Sharp Analysis of Stochastic Optimization under Global Kurdyka-Łojasiewicz Inequality

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study the complexity of finding the global solution to stochastic nonconvex optimization when the objective function satisfies global Kurdyka-Łojasiewicz (KL) inequality and the queries from stochastic gradient oracles satisfy mild expected smoothness assumption. We first introduce a general framework to analyze Stochastic Gradient Descent (SGD) and its associated nonlinear dynamics under the setting. As a byproduct of our analysis, we obtain a sample complexity of $\mathcal{O}(\epsilon^{-(4-\alpha)/\alpha})$ for SGD when the objective satisfies the so called α-PŁ condition, where α is the degree of gradient domination. Furthermore, we show that a modified SGD with variance reduction and restarting (PAGER) achieves an improved sample complexity of $\mathcal{O}(\epsilon^{-2/\alpha})$ when the objective satisfies the average smoothness assumption. This leads to the first optimal algorithm for the important case of $\alpha = 1$ which appears in applications such as policy optimization in reinforcement learning.

1 Introduction

Nonconvex optimization problems are ubiquitous in machine learning domains such as training deep neural networks [21] or policy optimization in reinforcement learning [50]. Stochastic Gradient Descent (SGD) and its variants are driving the practical success of machine learning approaches. Naturally, understanding the limits of performance of SGD in the nonconvex setting has become an important avenue of research in recent years [20, 4, 28, 42, 22, 14, 57].

We are interested in solving the unconstrained *stochastic*, *nonconvex* optimization problem of the form

$$\min_{x\in\mathbb{R}^d} f(x) := \mathbb{E}_{\xi\sim\mathcal{D}}\left[f_\xi(x)\right], \tag{1}$$

where $f(\cdot)$ is smooth and possibly nonconvex, and ξ is a random vector drawn from a distribution $\mathcal{D}$. Moreover, we are interested in an important special case of (1), when the expectation can be written as the average of n smooth functions:

$$\min_{x\in\mathbb{R}^d} \Big[f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x)\Big]. \tag{2}$$

For a general nonconvex differentiable objective $f : \mathbb{R}^d \to \mathbb{R}$, finding a global minimum of f is in general intractable [40, 52]. There are two common strategies to analyze optimization methods for nonconvex functions. The first one is to scale down the requirements on the solution of interest from global optimality to some relaxed version, e.g., first-order stationary point. However, such solutions do not exclude the possibility of approaching a suboptimal local minima or a saddle point. Another approach is to study nonconvex problems with additional structural assumption in the hope of convergence to global solutions. In this direction, several relaxations of convexity have been proposed

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

and analyzed, for instance, star convexity, quasar-convexity, error bounds condition, restricted secant inequality, and quadratic growth [25, 23, 22]. Many of the aforementioned relaxations have limited application in real-world problems.

Recently, there has been a surge of interest in the analysis of functions satisfying the so-called Kurdyka-Łojasiewicz (KŁ) inequality [8, 9]. Of particular interest is the family of functions that satisfy global KŁ inequality. Specifically, we say that $f(\cdot)$ satisfies *(global) KŁ inequality* if there exists some continuous function $\phi(\cdot)$ such that $||\nabla f(x)|| \geq \phi\left(f(x) - \inf_x f(x)\right)$ for all $x \in \mathbb{R}^d$. If this inequality is satisfied for $\phi(t) = \sqrt{2\mu}\, t^{1/\alpha}$, then we say that *(global) α-PŁ condition* is satisfied for $f(\cdot)$. The special case of KŁ condition, 2-PŁ, often referred as Polyak-Łojasiewicz or PŁ condition, was originally discovered independently in the seminal works of B. Polyak, T. Ležanski and S. Łojasiewicz [46, 31, 36, 37]. Notably, this class of problems has found many interesting emerging applications in machine learning, for instance, policy gradient (PG) methods in reinforcement learning [39, 1, 54], generalized linear models [38], over-parameterized neural networks [2, 55], linear quadratic regulator in optimal control [12, 18], and low-rank matrix recovery [7].

Despite increased popularity of KŁ and α-PŁ assumptions, the analysis of stochastic optimization under it remains limited and the majority of works focus on deterministic gradient methods. Indeed until recently, only the special case of α-PŁ with $\alpha = 2$ was mainly addressed in the literature [25, 22, 28, 51, 47]. In this paper, we study the sample complexities of stochastic optimization for the broader class of nonconvex functions with global KŁ property.

1.1 Related Works and Open Questions

Stochastic gradient descent. A plethora of existing works has studied the sample complexity of SGD and its variants for finding an ϵ-stationary point of general nonconvex function f, that is, a point $x \in \mathbb{R}^d$ for which $\mathbb{E}[||\nabla f(\hat{x})||] \leq \epsilon$. For instance, [20] showed that for a smooth objective (one with Lipschitz gradients) under bounded variance (BV) assumption, SGD with properly chosen stepsizes reaches an ϵ-stationary point with the sample complexity of $\mathcal{O}(\epsilon^{-4})$. Recently, [28, 54] further extended the result under a much milder *expected smoothness* (ES) assumption on stochastic gradient. While this sample complexity is known to be optimal for general nonconvex functions, a naive application of this result to the function value using α-PŁ condition would lead to a suboptimal $\mathcal{O}(\epsilon_f^{-4/\alpha})$ sample complexity for finding an ϵ_f-optimal solution, i.e., $\mathbb{E}\left[f(x) - f^\star\right] \leq \epsilon_f$. Recently, [19] studied SGD and established convergence rates for α-PŁ functions under BV assumption. Their sample complexity result is $\mathcal{O}(\epsilon_f^{-(4-\alpha)/\alpha})$ in our notation. Later [33] considered SGD scheme with random reshuffling under local and global KŁ conditions and provided convergence in the iterates for $\alpha \in (1, 2]$. We note that our proof techniques are different from [19] and [33] and are not limited merely to the case of BV assumption. In this work, we will answer the following open question: *What is the exact performance limit of* SGD *under global KŁ condition and a more practical model of stochastic oracle?*

Variance reduction. There has been extensive research on development of algorithms which improve the dependence on n and/or ϵ for both problems (1) and (2) (over simple methods such as SGD and Gradient Descent (GD)). One important family of techniques[1] is *variance reduction*, which has emerged from the seminal work of Le Roux et. al [48]. The main idea of *variance reduction* is to make use of the stochastic gradients computed at previous iterations to construct a better gradient estimator at a relatively small computational cost. Various modifications, generalizations, and improvements of the variance reduction technique appeared in subsequent work, for instance, [48, 27, 15] to name a few.

