $O(\frac{1}{T})$ convergence rates. Finally, the experimental results verify that our algorithms have a faster convergence rate than the existing zeroth-order proximal stochastic algorithm.

Introduction

Proximal gradient (PG) methods (Mine and Fukushima, 1981; Nesterov, 2004; Parikh, Boyd, and others, 2014) are a class of powerful optimization tools in artificial intelligence and machine learning. In general, it considers the following nonsmooth optimization problem:

$$\min_{x\in\mathbb{R}^d} f(x) + \psi(x), \tag{1}$$

where $f(x)$ usually is the loss function such as hinge loss and logistic loss, and $\psi(x)$ is the nonsmooth structure regularizer such as ℓ_1-norm regularization. In recent research, Beck and Teboulle (2009); Nesterov (2013) proposed the accelerate PG methods to solve convex problems by using the Nesterov's accelerated technique. After that, Li and Lin (2015) presented a class of accelerated PG methods for nonconvex optimization. More recently, Gu, Huo, and Huang (2018) introduced inexact PG methods for nonconvex nonsmooth optimization. To solve the big data problems, the incremental or stochastic PG methods (Bertsekas, 2011; Xiao and Zhang, 2014) were developed for large-scale convex optimization. Correspondingly, Ghadimi, Lan, and Zhang (2016); Reddi et al. (2016) proposed the stochastic PG methods for large-scale nonconvex optimization.

However, in many machine learning problems, the explicit expressions of gradients are difficult or infeasible to obtain. For example, in some complex graphical model inference (Wainwright, Jordan, and others, 2008) and structure prediction problems (Sokolov, Hitschler, and Riezler, 2018), it is difficult to compute the explicit gradients of the objective functions. Even worse, in bandit (Shamir, 2017) and black-box learning (Chen et al., 2017) problems, only the objective function values are available (the explicit gradients cannot be calculated). Clearly, the above PG methods will fail in dealing with these scenarios. The gradient-free (zeroth-order) optimization method (Nesterov and Spokoiny, 2017) is a promising choice to address these problems because it only uses the function values in optimization process. Thus, the gradient-free optimization methods have been increasingly embraced for solving many machine learning problems (Conn, Scheinberg, and Vicente, 2009).

Although many gradient-free methods have recently been developed and studied (Agarwal, Dekel, and Xiao, 2010; Nesterov and Spokoiny, 2017; Liu et al., 2018b), they often suffer from the high variances of zeroth-order gradient estimates. In addition, these algorithms are mainly designed for smooth or convex settings, which will be discussed in the below related works, thus limiting their applicability in a wide range of nonconvex nonsmooth machine learning problems such as involving the nonconvex loss functions and nonsmooth regularization.

In this paper, thus, we propose a class of faster gradient-free proximal stochastic methods for solving the nonconvex nonsmooth problem as follows:

$$\min_{x\in\mathbb{R}^d} F(x) =: f(x) + \psi(x),\ f(x) =: \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{2}$$

---

*Corresponding Author.

Table 1: Comparison of representative zeroth-order stochastic algorithms for finding an ϵ-approximate stationary point of nonconvex problem, i.e., $\mathbb{E}\|\nabla f(x)\|^2 \leq \epsilon$ or $\mathbb{E}\|g_\eta(x)\|^2 \leq \epsilon$. (S, NS, C and NC are the abbreviations of smooth, nonsmooth, convex and nonconvex, respectively. T is the whole iteration number, d is the dimension of data and n denotes the sample size.) $B(\leq n)$ is a mini-batch size.

Algorithm	**Reference**	**Gradient estimator**	**Problem**	**Convergence rate**
RSGF	Ghadimi and Lan (2013)	GauSGE	S(NC)	$O(\sqrt{\frac{d}{T}})$
ZO-SVRG	Liu et al. (2018c)	CooSGE	S(NC)	$O(\frac{d}{T})$
SZVR-G	Liu et al. (2018a)	GauSGE	S(NC)	$O(\max(d^{\frac{2}{3}}B^{\frac{1}{3}}, d^{\frac{1}{3}}B^{\frac{2}{3}})/T)$
		GauSGE	NS(NC)	$O(d^{\frac{5}{\sqrt{33}}}B^{\frac{1}{\sqrt{33}}}/T^{\sqrt{\frac{3}{11}}})$
RSPGF	Ghadimi, Lan, and Zhang (2016)	GauSGE	S(NC) + NS(C)	$O(\sqrt{\frac{d}{T}})$
ZO-ProxSVRG	Ours	CooSGE	S(NC) + NS(C)	$O(\frac{d}{T})$
		GauSGE	S(NC) + NS(C)	$O(\frac{d}{T} + d\sigma^2)$
ZO-ProxSAGA	Ours	CooSGE	S(NC) + NS(C)	$O(\frac{d}{T})$
		GauSGE	S(NC) + NS(C)	$O(\frac{d}{T} + d\sigma^2)$

where each $f_i(x)$ is a *nonconvex* and smooth loss function, and $\psi(x)$ is a convex and *nonsmooth* regularization term. Until now, there are few zeroth-order stochastic methods for solving the problem (2) except a recent attempt proposed in (Ghadimi, Lan, and Zhang, 2016). Specifically, Ghadimi, Lan, and Zhang (2016) have proposed a randomized stochastic projected gradient-free method (RSPGF), *i.e.*, a zeroth-order proximal stochastic gradient method. However, due to the large variance of zeroth-order estimated gradient generated from randomly selecting the sample and the direction of derivative, the RSPGE only has a convergence rate $O(\frac{1}{\sqrt{T}})$, which is significantly slower than $O(\frac{1}{T})$, the best convergence rate of the zeroth-order stochastic algorithm. To accelerate the RSPGF algorithm, we use the variance reduction strategies in the first-order methods, *i.e.*, SVRG (Xiao and Zhang, 2014) and SAGA (Defazio, Bach, and Lacoste-Julien, 2014), to reduce the variance of estimated stochastic gradient.