Finite-sum case. A number of recently proposed algorithms such as SNVRG [56], SARAH [43], STORM [14], SPIDER [16], and PAGE [34] achieve the sample complexity $\mathcal{O}\big(n + \frac{\sqrt{n}}{\epsilon^2}\big)$ when minimizing a general nonconvex function with finite sum structure (2). This result is also known to be optimal in this setting [34]. [26] studies SARAH in finite sum case under local KŁ assumption and proves convergence in the iterates. The study in [26] is only asymptotic analysis and the dependence on the parameters κ and n, which are important in practice for quantifying the improvement over GD and SGD are ignored. [32] proposes an SVRG-based algorithm for KŁ functions and [53, 43, 44] study other variance reduction techniques, but they only analyze the special case $\alpha = 2$. Under

[1] Another independent direction is to make use of higher order information [41, 17, 3].

2-PŁ condition, these methods further improve to $\mathcal{O}\big((n + \kappa\sqrt{n}) \log(\frac{1}{\epsilon_f})\big)$ sample complexity[2] for finding an ϵ_f-optimal solution. However, it is not clear if it is possible to provide any non-asymptotic guaranties for variance reduced methods under α-PŁ condition for any $\alpha \in [1, 2)$. In our work, we will answer the following open question: *What is the extent of improvement any variance reduction scheme can provide under global α-PŁ condition for finite-sum objectives of the form* (2)*?*

Online/streaming case. While variance reduction methods have been initially designed for problems of the form (2), it was later discovered that they also improve over SGD when solving (1) [30, 35][3]. The analysis of these methods was obtained for *general nonconvex* functions (for minimizing the norm of the gradient, $\mathbb{E}\left[\|\nabla f(\hat{x})\|\right] \leq \epsilon$) and later extended to 2-*PŁ objectives* for minimizing the function value, $\mathbb{E}\left[f(x) - f^\star\right] \leq \epsilon_f$. For example, the methods in [56, 43, 16, 34] achieve $\mathcal{O}(\epsilon^{-3})$ complexity improving over $\mathcal{O}(\epsilon^{-4})$ complexity of SGD for finding an ϵ-stationary point. Under the 2-PŁ condition, these results can be extended to global convergence with $\mathcal{O}(\epsilon_f^{-1})$ sample complexity [34]. However, in contrast to a general nonconvex case, variance reduction under 2-PŁ assumption does not show any improvement over SGD in terms of ϵ_f. We highlight that all existing analysis of variance reduction under α-PŁ condition *is restricted only to a special case* $\alpha = 2$. We refer the reader to Appendix C, where we elaborate on the key difficulties in the analysis for the cases $\alpha \in [1, 2)$. Since the direct analysis for $\alpha \in [1, 2)$ is challenging, in order to obtain the global convergence in this setting, one could naively translate the complexity for finding a stationary point of a general nonconvex function (which is $\mathcal{O}(\epsilon^{-3})$) to convergence in a function value by using α-PŁ condition: $\sqrt{2\mu}\left(f(\hat{x}) - f^\star\right)^{1/\alpha} \leq ||\nabla f(\hat{x})||$. This would result in $\mathcal{O}\big(\epsilon_f^{-3/\alpha}\big)$ sample complexity. However, there are two serious issues with this approach. First, this complexity does not provide any improvement over SGD in the most interesting practical case $\alpha = 1$ and gives strictly worse result for all $\alpha > 1$. Second, the guarantees for general nonconvex optimization hold on average, in the sense that the point $\hat{x}$ is sampled uniformly from all the iterates of the algorithm. It would be more desirable to instead derive last iterate convergence guarantees under KŁ (α-PŁ) condition. In this work, we will address the following open question: *Is it possible to accelerate the $\mathcal{O}\big(\epsilon_f^{-(4-\alpha)/\alpha}\big)$ sample complexity of* SGD *under global α-PŁ condition for stochastic objectives of the form* (1)*?*

1.2 Contributions

In this work, we provide an extensive analysis of stochastic optimization under global KŁ condition and answer all the above questions in the affirmative. More precisely, our contributions are as follows

- We provide a new framework for the analysis of the dynamics of SGD under global KŁ condition (see Section 3). It is based on the analysis of SGD dynamic which is governed by a recursive inequality (see Equation (6)). As a result of this analysis, we introduce a set of conditions (see Theorem 1) for designing proper stepsizes to guarantee convergence.
- Using this framework, we provide sharp analysis of SGD under a general ES assumption (Assumption 4) and demonstrate that the sample complexity $\mathcal{O}\big(\epsilon_f^{-(4-\alpha)/\alpha}\big)$ is tight for the dynamical system describing SGD.
- Next, we propose PAGER, a new variance reduction scheme with parameter restart. A carefully chosen sequence of parameters of PAGER allows the algorithm to adapt to the nonconvex geometry of the problem and establish state-of-the-art convergence guarantees for minimizing α-PŁ functions. In online setting (1), we obtain $\mathcal{O}\left(\epsilon_f^{-2/\alpha}\right)$ sample complexity of PAGER, which beats $\mathcal{O}\left(\epsilon_f^{-(4-\alpha)/\alpha}\right)$ complexity of SGD for the whole spectrum of parameters $\alpha \in [1, 2)$. In particular, for the important special case of 1-PŁ, this leads to the first optimal algorithm with $O(\epsilon_f^{-2})$ sample complexity, which already matches with the lower bound known for stochastic convex optimization [40].
- Furthermore, we obtain faster rates with PAGER in finite sum case (2), providing the first acceleration over GD and SGD under α-PŁ condition.

[2] $\kappa = \mathcal{L}/\mu$ is the analogue of condition number, $\mathcal{L}$ is defined in Assumption 6.

[3] Under additional assumptions such as smoothness of individual functions $f_\xi(\cdot)$ or even milder condition such as *average L-smoothness* (Assumption 6).

In Table 1, we summarize the sample complexity results for stochastic optimization under α-PŁ and BV assumptions. We also establish sharp convergence results for convergence in the iterates to the set $X^\star$ of optimal points and provide a summary in Table 2 in the Appendix.

Table 1: Summary of sample complexity results for α-PŁ functions (Assumption 3) under average $\mathcal{L}$-smoothness (Assumptions 6) and bounded variance (Assumptions 5). Quantities: α = PŁ power; μ = PŁ constant; $\kappa = \mathcal{L}/\mu$; σ^2 = variance. The entries of the table show the expected number of stochastic gradient calls to achieve $\mathbb{E}\left[f(x_k) - f^\star\right] \leq \epsilon_f$.

Method	Finite sum case	Online case
GD	$\mathcal{O}\left(n\kappa\left(\frac{1}{\epsilon_f}\right)^{\frac{2-\alpha}{\alpha}}\right)$	N/A
SGD	$\mathcal{O}\left(\frac{\kappa\sigma^2}{\mu}\left(\frac{1}{\epsilon_f}\right)^{\frac{4-\alpha}{\alpha}}\right)$	$\mathcal{O}\left(\frac{\kappa\sigma^2}{\mu}\left(\frac{1}{\epsilon_f}\right)^{\frac{4-\alpha}{\alpha}}\right)$
PAGER	$\widetilde{\mathcal{O}}\left(n + \sqrt{n}\kappa\left(\frac{1}{\epsilon_f}\right)^{\frac{2-\alpha}{\alpha}}\right)$ (new)	$\mathcal{O}\left(\left(\frac{\sigma^2}{\mu} + \kappa^2\right)\left(\frac{1}{\epsilon_f}\right)^{\frac{2}{\alpha}}\right)$ (new)

2 Assumptions and Discussion

In this section, we introduce the assumptions we make throughout the paper.

Assumption 1. *The gradient of $f(\cdot)$ is Lipschitz continuous, that is, for all x and y, $\|\nabla f(x) - \nabla f(y)\| \leq L\,\|x - y\|$, $L > 0$ is referred to as the Lipschitz constant.*

Furthermore, we assume that the objective function f is lower bounded, i.e., $f^* := \inf_x f(x) > -\infty$, and it satisfies the following inequality

Assumption 2 (global KŁ or global Kurdyka-Łojasiewicz)**.** *Let $\phi : \mathbb{R}^+ \to \mathbb{R}^+$ be a continuous function such that $\phi(0) = 0$ and $\phi^2(\cdot)$ is convex. The function $f(\cdot)$ is said to satisfy global Kurdyka-Łojasiewicz inequality if*

$$||\nabla f(x)|| \geq \phi\left(f(x) - f^*\right) \quad \textit{for all } x \in \mathbb{R}^d. \tag{3}$$

Assumption 3 (α-PŁ or Polyak-Łojasiewicz)**.** *There exists $\alpha \in [1, 2]$ and $\mu > 0$ such that*

$$\|\nabla f(x)\|^\alpha \geq (2\mu)^{\alpha/2}\left(f(x) - f^*\right) \quad \textit{for all } x \in \mathbb{R}^d. \tag{4}$$

We refer to α as the PŁ power and μ as the PŁ constant.

It is straightforward to see that the α-PŁ is a special case of KŁ with $\phi(t) = \sqrt{2\mu}\, t^{1/\alpha}$.

Connections with other assumptions. Another commonly adopted way to define the global KŁ property is to assume that $\rho'\left(f(x) - f^\star\right) \cdot \|\nabla f(x)\| \geq 1$ for all $x \in \mathbb{R}^d$, where $\rho(t)$ is called a disingularizing function and $\rho'(\cdot)$ denotes its derivative. Moreover, disingularizing function satisfies the following conditions, it is continuous, concave, $\rho(0) = 0$, and $\rho'(t) > 0$ [33].

If the above assumption holds for $\rho(t) := \frac{1}{\theta}t^\theta$ and $\theta > 0$, then Assumption 3 is satisfied with PŁ power $\alpha = \frac{1}{1-\theta}$. However, α-PŁ condition is more general since it allows to consider the case $\alpha = 1$. For example, consider the function of one variable $f(x) = (e^x + e^{-x})/2 - 1$, then $|f'(x)| \geq f(x)$ for all x. Thus $f(x)$ satisfies Assumption 3 with $\alpha = 1$ and $\mu = 1/2$. Moreover, this function is convex, but it does not satisfy inequality $\rho'\left(f(x) - f^\star\right) \cdot \|\nabla f(x)\| \geq 1$ for any choice of $\rho(t)$. We also provide several non-convex problems for which α-PŁ holds with $\alpha \in [1, 2]$ in the Appendix A. Other forms of $\phi(t)$ also appear in practice, e.g., squared cross entropy loss function satisfies the KL condition with $\phi(t) = \min\{t, \sqrt{t}\}$, [49].

The intuition behind the special case $\alpha = 1$ is that the function is allowed to be flat near the set of optimal points $X^\star = \arg\min_x f(x)$.

Assumption 4 (k-ES, Expected Smoothness of order k)**.** *The stochastic gradient estimator $g_k(x, \xi)$ is an unbiased estimate of the gradient $\nabla f(x)$ at any given point x and its second moment satisfies*

$$\mathbb{E}\left[\|g_k(x,\xi)\|^2\right] \leq 2A \cdot h\big(f(x) - f^*\big) + B \cdot ||\nabla f(x)||^2 + \frac{C}{b_k}, \quad \textit{for all } x \in \mathbb{R}^d, \tag{5}$$

where A, B, C are non-negative constants. $h : \mathbb{R}^+ \to \mathbb{R}^+$ is a concave continuously differentiable function with $h'(t) \geq 0$, $h(0) = 0$. The expectation is taken over random vector $\xi \sim \mathcal{D}$. We call b_k the cost of such estimator.