Although SVRG and SAGA have shown good performances, applying these strategies to the zeroth-order method is **not a trivial task**. The main challenge arises due to that both SVRG and SAGA rely on the assumption that a stochastic gradient is an **unbiased** estimate of the true full gradient. However, it does not hold in the zeroth-order algorithms. In the paper, thus, we will fill this gap between zeroth-order proximal stochastic method and the classic variance reduction approaches (SVRG and SAGA).

Main Contributions

In summary, our main contributions are summarized as follows:

- We propose a class of faster gradient-free proximal stochastic methods (ZO-ProxSVRG and ZO-ProxSAGA), based on the variance reduction techniques of SVRG and SAGA. Our new algorithms only use the objective function values in the optimization process.
- Moreover, we provide the theoretical analysis on the convergence properties of both new ZO-ProxSVRG and ZO-ProxSAGA methods. Table 1 shows the specifical convergence rates of the proposed algorithms and other related ones. In particular, our algorithms have faster convergence rate $O(\frac{1}{T})$ than $O(\frac{1}{\sqrt{T}})$ of the RSPGF (Ghadimi, Lan, and Zhang, 2016) (the existing stochastic PG algorithm for solving nonconvex nonsmoothing problems).
- Extensive experimental results and theoretical analysis demonstrate the effectiveness of our algorithms.

Related Works

Gradient-free (zeroth-order) methods have been effectively used to solve many machine learning problems, where the explicit gradient is difficult or infeasible to obtain, and have also been widely studied. For example, Nesterov and Spokoiny (2017) proposed several random gradient-free methods by using Gaussian smoothing technique. Duchi et al. (2015) proposed a zeroth-order mirror descent algorithm. More recently, Yu et al. (2018); Dvurechensky, Gasnikov, and Gorbunov (2018) presented the accelerated zeroth-order methods for the convex optimization. To solve the nonsmooth problems, the zeroth-order online or stochastic ADMM methods (Liu et al., 2018b; Gao, Jiang, and Zhang, 2018) have been introduced.

The above zeroth-order methods mainly focus on the (strongly) convex problems. In fact, there exist many nonconvex machine learning tasks, whose explicit gradients are not available, such as the nonconvex black-box learning problems (Chen et al., 2017; Liu et al., 2018c). Thus, several recent works have begun to study the zeroth-order stochastic methods for the nonconvex optimization. For example, Ghadimi and Lan (2013) proposed the randomized stochastic gradient-free (RSGF) method, *i.e.*, a zeroth-order stochastic gradient method. To accelerate optimization, more recently, Liu et al. (2018c,a) proposed the zeroth-order stochastic variance reduction gradient (ZO-SVRG) methods. Moreover, to solve the large-scale machine learning problems, some asynchronous parallel stochastic zeroth-order algorithms have been proposed in (Gu, Huo, and Huang, 2016; Lian et al., 2016; Gu et al., 2018).

Although the above zeroth-order stochastic methods can effectively solve the nonconvex optimization, there are few zeroth-order stochastic methods for the *nonconvex nonsmooth* composite optimization except the RSPGF method presented in (Ghadimi, Lan, and Zhang, 2016). In addition, Liu et al. (2018a) have also studied the zeroth-order algorithm for solving the nonconvex nonsmooth problem, which is different from problem (2).

Zeroth-Order Proximal Stochastic Method Revisit

In this section, we briefly review the zeroth-order proximal stochastic gradient (ZO-ProxSGD) method to solve the problem (2). Before that, we first revisit the proximal gradient descent (ProxGD) method (Mine and Fukushima, 1981).

ProxGD is an effective method to solve the problem (2) via the following iteration:

$$x_{t+1} = \mathrm{Prox}_{\eta\psi}\big(x_t - \eta\nabla f(x_t)\big),\ t = 0, 1, \cdots, \tag{3}$$

where $\eta > 0$ is a step size, and $\mathrm{Prox}_{\eta\psi}(\cdot)$ is a *proximal operator* defined as:

$$\mathrm{Prox}_{\eta\psi}(x) = \arg\min_{y\in\mathbb{R}^d} \big\{\psi(y) + \frac{1}{2\eta}\|y - x\|^2\big\}. \tag{4}$$

As discussed above, because ProxGD needs to compute the gradient at each iteration, it cannot be applied to solve the problems, where the explicit gradient of function $f(x)$ is not available. For example, in the black-box machine learning model, only function values (*e.g.*, prediction results) are available Chen et al. (2017). To avoid computing explicit gradient, we use the zeroth-order gradient estimators (Nesterov and Spokoiny, 2017; Liu et al., 2018c) to estimate the gradient only by function values.

- Specifically, we use the **Gau**ssian **S**moothing **G**radient **E**stimator (**GauSGE**) (Nesterov and Spokoiny, 2017; Ghadimi, Lan, and Zhang, 2016) to estimate the gradients as follows:

  $$\hat{\nabla} f_i(x) = \frac{f_i(x + \mu u_i) - f_i(x)}{\mu} u_i,\ \ i \in [n], \tag{5}$$

  where μ is a smoothing parameter, and $\{u_i\}_{i=1}^n$ denote *i.i.d.* random directions drawn from a zero-mean isotropic multivariate Gaussian distribution $\mathcal{N}(0, I)$.
- Moreover, to obtain better estimated gradient, we can use the **Coo**rdinate **S**moothing **G**radient **E**stimator (**CooSGE**) (Gu, Huo, and Huang, 2016; Gu et al., 2018; Liu et al., 2018c) to estimate the gradients as follows:

  $$\hat{\nabla} f_i(x) = \sum_{j=1}^d \frac{f_i(x + \mu_j e_j) - f_i(x - \mu_j e_j)}{2\mu_j} e_j,\ \ i \in [n], \tag{6}$$

  where μ_j is a coordinate-wise smoothing parameter, and e_j is a standard basis vector with 1 at its j-th coordinate, and 0 otherwise. Although the CooSGE need more function queries than the GauSGE, it can get better estimated gradient, and even can make the algorithms to obtain a faster convergence rate.