This assumption encompasses previous assumptions in the literature. For instance, it is straightforward to see that an estimator satisfies the standard *bounded variance assumption* [20] when $h(t) = 0, B = 1$, and $b_k = 1$ in (5). Gradient estimators with *relaxed growth assumption* [6, 10] are also special cases of (5) for $h(t) = 0$ and $b_k = 1$. A closely related assumption to the relaxed growth was introduced in [51] which holds when $h(t) = 0$ and $C = 0$ in (5). *Expected smoothness assumption* [28, 23, 54] is the closest assumption to k-ES and it holds when $h(t) = t$ and $b_k = 1$. Notably, ES assumption is satisfied in practical scenarios such as mini-batching, importance sampling and compressed communication [28]. More recently, it has been shown that PG method with softmax policies and log barrier regularization can be modeled using ES assumption [54]. Note that due to the first term in (5), i.e., $2A \cdot h(f(x) - f^*)$, the second moment can be large when the objective gap at x is large. Such property is not captured by standard *bounded variance* (Assumption 5). The flexibility and advantages of introducing such additional term are elaborated in detail in the literature [23, 28, 24, 54].

We highlight two special cases for the sequence b_k: $b_k = \Theta(k^\tau)$ with $\tau \geq 0$ and $b_k = \Theta(q^k)$ with $q > 1$. For example, Monte Carlo sampling and mini-batching allow us to design estimators with such $\{b_k\}_{k\geq 0}$ sequences. When a gradient estimator satisfies Assumption 4, unless b_k is bounded for all k, it essentially means that we have a mechanism to reduce its variance. More precisely, the variance decreases according to the sequence $1/b_k$. As we show in Section 4, if such estimator exists, it results in a better convergence rate compared to vanilla SGD and the improvement is captured by sequence b_k. On the other hand, usually the access to such estimator comes with a *cost* proportional to b_k, e.g., mini-batch setting described in Section 4. Thus we refer to b_k as the *cost* of the estimator $g_k(x, \xi)$.

3 Stochastic Gradient Method

Algorithm 1 summarizes the steps of a slightly modified SGD which we analyze in this work. We call this algorithm SGD with restarts. This algorithm updates the point x for T number of iterations within an inner-loop. Note that, in the inner-loop, the step-size remains unchanged and the iterates are updated via $x_{t+1} = x_t - \eta g_k(x_t, \xi_t)$, where η is the step-size and $\{\xi_t\}_{t\geq 0}$ are independent random vectors. The cost b_k of the gradient estimator $g_k(x, \xi)$ remain the same within the inner loop of Algorithm 1.

Algorithm 1: SGD with restarts

```
Initialization: x, T, K, {η_k : k = 0, ..., K − 1}
for k = 0, ..., K − 1 do
    η ← η_k
    for t = 0, ..., T − 1 do
        x ← x − η g_k(x, ξ_t)
return x
```

3.1 Dynamics of SGD

Let $\{x_t\}_{t\geq 0}$ be the sequence of points generated by the inner loop of Algorithm 1, and Assumptions 1, 2, 4 are satisfied. Then the dynamics of SGD in the inner-loop of Algorithm 1 is characterized by

Lemma 1. *Under Assumptions 1, 2, and 4 with constant cost, i.e., $b := b_k$, we obtain*

$$\delta_{t+1} \leq \delta_t + a\eta^2 \cdot h\big(\delta_t\big) - \frac{\eta}{2}\phi^2(\delta_t) + \frac{d\eta^2}{b}, \tag{6}$$

where $\delta_t := \mathbb{E}\left[f(x_t) - f^\star\right]$, $a := LA$, $d := \frac{LC}{2}$, $\eta := \eta_k$.

Understanding the dynamics of this recursion, allows us to establish the global convergence of SGD. Our approach consists of two main steps: i) Finding the stationary[4] point of (6) when the inequality is replaced by equality and for a fixed step-size, $\eta_k = \eta$, which we denote by $r(\eta)$. ii) Selecting the step-sizes $\{\eta_k\}$ and sequence $\{b_k\}_{k\geq 0}$, such that the corresponding stationary points $\{r(\eta_k)\}_{k\geq 0}$ (defined below) converge to zero as k increases.

[4] Stationary point of a dynamic is its convergence point.

The stationary point of (6) after replacing inequality with equality must satisfy the equation:

$$a\eta^2 h(t) + \frac{d\eta^2}{b} = \frac{\eta}{2}\phi^2(t). \tag{7}$$

Let us call this stationary point $r(\eta)$. To complete the first step, we approximate $r(\eta)$ by a polynomial function of η. In other words, we find $\nu \in \mathbb{R}^+$ such that $r(\eta) = \Theta(\eta^\nu)$.

For the second step of our framework, we should design the stepsizes. Next result introduces a set of conditions that allow us to design the stepsizes, which will guarantee convergence. The detailed derivations are presented in the Appendix.

Theorem 1. *Suppose there exist $\nu \geq 0, \{\omega_j\}_{j\geq 0}$, and $\zeta \geq 0$ such that $\eta_k = \Theta(k^{-\zeta})$, $r(\eta_k) = \Theta(k^{-\zeta\nu})$, $|1-\omega_k| < 1$, and*

$$1 + a\eta_k^2 h'\big(r(\eta_k)\big) - \eta_k \phi'\big(r(\eta_k)\big)\phi\big(r(\eta_k)\big) = 1 - \omega_k k^{-1}. \tag{8}$$

Then, δ_k=$\mathcal{O}(k^{-\zeta\nu})$ and the iteration complexity of Algorithm 1 with T=$\Omega(1/\min_j \omega_j)$ is $\mathcal{O}(\epsilon_f^{-1/(\zeta\nu)})$.

As a consequence of Theorem 1, we present the iteration complexity of SGD for α-PŁ functions.