Finally, based on these estimated gradients, we give a zeroth-order proximal gradient descent (ZO-ProxGD) method, which performs the following iteration:

$$x_{t+1} = \mathrm{Prox}_{\eta\psi}\big(x_t - \eta\hat{\nabla} f(x_t)\big),\ t = 0, 1, \cdots, \tag{7}$$

where $\hat{\nabla} f(x) = \frac{1}{n}\sum_{i=1}^n \hat{\nabla} f_i(x)$.

Since ZO-ProxGD needs to estimate full gradient $\hat{\nabla} f(x) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(x)$, when n is large in the problem (2), its high cost per iteration is prohibitive. As a result, Ghadimi, Lan, and Zhang (2016) proposed the RSPGF (*i.e.*, ZO-ProxSGD) with performing the following iteration:

$$x_{t+1} = \mathrm{Prox}_{\eta\psi}\big(x_t - \eta\hat{\nabla} f_{\mathcal{I}_t}(x_t)\big),\ t = 0, 1, \cdots, \tag{8}$$

where $\hat{\nabla} f_{\mathcal{I}_t}(x_t) = \frac{1}{b}\sum_{i\in\mathcal{I}_t} \hat{\nabla} f_i(x)$, $\mathcal{I}_t \in \{1, 2, \cdots, n\}$ and $b = |\mathcal{I}_t|$ is the mini-batch size.

New Faster Zeroth-Order Proximal Stochastic Methods

In this section, to efficiently solve the large-scale nonconvex nonsmooth problems, we propose a class of faster zeroth-order proximal stochastic methods with the variance reduction (VR) techniques of SVRG and SAGA, respectively.

ZO-ProxSVRG

In the subsection, we propose the zeroth-order proximal SVRG (ZO-ProxSVRG) method by using VR technique of SVRG in (Xiao and Zhang, 2014; Reddi et al., 2016).

The corresponding algorithmic framework is described in Algorithm 1, where we use a mixture stochastic gradient $\hat{v}_t^s = \hat{\nabla} f_{\mathcal{I}_t}(x_t^s) - \hat{\nabla} f_{\mathcal{I}_t}(\tilde{x}^s) + \hat{\nabla} f(\tilde{x}^s)$. Note that $\mathbb{E}_{\mathcal{I}_t}[\hat{v}_t^s] = \hat{\nabla} f(x_t^s) \neq \nabla f(x_t^s)$, *i.e.*, this stochastic gradient is a **biased** estimate of the true full gradient. Although the SVRG has shown a great promise, it relies upon the assumption that the stochastic gradient is an **unbiased** estimate of the true full gradient. Thus, adapting the similar ideas of SVRG to zeroth-order optimization is not a trivial task. To address this issue, we analyze the upper bound for the variance of the estimated gradient $\hat{v}_t^s$, and choose the appropriate step size η and smoothing parameter μ to control this variance, which will be in detail discussed in the below theorems.

Next, we derive the upper bounds for the variance of estimated gradient $\hat{v}_t^s$ based on the CooSGE and the GauSGE, respectively.

Lemma 1. *In Algorithm 1 using the CooSGE, given the mixture estimated gradient* $\hat{v}_t^s = \hat{\nabla} f_{\mathcal{I}_t}(x_t^s) - \hat{\nabla} f_{\mathcal{I}_t}(\tilde{x}^s) + \hat{\nabla} f(\tilde{x}^s)$*, then the following inequality holds*

$$\mathbb{E}\|\hat{v}_t^s - \nabla f(x_t^s)\|^2 \leq \frac{2\delta_n L^2 d}{b}\mathbb{E}\|x_t^s - \tilde{x}^s\|^2 + \frac{L^2 d^2 \mu^2}{2}, \tag{9}$$

where $0 \leq \delta_n \leq 1$.

Remark 1. *Lemma 1 shows that variance of* $\hat{v}_t^s$ *has an upper bound. As the number of iterations increases, both* x_t^s *and* $\tilde{x}^s$ *will approach the same stationary point* x^**, then the variance of stochastic gradient decreases, but does not vanishes, due to using the zeroth-order estimated gradient.*

Lemma 2. *In Algorithm 1 using the GauSGE, given the estimated gradient* $\hat{v}_t^s = \hat{\nabla} f_{\mathcal{I}_t}(x_t^s) - \hat{\nabla} f_{\mathcal{I}_t}(\tilde{x}^s) + \hat{\nabla} f(\tilde{x}^s)$*, then the following inequality holds*

$$\mathbb{E}\|\hat{v}_t^s - \nabla f(x_t)\|^2 \leq (2 + \frac{12\delta_n}{b})(d+6)^3 L^2 \mu^2 + \frac{6\delta_n L^2}{b}\mathbb{E}\|x_t^s - \tilde{x}^s\|^2 + (4 + \frac{24\delta_n}{b})(2d+9)\sigma^2. \quad (10)$$

Remark 2. *Lemma 2 shows that variance of* $\hat{v}_t^s$ *has an upper bound. As the number of iterations increases, both* x_t^s *and* $\tilde{x}^s$ *will approach the same stationary point* x^**, then the variance of stochastic gradient decreases.*

Algorithm 1 ZO-ProxSVRG for Nonconvex Optimization

```
Input: mini-batch size $b$, $S$, $m$ and step size $\eta > 0$;
Initialize: $x_0^1 = \tilde{x}^1 \in \mathbb{R}^d$;
for $s = 1, 2, \cdots, S$ do
    $\hat{\nabla} f(\tilde{x}^s) = \frac{1}{n}\sum_{i=1}^n \hat{\nabla} f_i(\tilde{x}^s)$;
    for $t = 0, 1, \cdots, m-1$ do
        Uniformly randomly pick a mini-batch $\mathcal{I}_t \subseteq \{1, 2, \cdots, n\}$ such that $|\mathcal{I}_t| = b$;
        Using (5) or (6) to estimate mixture stochastic gradient $\hat{v}_t^s = \hat{\nabla} f_{\mathcal{I}_t}(x_t^s) - \hat{\nabla} f_{\mathcal{I}_t}(\tilde{x}^s) + \hat{\nabla} f(\tilde{x}^s)$;
        $x_{t+1}^s = \text{Prox}_{\eta\psi}(x_t^s - \eta\hat{v}_t^s)$;
    end for
    $\tilde{x}^{s+1} = x_m^s$ and $x_0^{s+1} = x_m^s$;
end for
Output: Iterate $x$ chosen uniformly random from $\{(x_t^s)_{t=1}^m\}_{s=1}^S$.
```

ZO-ProxSAGA

In the subsection, we propose the zeroth-order proximal SAGA (ZO-ProxSAGA) method via using VR technique of SAGA in (Defazio, Bach, and Lacoste-Julien, 2014; Reddi et al., 2016).

The corresponding algorithmic description is given in Algorithm 2, where we use a mixture stochatic gradient $\hat{v}_t = \frac{1}{b}\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(x_t) - \hat{\nabla} f_{i_t}(z_{i_t}^t)\big) + \hat{\phi}_t$. Similarly, $\mathbb{E}_{\mathcal{I}_t}[\hat{v}_t] = \hat{\nabla} f(x_t^s) \neq \nabla f(x_t^s)$, *i.e.*, this stochastic gradient is a **biased** estimate of the true full gradient. Note that in Algorithm 2, due to $\sum_{i_t \in \mathcal{I}_t} \hat{\nabla} f_{i_t}(z_{i_t}^{t+1}) = \sum_{i_t \in \mathcal{I}_t} \hat{\nabla} f_{i_t}(x_t)$, the step 8 can use directly the term $\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(x_t) - \hat{\nabla} f_{i_t}(z_{i_t}^t)\big)$, which is computed in the step 5, to avoid unnecessary calculations. Next, we give the upper bounds for the variance of stochastic gradient $\hat{v}_t$ based on the CooSGE and the GauSGE, respectively.

Lemma 3. *In Algorithm 2 using the CooSGE, given the estimated gradient* $\hat{v}_t = \frac{1}{b}\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(x_t) - \hat{\nabla} f_{i_t}(z_{i_t}^t)\big) + \hat{\phi}_t$ *with* $\hat{\phi}_t = \frac{1}{n}\sum_{i=1}^n \hat{\nabla} f_i(z_i^t)$*, then the following inequality holds*

$$\mathbb{E}\|\hat{v}_t - \nabla f(x_t)\|^2 \leq \frac{2L^2 d}{nb}\sum_{i=1}^n \mathbb{E}\|x_t - z_i^t\|_2^2 + \frac{L^2 d^2 \mu^2}{2}. \quad (11)$$

Remark 3. *Lemma 3 shows that variance of* $\hat{v}_t$ *has an upper bound. As the number of iterations increases, both* x_t *and* $\{z_i^t\}_{i=1}^n$ *will approach the same stationary point, then the variance of stochastic gradient decreases.*

Lemma 4. *In Algorithm 2 using GauSGE, given the estimated gradient* $\hat{v}_t = \frac{1}{b}\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(x_t) - \hat{\nabla} f_{i_t}(z_{i_t}^t)\big) + \hat{\phi}_t$ *with* $\hat{\phi}_t = \frac{1}{n}\sum_{i=1}^n \hat{\nabla} f_i(z_i^t)$*, then the following inequality holds*

$$\mathbb{E}\|\hat{v}_t - \nabla f(x_t)\|^2 \leq (2 + \frac{12}{b})(d+6)^3 L^2 \mu^2 + \frac{6L^2}{nb}\sum_{i=1}^n \mathbb{E}\|x_t - z_i^t\|^2 + (4 + \frac{24}{b})(2d+9)\sigma^2. \quad (12)$$

Remark 4. *Lemma 4 shows that variance of* $\hat{v}_t$ *has an upper bound. As the number of iterations increases, both* x_t *and* $\{z_i^t\}_{i=1}^n$ *will approach the same stationary point* x^**, then the variance of stochastic gradient decreases.*

Algorithm 2 ZO-ProxSAGA for Nonconvex Optimization

```
1:  Input: mini-batch size $b$, $T$ and step size $\eta > 0$;
2:  Initialize: $x_0 \in \mathbb{R}^d$, and $z_i^0 = x_0$ for $i \in \{1, 2, \cdots, n\}$, $\hat{\phi}_0 = \frac{1}{n}\sum_{i=1}^n \hat{\nabla} f_i(z_i^0)$;
3:  for $t = 0, 1, \cdots, T-1$ do
4:      Uniformly randomly pick a mini-batch $\mathcal{I}_t \subseteq \{1, 2, \cdots, n\}$ (with replacement) such that $|\mathcal{I}_t| = b$;
5:      Using (5) or (6) to estimate mixture stochastic gradient $\hat{v}_t = \frac{1}{b}\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(x_t) - \hat{\nabla} f_{i_t}(z_{i_t}^t)\big) + \hat{\phi}_t$;
6:      $x_{t+1} = \text{Prox}_{\eta\psi}(x_t - \eta\hat{v}_t)$;
7:      $z_{i_t}^{t+1} = x_t$ for $i \in \mathcal{I}_t$ and $z_i^{t+1} = z_i^t$ for $i \notin \mathcal{I}_t$;
8:      $\hat{\phi}_{t+1} = \hat{\phi}_t - \frac{1}{n}\sum_{i_t \in \mathcal{I}_t}\big(\hat{\nabla} f_{i_t}(z_{i_t}^t) - \hat{\nabla} f_{i_t}(z_{i_t}^{t+1})\big)$;
9:  end for
10: Output: Iterate $x$ chosen uniformly random from $\{x_t\}_{t=1}^T$.
```

Convergence Analysis

In this section, we conduct the convergence analysis of both ZO-ProxSVRG and ZO-ProxSAGA. First, we give some mild assumptions regarding problem (2) as follows:

Assumption 1. *For* $\forall i \in \{1, 2, \cdots, n\}$*, gradient of the function* f_i *is Lipschitz continuous with a Lipschitz constant* $L > 0$*, such that*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|, \; \forall x, y \in \mathbb{R}^d,$$

which implies

$$f_i(x) \leq f_i(y) + \nabla f_i(y)^T(x - y) + \frac{L}{2}\|x - y\|^2.$$

Assumption 2. *The gradient is bounded as* $\|\nabla f_i(x)\|^2 \leq \sigma^2$ *for all* $i = 1, 2, \cdots, n$.