Corollary 1. *Consider a special case of Assumption 4 with $h(t) = t^\beta$ and $b_k = k^\tau$, where $\beta \in (0, 1]$ and $\tau \geq 0$. Suppose the objective function f satisfies Assumptions 1 and 3. Let $\gamma := \alpha\beta$. Then, for any $\epsilon_f > 0$, Algorithm 1 returns a point x with $\mathbb{E}\left[f(x) - f^\star\right] \leq \epsilon_f$ after $N := K \cdot T$ iterations.*

i) If $\gamma = 2$ ($\alpha = 2$ and $\beta = 1$), we have

$$N = \mathcal{O}(\epsilon_f^{-\frac{1}{1+\tau}}), \text{ with } \eta_k = \Theta(k^{-1}).$$

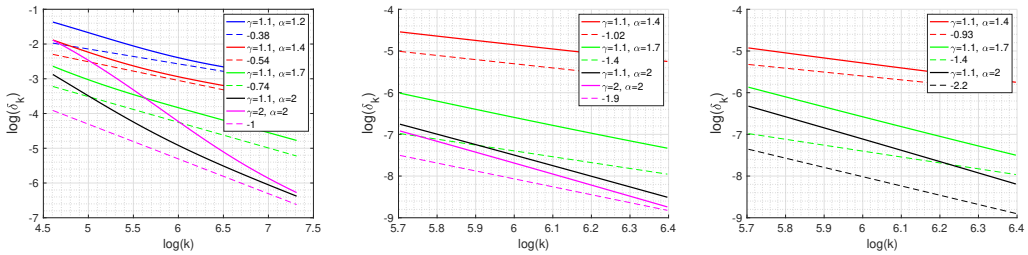
ii) If $\gamma < 2$, we have

$$N = \mathcal{O}\big(\epsilon_f^{-\frac{4-\alpha}{\alpha(\tau+1)}}\big) \text{ with } \eta_k = \Theta(k^{-\frac{\tau+1}{2-\alpha/2}+\tau}) \text{ if } \tau \leq \frac{\gamma}{4-\alpha-\gamma}, \text{and}$$
$$N = \mathcal{O}\big(\epsilon_f^{-\frac{4-\alpha-\gamma}{\alpha}}\big) \text{ with } \eta_k = \Theta(k^{-\frac{2-\gamma}{4-\alpha-\gamma}}) \text{ if } \tau > \frac{\gamma}{4-\alpha-\gamma}.$$

To verify the above result empirically, we simulated δ_t in (6) throughout all iterations of Algorithms 1 for different sets of parameters and presented the results in Figure 1 along with their corresponding convergence rates given in Corollary 1. As it is shown in these figure, the above convergence rates correctly capture the behaviour of the dynamics in (6). As an example, in Figure 1(b), the red solid curve shows the rate of δ_k, i.e., $\log(\delta_k)$ as a function of $\log(k)$ for $\gamma = 1.1, \alpha = 1.4$, and $\tau = 0.9$. Based on Corollary 1, the convergence rate of Algorithm 1 for this setting is $\mathcal{O}(\epsilon_f^{-0.98})$ or equivalently $\log(\delta_k) = (-\frac{\alpha(\tau+1)}{4-\alpha})\log(k) \approx -1.02\log(k)$ which is shown by red dashed line. Next, we discuss how the results in Corollary 1 generalizes the existing work in the literature.

Comparison to related works. Authors in [28] studied the convergence of SGD for 2-PŁ objectives, under a stronger assumption than Assumption 4. More precisely, they assumed an estimator that satisfies Assumption 4 with $\tau = 0$ and $h(t) = t$ and obtained the convergence rate of $\mathcal{O}(\epsilon_f^{-1})$. This is consistent with our rate presented in the Corollary 1. It is worth noting that in this setting, $\mathcal{O}(\epsilon_f^{-1})$ is optimal [28].

The authors in [54] studied the performance of SGD for 1-PŁ objectives. Assuming that the gradient estimator satisfies Assumption 4 with $\tau = 0$ and $h(t) = t$, they obtain $\mathcal{O}(\epsilon_f^{-3})$ sample complexity. This result can be recovered from Corollary 1 by setting $\tau = 0, \gamma = 1$, and $\alpha = 1$. Note that in this case, the cost of each iteration is $b_k = 1$, which means that the iteration complexity coincides with the sample complexity. We note that our proof technique is different than in [54] and allows to consider more general assumptions. Finally, [19] studied SGD with α-PŁ objectives for $\alpha \in [1, 2]$ under bounded variance Assumption 5 with $b_k = 1$ and obtained similar convergence rate to ours. We recover their result as a special case by setting $A = 0$ and $\tau = 0$ in Corollary 1, if we set $T = 1$ we also recover the same (up to a constant) step-sizes $\eta_k = \Theta\left(k^{-\frac{2}{4-\alpha}}\right)$. However, we highlight that our proof technique is different from [19], and more generic since it holds for a general Assumption 4.



(a) $\tau = 0$. (b) $\tau = 0.9$. (c) $\tau = 2$.

Figure 1: Behavior of the dynamics in (6) for $h(t) = t^{\beta}$, $\phi(t) = \sqrt{2\mu}\, t^{1/\alpha}$, $\tau \in \{0, 0.9, 2\}$, and different α, β. Each solid line shows $\log(\delta_k)$ as a function of $\log(k)$, for a given set of parameters and each dashed line shows the corresponding theoretical convergence rate of δ_k presented in Corollary 1. The numbers assigned to dashed lines indicate the slope of those lines. (a) and (b) verify the case corresponding to $\tau \leq \gamma/(4-\alpha-\gamma)$ and (c) verifies the case $\tau > \gamma/(4-\alpha-\gamma)$. Note that the distance between the dashed and solid lines is due to constant factors.

3.2 Sample complexity of SGD

The result of Corollary 1 suggests that by increasing the cost of the gradient estimator b_k over the iterations, one can achieve a better iteration complexity of Algortihm 1. In particular, it improves with τ until it reaches the minimum $N = \mathcal{O}\left(\epsilon_f^{-\frac{4-\alpha-\gamma}{\alpha}}\right)$ at $\tau = \frac{\gamma}{4-\alpha-\gamma}$ and does not change for larger values of τ. However, we are merely interested in the iteration complexity in practice, since the computational cost at each iteration can be prohibitively large. A more adequate measure is the total computational cost (sample complexity) of the method. It is interesting whether increasing b_k over the iterations may also result in a better sample complexity for finding an ϵ_f-optimal solution, than for the constant choice, e.g., $b_k = 1$. The following lemma shows the contrary.