The first assumption is standard for the convergence analysis of the zeroth-order algorithms (Ghadimi, Lan, and Zhang, 2016; Nesterov and Spokoiny, 2017; Liu et al., 2018c). The second assumption gives the bounded gradient used in (Nesterov and Spokoiny, 2017; Liu et al., 2018b), which is relatively stricter than the bounded variance of gradient in (Lian et al., 2016; Liu et al., 2018c,a), due to that we need to analyze more complex problem (2) including a non-smooth part. Next, we introduce the standard *gradient mapping* (Parikh, Boyd, and others, 2014) used in the convergence analysis as follows:

$$g_\eta(x) = \frac{1}{\eta}\big(x - \mathrm{Prox}_{\eta\psi}(x - \eta\nabla f(x))\big). \tag{13}$$

For the nonconvex problems, if $g_\eta(x) = 0$, the point x is a critical point (Parikh, Boyd, and others, 2014). Thus, we can use the following definition as the convergence metric.

Definition 1. *(Reddi et al., 2016) A solution x is called ϵ-accurate, if $\mathbb{E}\|g_\eta(x)\|^2 \leq \epsilon$ for some $\eta > 0$.*

Convergence Analysis of ZO-ProxSVRG

In the subsection, we show the convergence analysis of the ZO-ProxSVRG with the CooSGE (**ZO-ProxSVRG-CooSGE**) and the GauSGE (**ZO-ProxSVRG-GauSGE**), respectively.

Theorem 1. *Assume the sequence $\{(x_t^s)_{t=1}^m\}_{s=1}^S$ generated from Algorithm 1 using the **CooSGE**, and define a sequence $\{c_t\}_{t=1}^m$ as follows: for $s = 1, 2, \cdots, S$*

$$c_t = \begin{cases} \dfrac{\delta_n L^2 d\eta}{b} + c_{t+1}(1+\beta), \ 0 \leq t \leq m-1; \\ 0, \ t = m \end{cases} \tag{14}$$

where $\beta > 0$. Let $T = mS$, $\eta = \frac{\rho}{dL}$ $(0 < \rho < \frac{1}{2})$ and b satisfies the following inequality:

$$\frac{8\rho^2 m^2}{b} + \rho \leq 1, \tag{15}$$

then we have

$$\mathbb{E}\|g_\eta(x_t^s)\|^2 \leq \frac{\mathbb{E}[F(x_0^1) - F(x_*)]}{T\gamma} + \frac{L^2 d^2 \mu^2 \eta}{4\gamma}, \tag{16}$$

where $\gamma = \frac{\eta}{2} - L\eta^2$ and x^* is an optimal solution of the problem (2). *Further let $b = [n^{\frac{2}{3}}]$, $m = [n^{\frac{1}{3}}]$, $\rho = \frac{1}{4}$ and $\mu = O(\frac{1}{\sqrt{dT}})$, we have*

$$\mathbb{E}\|g_\eta(x_t^s)\|^2 \leq \frac{16dL\mathbb{E}[F(x_0^1) - F(x_*)]}{T} + O(\frac{d}{T}). \tag{17}$$

Remark 5. *Theorem 1 shows that, given $\mu = O(\frac{1}{\sqrt{dT}})$, $b = [n^{\frac{2}{3}}]$ and $m = [n^{\frac{1}{3}}]$, the ZO-ProxSVRG-CooSGE has $O(\frac{d}{T})$ convergence rate.*

Theorem 2. *Assume the sequence $\{(x_t^s)_{t=1}^m\}_{s=1}^S$ generated from Algorithm 1 using the GauSGE, and define a sequence $\{c_t\}_{t=1}^m$ as follows: for $s = 1, 2, \cdots, S$*

$$c_t = \begin{cases} \dfrac{3\delta_n L^2 \eta}{b} + c_{t+1}(1+\beta), \ 0 \leq t \leq m-1; \\ 0, \ t = m \end{cases} \tag{18}$$

where $\beta > 0$. Let $\eta = \frac{\rho}{L}$ $(0 < \rho < \frac{1}{2})$ and b satisfies the following inequality:

$$\frac{24\rho^2 m^2}{b} + \rho \leq 1, \tag{19}$$

then we have

$$\begin{aligned}\mathbb{E}\|g_\eta(x_t^s)\|^2 \leq & \frac{\mathbb{E}[F(x_0^1) - F(x_*)]}{T\gamma} + (1 + \frac{6\delta_n}{b})(d+6)^3\frac{L^2\mu^2\eta}{\gamma} \\ & + (2 + \frac{12\delta_n}{b})(2d+9)\frac{\sigma^2\eta}{\gamma},\end{aligned} \tag{20}$$

where $\gamma = \frac{\eta}{2} - \eta^2 L$ and x^* is an optimal solution of the problem (2). *Further let $b = [n^{\frac{2}{3}}]$, $m = [n^{\frac{1}{3}}]$, $\rho = \frac{1}{6}$ and $\mu = O(\frac{1}{d\sqrt{T}})$, we have*

$$\begin{aligned}\mathbb{E}\|g_\eta(x_t^s)\|^2 \leq & \frac{18L\mathbb{E}[F(x_0^1) - F(x_*)]}{T} + O(\frac{d}{T}) \\ & + O(d\sigma^2).\end{aligned} \tag{21}$$

Remark 6. *Theorem 2 shows that given $\mu = O(\frac{1}{d\sqrt{T}})$, $b = [n^{\frac{2}{3}}]$ and $m = [n^{\frac{1}{3}}]$, the ZO-ProxSVRG-GauSGE has $O(\frac{d}{T} + d\sigma^2)$ convergence rate, in which the part $O(d\sigma^2)$ generates from the GauSGE.*

Convergence Analysis of ZO-ProxSAGA

In this subsection, we provide the convergence analysis of the ZO-ProxSAGA with the CooSGE (**ZO-ProxSAGA-CooSGE**) and the GauSGE (**ZO-ProxSAGA-GauSGE**), respectively.

Theorem 3. *Assume the sequence $\{x_t\}_{t=1}^T$ generated from Algorithm 2 using the CooSGE, and define a positive sequence $\{c_t\}_{t=1}^T$ as follows:*

$$c_t = \frac{L^2 d\eta}{b} + c_{t+1}(1-p)(1+\beta) \tag{22}$$

where $\beta > 0$. Let $c_T = 0$, $\eta = \frac{\rho}{Ld}$ $(0 < \rho < \frac{1}{2})$, and b satisfies the following inequality:

$$\frac{32\rho^2 n^2}{b^3} + \rho \leq 1, \tag{23}$$

then we have

$$\mathbb{E}\|g_\eta(x_t)\|^2 \leq \frac{\mathbb{E}[F(x_0) - F(x_*)]}{T\gamma} + \frac{L^2 d^2 \mu^2 \eta}{4\gamma}, \tag{24}$$

where $\gamma = \frac{\eta}{2} - L\eta^2$ and x^* is an optimal solution of the problem (2). *Further let $b = [n^{\frac{2}{3}}]$, $\rho = \frac{1}{8}$ and $\mu = O(\frac{1}{\sqrt{dT}})$, we have*

$$\mathbb{E}\|g_\eta(x_t)\|^2 \leq \frac{64dL\mathbb{E}[F(x_0) - F(x_*)]}{3T} + O(\frac{d}{T}). \tag{25}$$

Remark 7. *Theorem 3 shows that given $\mu = O(\frac{1}{\sqrt{dT}})$ and $b = [n^{\frac{2}{3}}]$, the ZO-ProxSAGA-CooSGE has $O(\frac{d}{T})$ convergence rate.*

Theorem 4. *Assume the sequence* $\{x_t\}_{t=1}^T$ *generated from Algorithm 2 using the GauSGE, and define a positive sequence* $\{c_t\}_{t=1}^T$ *as follows:*

$$c_t = \frac{3L^2\eta}{b} + c_{t+1}(1-p)(1+\beta), \tag{26}$$

where $\beta > 0$*. Let* $c_T = 0$*,* $\eta = \frac{\rho}{L}(0 < \rho < \frac{1}{2})$ *and b satisfies the following inequality:*

$$\frac{96\rho^2 n^2}{b^3} + \rho \leq 1, \tag{27}$$

then we have

$$\mathbb{E}\|g_\eta(x_t)\|^2 \leq \frac{\mathbb{E}[F(x_0) - F(x_*)]}{T\gamma} + \frac{(2+\frac{12}{b})(2d+9)\sigma^2\eta}{\gamma} + \frac{(1+\frac{6}{b})(d+6)^3L^2\mu^2\eta}{\gamma}, \tag{28}$$

where $\gamma = \frac{1}{2\eta} - L\eta^2$ *and* x^* *is an optimal solution of the problem* (2)*. Further let* $b = [n^{\frac{2}{3}}]$*,* $\rho = \frac{1}{12}$ *and* $\mu = O(\frac{1}{d\sqrt{T}})$*, we have*

$$\mathbb{E}\|g_\eta(x_t)\|^2 \leq \frac{144L\mathbb{E}[F(x_0) - F(x_*)]}{5T} + O(\frac{d}{T}) + O(d\sigma^2). \tag{29}$$

Remark 8. *Theorem 4 shows that given* $\mu = O(\frac{1}{d\sqrt{T}})$ *and* $b = [n^{\frac{2}{3}}]$*, the ZO-ProxSAGA-GauSGE has* $O(\frac{d}{T} + d\sigma^2)$ *convergence rate, in which the part* $O(d\sigma^2)$ *generates from the GauSGE.*

All related proofs are in the supplementary document.

Experiments

In this section, we will compare the proposed algorithms (ZO-ProxSVRG-CooSGE, ZO-ProxSVRG-GauSGE, ZO-ProxSAGA-CooSGE, ZO-ProxSAGA-GauSGE) with the RSPGF method (Ghadimi, Lan, and Zhang, 2016) on two applications: **black-box binary classification** and **adversarial attacks on black-box deep neural networks (DNNs)**. Note that the RSPGF uses the GauSGE to estimate gradient.

Black-Box Binary Classification

Experimental Setup In this experiment, we apply our algorithms to learn the black-box binary classification problem. Specifically, given a set of training samples $\{a_i, l_i\}_{i=1}^n$, where $a_i \in \mathbb{R}^d$ and $l_i \in \{-1, 1\}$, we find the optimal predictor $x \in \mathbb{R}^d$ by solving the following problem:

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n f_i(x) + \lambda_1\|x\|_1 + \lambda_2\|x\|_2^2, \tag{30}$$

where $f_i(x)$ is the black-box loss function, that only returns the function value given an input. Here, we specify the non-convex *sigmoid loss* function $f_i(x) = \frac{1}{1+\exp(l_i a_i^T x)}$ in the black-box setting.

Table 2: Real data for black-box binary classification

datasets	$\#samples$	$\#features$	$\#classes$
20news	16,242	100	2
a9a	32,561	123	2
w8a	64,700	300	2
covtype.binary	581,012	54	2

In the experiment, we use the publicly available real datasets[1], which are summarized in Table 2. In the algorithms, we fix the mini-batch size $b = 20$, the smoothing parameters $\mu = \frac{1}{d\sqrt{t}}$ in the GauSGE and $\mu = \frac{1}{\sqrt{dt}}$ in the GooSGE. Meanwhile, we fix $\lambda_1 = \lambda_2 = 10^{-5}$, and use the same initial solution x_0 from the standard normal distribution in each experiment. For each dataset, we use half of the samples as training data, and the rest as testing data.