Proposition 1. *Let the assumptions of Corollary 1 hold, $b_k = \Theta\left(k^{\tau}\right)$, $T = \Theta\left(1\right)$. Then the expected total computational cost (sample complexity) of Algorithm 1 is*

$$\textit{cost} := T \cdot \sum_{k=0}^{K-1} b_k = \begin{cases} \mathcal{O}\left(\epsilon_f^{-\frac{4-\alpha}{\alpha}}\right) & \textit{for } 0 \leq \tau \leq \frac{\gamma}{4-\alpha-\gamma}, \\ \mathcal{O}\left(\epsilon_f^{-\frac{(4-\alpha-\gamma)(\tau+1)}{\alpha}}\right) & \textit{for } \tau > \frac{\gamma}{4-\alpha-\gamma}. \end{cases}$$

The above result implies that increasing the cost of the gradient estimator with iterations does not improve the total sample complexity of Algorithm 1. Therefore, one can simply select $b_k = 1$ ($\tau = 0$) and obtain $\mathcal{O}\left(\epsilon_f^{-\frac{4-\alpha}{\alpha}}\right)$ sample complexity.

3.3 Tightness of rates in Corollary 1

In this section, we show that when $\tau = 0$, the convergence rates presented in Corollary 1 are tight for the dynamic (6) describing the progress of SGD. More precisely, if there exists a function f and a gradient estimator satisfying the assumptions in Corollary 1 such that its corresponding recursive inequality (6) is an equality, then its convergence rate, presented in Corollary 1 cannot be improved by any choices of stepsizes $\{\eta_k\}_{k\geq 0}$. Next proposition summarizes our results about the tightness of our convergence rates in Corollary 1.

Proposition 2. *Consider the following recursion*

$$\delta_{k+1} = \delta_k + a\eta_k^2 \cdot h\big(\delta_k\big) - \frac{\eta_k}{2}\phi^2(\delta_k) + \frac{d\eta_k^2}{b_k}, \quad \textit{for all } k \geq 0,$$

where $a \geq 0$, $d > 0$, $h(t) = t^{\beta}$ with $\beta \in (0, 1]$, $\phi(t) = \sqrt{2\mu}t^{1/\alpha}$ with $\alpha \in [1, 2]$, and $b_k = \Theta(1)$. Then $\delta_k = \Omega(k^{-\frac{\alpha}{4-\alpha}})$ for any sequence of $\{\eta_k\}_{k\geq 0}$. Moreover, this rate is achieved by the choice $\eta_k = \Theta(k^{-\frac{1}{2-\alpha/2}})$.

Algorithm 2: PAGER (PAGE with restarts)

Initialization: $\bar{x}_0, \bar{g}_0, K, \{\Lambda_k = (\eta_k, T_k, p_k, b_k, b'_k) : k = 0, ..., K-1\}$
for $k = 0, \ldots, K-1$ **do**
  $(x_0, g_0) \leftarrow (\bar{x}_k, \bar{g}_k)$
  $(\eta, p, b, b') \leftarrow (\eta_k, p_k, b_k, b'_k)$
  for $t = 0, \ldots, T_k - 1$ **do**
    $x_{t+1} = x_t - \eta g_t$
    Sample $\chi \sim \text{Bernoulli}(p)$
    if $\chi = 1$ **then**
      $g_{t+1} = \frac{1}{b} \sum_{i=1}^{b} \nabla f_{\xi^i_{t+1}}(x_{t+1})$
    else
      $g_{t+1} = g_t + \frac{1}{b'} \sum_{i=1}^{b'} \nabla f_{\xi^i_{t+1}}(x_{t+1}) - \frac{1}{b'} \sum_{i=1}^{b'} \nabla f_{\xi^i_{t+1}}(x_t)$
  $(\bar{x}_{k+1}, \bar{g}_{k+1}) \leftarrow (x_{t+1}, g_{t+1})$
Return: $\bar{x}_K$

4 Faster Rates with Variance Reduction

To simplify the exposition of the results in this section, let us assume that $g_k(x_t, \xi_t)$ is constructed explicitly via mini-batching $g_k(x_t, \xi_t) := \frac{1}{b_k} \sum_{i=1}^{b_k} \nabla f_{\xi^i_t}(x_t)$, where $\xi_t := \left(\xi^1_t, \ldots, \xi^{b_k}_t\right)$ is a random vector of independent entries, ξ_t are independent for all iterations, $\{\nabla f_{\xi^i_t}(x_t)\}_{i=1}^{b_k}$ are queries provided by an oracle such that $\mathbb{E}[\nabla f_{\xi^i_t}(x_t)] = \nabla f(x_t)$ and $\mathbb{E}[||\nabla f_{\xi^i_t}(x_t) - \nabla f(x_t)||^2] \leq \sigma^2$ for all $t \geq 0$. The variance of this estimator diminishes linearly in the size of the mini-batch b_k, i.e., $g_k(x_t, \xi_t)$ satisfies

Assumption 5 (k-BV, bounded variance)**.** *Let Assumption 4 hold with $A = 0$, $B = 1$ and $C = \sigma^2$, i.e.,* $\mathbb{E}\left[\|g_k(x, \xi) - \nabla f(x)\|^2\right] \leq \frac{\sigma^2}{b_k}$.

Additionally, we assume that we have access to a gradient estimator $g'_k(x, \xi)$, which satisfies the following

Assumption 6 (Average $\mathcal{L}$-smoothness (of order k))**.** *Let $g'_k(x, \xi) := \frac{1}{b'_k} \sum_{i=1}^{b'_k} \nabla f_{\xi^i}(x)$ and $g'_k(y, \xi) := \frac{1}{b'_k} \sum_{i=1}^{b'_k} \nabla f_{\xi^i}(y)$ be unbiased mini-batch estimators of the gradient of $f(\cdot)$ at points x and y, respectively for shared stochasticity $\xi^i \sim \mathcal{D}$ for each $i = 1, \ldots, b'_k$ and $\xi = (\xi^1, \ldots, \xi^{b_k})$. Define $\widetilde{\Delta}(x, y) := g'_k(x, \xi) - g'_k(y, \xi)$. The average $\mathcal{L}$-smoothness (of order k) holds if there exists $\mathcal{L} \geq 0$ such that $\mathbb{E}\left[\left\|\widetilde{\Delta}(x, y) - \Delta(x, y)\right\|^2\right] \leq \frac{\mathcal{L}^2}{b'_k} \|x - y\|^2$ for all $x, y \in \mathbb{R}^d$, where $\Delta(x, y) := \nabla f(x) - \nabla f(y)$.*

Remark 1. *The Assumption 6 holds in several standard settings. For instance, if each $\nabla f_{\xi^i}(x)$ is Lipschitz with constant $\bar{L}$ (almost surely or on average), then Assumption 6 holds with $\mathcal{L} \leq \bar{L}$. Another example is when $f(\cdot)$ is of the form* (2) *and $b'_k = n$, then $\mathcal{L} = 0$.*

4.1 PAGER – a new variance reduction for α-PŁ objectives

We remark from the analysis of Algorithm 1 in Section 3 that merely playing with choice of η_k and b_k (chosen as polynomial functions of k) is not sufficient to improve the convergence, hence, we need to construct more sophisticated gradient estimator and reduce the variance using control variate. Now, we highlight the main algorithmic ingredients of our construction. First, let us describe the variance reduced estimator named PAGE, which will be the main building block for our Algorithm 2. PAGE was introduced and analyzed in [34] and is known to be optimal for finding a first order stationary point. Moreover, it is easy to implement and designed via a small modification to mini-batch SGD

$$g_{t+1} = \begin{cases} \frac{1}{b} \sum_{i=1}^{b} \nabla f_{\xi^i_{t+1}}(x_{t+1}), & \text{w.p.} \quad p, \\ g_t + \frac{1}{b'} \sum_{i=1}^{b'} \left(\nabla f_{\xi^i_{t+1}}(x_{t+1}) - \nabla f_{\xi^i_{t+1}}(x_t)\right), & \text{w.p. } 1 - p, \end{cases}$$

where p is a small probability and mini-batch sizes satisfy $b > b'$.

However, while the method looks simple, the extension of its analysis to α-PŁ functions faces several difficulties. [5] Therefore, we introduce a new method, which we call PAGER (Algorithm 2) – a *Probabilistic Average Gradient Estimator with parameter Restart*. It takes as input the sequence of parameters $\{\Lambda_k := (\eta_k, T_k, p_k, b_k, b'_k) : k = 0, ..., K-1\}$, where T_k is the length of stage k, η_k, p_k, b_k, b'_k step-size, probability, and batch-sizes at stage k. PAGER updates this sequence of parameters in the outer loop $k = 0, \ldots, K-1$ and applies PAGE estimator with a fixed set of parameters in the inner loop $t = 0, \ldots, T_k - 1$. We will select $\{\Lambda_k\}_{k\geq 0}$ depending on the PŁ power α to capture the dependence on the geometry of the problem and establish fast rates for each α in settings (1) and (2).

4.2 Online case

We present convergence guarantees for Algorithm 2 in the setting (1).

Theorem 2. *Let $f(\cdot)$ have the form* (1) *and satisfy Assumptions 1, 3 (with $\alpha \in [1, 2)$), 5 and 6, let the sequences*[6] *in Algorithm 2 be chosen as $b'_k = \Theta\big(2^{\frac{(2-\alpha)k}{\alpha}}\big)$, $p_k = \Theta\big(2^{\frac{-(2-\alpha)k}{\alpha}}\big)$, $b_k = \Theta\big(2^{\frac{2k}{\alpha}}\big)$, $T_k = \Theta\big(2^{\frac{(2-\alpha)k}{\alpha}}\big)$, $\eta_k = \Theta\big(1\big)$. Then, for any $\epsilon_f > 0$ Algorithm 2 returns a point x with $\mathbb{E}\left[f(x) - f^\star\right] \leq \epsilon_f$ after $N := \sum_{k=0}^{K-1} T_k = \mathcal{O}\big(\epsilon_f^{-\frac{2-\alpha}{\alpha}}\big)$ iterations. The expected total computational cost (sample complexity) is $\mathcal{O}\left(\epsilon_f^{-\frac{2}{\alpha}}\right)$.*

Improvement over SGD. Theorem 2 implies that PAGER improves the sample complexity of SGD from $\mathcal{O}\big(\epsilon_f^{-\frac{4-\alpha}{\alpha}}\big)$ to $\mathcal{O}\big(\epsilon_f^{-\frac{2}{\alpha}}\big)$ under α-PŁ condition for the whole spectrum of parameters $\alpha \in [1, 2)$. In the case $\alpha = 1$, which holds in many interesting applications (see Appendix A for examples), this leads to $\mathcal{O}\big(\epsilon_f^{-2}\big)$ sample complexity compared to the best known $\mathcal{O}\big(\epsilon_f^{-3}\big)$ for SGD.

Relation to convex optimization and last iterate convergence. As a consequence of our analysis we obtain *the optimal sample complexity for convex stochastic optimization* under the additional assumption that the iterates of the method remain bounded, i.e., $\|x_t - x^\star\|^2 \leq D$ for all $t \geq 0$, where $x^\star \in \arg\min_x f(x)$.[7] For 1-PŁ objectives, PAGER has $\mathcal{O}\left(\epsilon_f^{-2}\right)$ sample complexity. Since the iterates of the algorithm are bounded, convexity $\langle \nabla f(x), x - x^\star\rangle \geq f(x) - f(x^\star)$ implies 1-PŁ with $\mu = \frac{1}{2D}$. This observation implies convergence of PAGER for convex objectives with $\mathcal{O}\left(\epsilon_f^{-2}\right)$ sample complexity, which is known to be non-improvable for convex stochastic optimization [40].

Moreover, we highlight that this result holds for the *last iterate* of PAGER, while the standard analysis of first order methods for convex functions guarantees convergence for the average iterate [29]. The last iterate convergence for convex objectives was only recently established for SGD by following an involved analysis with a careful control of iterates via suffix-averaging scheme [19].