Experimental Results Figures 1 and 2 show that both objective values and test losses of the proposed methods faster decrease than the RSPGF method, as the time increases. In particular, both the ZO-ProxSVRG and ZO-ProxSAGA using the CooSGE show the better performances than the counterparts using the GauSGE. From these results, we find that the CooSGE shows the better performances than the CauSGE in estimating gradients. Moreover, these results also demonstrate that both the ZO-ProxSVRG and ZO-ProxSAGA using the CooSGE have a relatively faster convergence rate than the counterparts using the GauSGE. Since the ZO-ProxSAGA has less function query complexity than the ZO-ProxSVRG, it shows the better performances than the ZO-ProxSVRG. For example, the ZO-ProxSVRG-CooSGE needs $O(ndS + bdT)$ function queries, while ZO-SAGA-CooSGE needs $O(bdT)$ function queries.

Adversarial Attacks on Black-Box DNNs

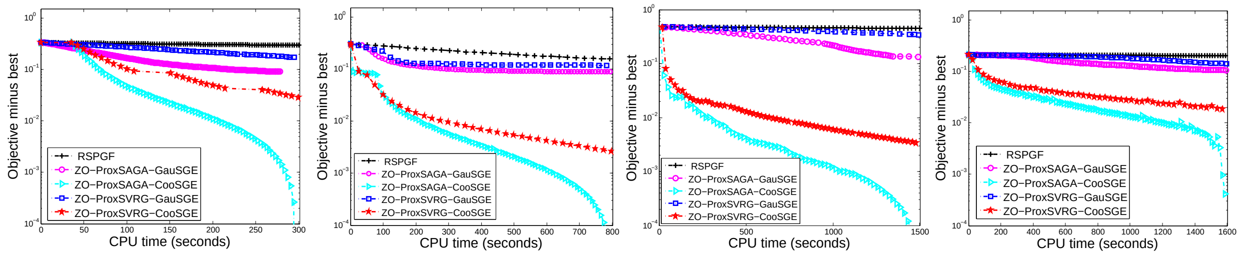
In this experiment, we apply our methods to generate adversarial examples to attack a pre-trained neural network model. Following (Chen et al., 2017; Liu et al., 2018c), the parameters of given model are hidden from us and only its outputs are accessible. In this case, we can not compute the gradients by using back-propagation algorithm. Thus, we use the zeroth-order algorithms to find an universal adversarial perturbation $x \in \mathbb{R}^d$ that could fool the samples $\{a_i \in \mathbb{R}^d,\ l_i \in \mathbb{N}\}_{i=1}^n$, which can be specified as the following elastic-net attacks to black-box DNNs problem:

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n \max\left\{F_{l_i}(a_i + x) - \max_{j\neq l_i} F_j(a_i + x), 0\right\} + \lambda_1\|x\|_1 + \lambda_2\|x\|_2^2, \tag{31}$$

where λ_1 and λ_2 are nonnegative parameters to balance attack success rate, distortion and sparsity. Here $F(a) = [F_1(a), \cdots, F_K(a)] \in [0,1]^K$ represents the final layer output of neural network, which is the probabilities of K classes.

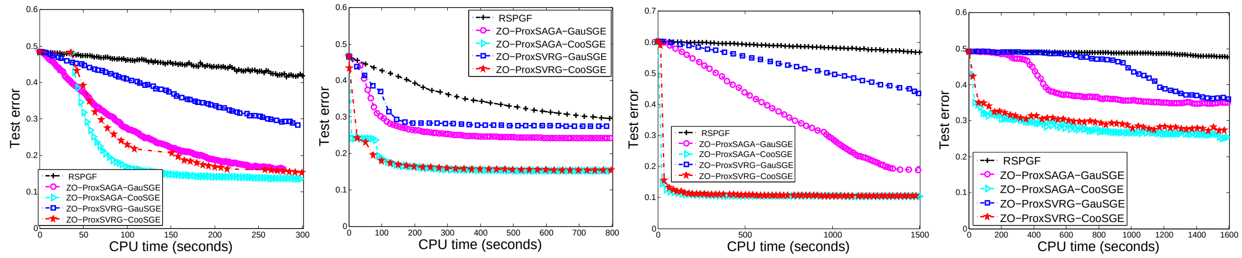
---

[1] *20news* is from the website https://cs.nyu.edu/~roweis/data.html; *a9a*, *w8a* and *covtype.binary* are from the website www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.



(a) 20news (b) a9a (c) w8a (d) covtype.binary

Figure 1: Objective value *versus* CPU time on black-box binary classification.

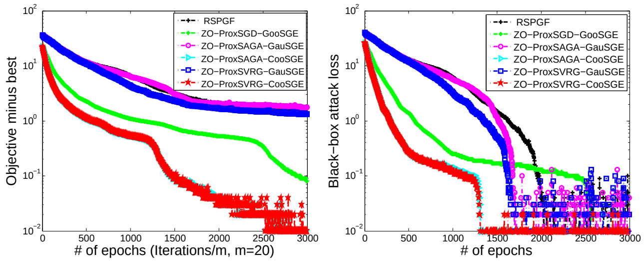


(a) 20news (b) a9a (c) w8a (d) covtype.binary

Figure 2: Test loss *versus* CPU time on black-box binary classification.

Following (Liu et al., 2018c), we use a pre-trained DNN[2] on the MNIST dataset as the target black-box model, which achieves 99.4% test accuracy. In the experiment, we select $n = 10$ examples from the same class, and set the batch size $b = 5$ and a constant step size $\eta = 1/d$ for the zeroth-order algorithms, where $d = 28 \times 28$. In addition, we set $\lambda_1 = 10^{-3}$ and $\lambda_2 = 1$ in the experiment.

Figure 3 shows that both objective values and black-box attack losses (*i.e.* the first part of the problem (31)) of the proposed algorithms faster decrease than the RSPGF method, as the number of iteration increases. Here, we add the ZO-ProxSGD-CooSGE method for comparison, which is obtained by combining the ZO-ProxSGD method with the CooSGE. Interestingly, the ZO-ProxSGD-CooSGE shows better performance than both the ZO-ProxSVRG-GauSGE and ZO-ProxSAGA-GauSGE, which further demonstrates that the CooSGE can have better performance than the CauSGE in estimating gradient. Although having a relatively good performance in generating the adversarial samples, the ZO-ProxSGD still shows worse performance than both the ZO-ProxSVRG-CooSGE and ZO-ProxSAGA-CooSGE, due to not using the VR technique.



(a) *Objective function value* (b) *Black-box attack loss*

Figure 3: Objective value and attack loss on generating adversarial samples from black-box DNNs.

Conclusions

In this paper, we proposed a class of faster gradient-free proximal stochastic methods based on the zeroth-order gradient estimators, *i.e.*, the GauSGE and the CooSGE, which only use the objective function values in the optimization. Moreover, we provided the theoretical analysis on the convergence properties of the proposed algorithms (ZO-ProxSVRG and ZO-ProxSAGA) based on the CooSGE and the GauSGE, respectively. In particular, both the ZO-ProxSVRG and ZO-ProxSAGA using the CooSGE have relatively faster convergence rates than the counterparts using the GauSGE, since the CooSGE has better performance than the CauSGE in estimating gradients.

Acknowledgments

F. Huang and S. Chen were partially supported by the Natural Science Foundation of China (NSFC) under Grant No. 61806093 and No. 61682281, and the Key Program of NSFC under Grant No. 61732006, and Jiangsu Postdoctoral Research Grant Program No. 2018K004A. F. Huang, Z. Huo, H. Huang were partially supported by U.S. NSF

---

[2] https://github.com/carlini/nn_robust_attacks.

IIS 1836945, IIS 1836938, DBI 1836866, IIS 1845666, IIS 1852606, IIS 1838627, IIS 1837956.

References

Agarwal, A.; Dekel, O.; and Xiao, L. 2010. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT*, 28–40. Citeseer.

Beck, A., and Teboulle, M. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* 2(1):183–202.

Bertsekas, D. P. 2011. Incremental proximal methods for large scale convex optimization. *Mathematical programming* 129(2):163–195.

Chen, P.-Y.; Zhang, H.; Sharma, Y.; Yi, J.; and Hsieh, C.-J. 2017. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *The 10th ACM Workshop on Artificial Intelligence and Security*, 15–26. ACM.

Conn, A. R.; Scheinberg, K.; and Vicente, L. N. 2009. *Introduction to derivative-free optimization*, volume 8. Siam.

Defazio, A.; Bach, F.; and Lacoste-Julien, S. 2014. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, 1646–1654.

Duchi, J. C.; Jordan, M. I.; Wainwright, M. J.; and Wibisono, A. 2015. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory* 61(5):2788–2806.

Dvurechensky, P.; Gasnikov, A.; and Gorbunov, E. 2018. An accelerated method for derivative-free smooth stochastic convex optimization. *arXiv preprint arXiv:1802.09022*.

Gao, X.; Jiang, B.; and Zhang, S. 2018. On the information-adaptive variants of the admm: an iteration complexity perspective. *Journal of Scientific Computing* 76(1):327–363.

Ghadimi, S., and Lan, G. 2013. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization* 23:2341–2368.

Ghadimi, S.; Lan, G.; and Zhang, H. 2016. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming* 155(1-2):267–305.

Gu, B.; Huo, Z.; Deng, C.; and Huang, H. 2018. Faster derivative-free stochastic algorithm for shared memory machines. In *ICML*, 1807–1816.

Gu, B.; Huo, Z.; and Huang, H. 2016. Zeroth-order asynchronous doubly stochastic algorithm with variance reduction. *arXiv preprint arXiv:1612.01425*.

Gu, B.; Huo, Z.; and Huang, H. 2018. Inexact proximal gradient methods for non-convex and non-smooth optimization. In *AAAI*.

Li, H., and Lin, Z. 2015. Accelerated proximal gradient methods for nonconvex programming. In *Advances in neural information processing systems*, 379–387.

Lian, X.; Zhang, H.; Hsieh, C. J.; Huang, Y.; and Liu, J. 2016. A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order. In *Advances in Neural Information Processing Systems*, 3054–3062.

Liu, L.; Cheng, M.; Hsieh, C.-J.; and Tao, D. 2018a. Stochastic zeroth-order optimization via variance reduction method. *CoRR* abs/1805.11811.

Liu, S.; Chen, J.; Chen, P.-Y.; and Hero, A. 2018b. Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications. In *The Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, 288–297.

Liu, S.; Kailkhura, B.; Chen, P.-Y.; Ting, P.; Chang, S.; and Amini, L. 2018c. Zeroth-order stochastic variance reduction for nonconvex optimization. *arXiv preprint arXiv:1805.10367*.

Mine, H., and Fukushima, M. 1981. A minimization method for the sum of a convex function and a continuously differentiable function. *Journal of Optimization Theory & Applications* 33(1):9–23.

Nesterov, Y., and Spokoiny, V. G. 2017. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics* 17:527–566.

Nesterov, Y. 2004. *Introductory Lectures on Convex Programming Volume I: Basic course*. Kluwer, Boston.

Nesterov, Y. 2013. Gradient methods for minimizing composite functions. *Mathematical Programming* 140(1):125–161.

Parikh, N.; Boyd, S.; et al. 2014. Proximal algorithms. *Foundations and Trends® in Optimization* 1(3):127–239.

Reddi, S.; Sra, S.; Poczos, B.; and Smola, A. J. 2016. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. In *Advances in Neural Information Processing Systems*, 1145–1153.

Shamir, O. 2017. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research* 18(52):1–11.

Sokolov, A.; Hitschler, J.; and Riezler, S. 2018. Sparse stochastic zeroth-order optimization with an application to bandit structured prediction. *arXiv preprint arXiv:1806.04458*.

Wainwright, M. J.; Jordan, M. I.; et al. 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning* 1(1–2):1–305.

Xiao, L., and Zhang, T. 2014. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization* 24(4):2057–2075.

Yu, X.; King, I.; Lyu, M. R.; and Yang, T. 2018. A generic approach for accelerating stochastic zeroth-order convex optimization. In *IJCAI*, 3040–3046.