4.3 Finite sum case

Let $f(\cdot)$ have the finite sum form (2). Then we obtain the following result.

Theorem 3. *Let $f(\cdot)$ have the form* (2) *and satisfy Assumptions 1, 3 (with $\alpha \in [1, 2)$) and 6, let the sequences be chosen as $p_k = \frac{1}{n+1}$, $b'_k = 1$, $b_k = n$, $T_k = \Theta\big(2^{\frac{(2-\alpha)k}{\alpha}}\big)$, $\eta_k = \Theta\big(1\big)$. Then, for any $\epsilon_f > 0$, Algorithm 2 returns a point x with $\mathbb{E}\left[f(x) - f^\star\right] \leq \epsilon_f$ after $N := \sum_{k=0}^{K-1} T_k = \widetilde{\mathcal{O}}\big(n + \sqrt{n}\kappa\epsilon_f^{-\frac{2-\alpha}{\alpha}}\big)$ iterations. The expected total computational cost (sample complexity) is $\widetilde{\mathcal{O}}\big(n + \sqrt{n}\kappa\epsilon_f^{-\frac{2-\alpha}{\alpha}}\big)$.*

Theorem 3 quantifies the improvement of PAGER over GD in the finite sum setting in terms of n and over SGD in terms of ϵ_f, see Table 1 for comparison. Recall that GD has sample complexity $\mathcal{O}\big(n\kappa\epsilon_f^{\frac{-(2-\alpha)}{\alpha}}\big)$. When n is large, we get the improvement of order $\sqrt{n}$. Notice that in the limit $\alpha \to 2$, it matches the best known result for 2-PŁ objectives [34].

[5] We refer the reader to Appendix C, where we explain the challenges in the analysis of variance reduction under α-PŁ condition and show how we overcome these difficulties using the restart strategy.

[6] For brevity, in Theorem 2 we define the input sequences up to constants hidden in $\Theta(\cdot)$ notation. In fact, our analysis allows to specify these constants and we present detailed derivations in Appendix C.

[7] Note that this assumption is mild since it holds for the iterates of PAGER, for example, if we additionally assume that $f(\cdot)$ is coercive, i.e., $f(x) \to \infty$ for $x \to \infty$.

References

[1] Alekh Agarwal, Sham M. Kakade, Jason D. Lee, and Gaurav Mahajan. On the Theory of Policy Gradient Methods: Optimality, Approximation, and Distribution Shift. *arXiv preprint arXiv:1908.00261*, 2020.

[2] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A Convergence Theory for Deep Learning via Over-Parameterization. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR, 2019.

[3] Yossi Arjevani, Yair Carmon, John C. Duchi, Dylan J. Foster, Ayush Sekhari, and Karthik Sridharan. Second-order information in non-convex stochastic optimization: Power and limitations. In *Proceedings of Thirty Third Conference on Learning Theory*, page 242–299, 2020.

[4] Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *arXiv preprint arXiv:1912.02365*, 2019.

[5] Jonathan Baxter and Peter L. Bartlett. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research*, page 319–350, 2001.

[6] Dimitri P Bertsekas and John N Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM Journal on Optimization*, 2000.

[7] Yingjie Bi, Haixiang Zhang, and Javad Lavaei. Local and Global Linear Convergence of General Low-rank Matrix Recovery Problems. *arXiv preprint arXiv:2104.13348*, 2021.

[8] Jérôme Bolte, Aris Daniilidis, and Adrian Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.

[9] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1):459–494, 2014.

[10] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

[11] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[12] Jingjing Bu, Afshin Mesbahi, Maryam Fazel, and Mehran Mesbahi. LQR through the lens of first order methods: Discrete-time case. *arXiv preprint arXiv:1907.08921*, 2019.

[13] Xin Chen, Niao He, Yifan Hu, and Zikun Ye. Efficient Algorithms for Minimizing Compositions of Convex Functions and Random Functions and Its Applications in Network Revenue Management. *arXiv preprint arXiv:2205.01774*, 2022.

[14] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex SGD. *Advances in neural information processing systems*, 32, 2019.

[15] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[16] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. SPIDER: Near-Optimal Non-Convex Optimization via Stochastic Path-Integrated Differential Estimator. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[17] Cong Fang, Zhouchen Lin, and Tong Zhang. Sharp analysis for nonconvex sgd escaping from saddle points. In *Proceedings of the Thirty-Second Conference on Learning Theory*, Proceedings of Machine Learning Research, pages 1192–1234, 2019.

[18] Ilyas Fatkhullin and Boris Polyak. Optimizing static linear feedback: Gradient method. *SIAM Journal on Control and Optimization*, 59(5):3887–3911, 2021.

[19] Xavier Fontaine, Valentin De Bortoli, and Alain Durmus. Convergence rates and approximation results for SGD and its continuous-time counterpart. In *Conference on Learning Theory*, pages 1965–2058. PMLR, 2021.

[20] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[21] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016.

[22] Robert Gower, Othmane Sebbouh, and Nicolas Loizou. SGD for structured nonconvex functions: Learning rates, minibatching and interpolation. In *International Conference on Artificial Intelligence and Statistics*, pages 1315–1323. PMLR, 2021.

[23] Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General analysis and improved rates. In *International Conference on Machine Learning*, pages 5200–5209. PMLR, 2019.

[24] Benjamin Grimmer. Convergence rates for deterministic and stochastic subgradient methods without lipschitz continuity. *SIAM Journal on Optimization*, 29(2):1350–1365, 2019.

[25] Mark Schmidt Hamed Karimi, Julie Nutini. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. *arXiv preprint arXiv:1608.04636v4*, 2016.

[26] Jia Hu, Congying Han, Tiande Guo, and Tong Zhao. On the Convergence of Stochastic Splitting Methods for Nonsmooth Nonconvex Optimization. *arXiv: Optimization and Control*, 2021.

[27] Rie Johnson and Tong Zhang. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

[28] Ahmed Khaled and Peter Richtárik. Better theory for SGD in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

[29] Guanghui Lan. *First-Order and Stochastic Optimization Methods for Machine Learning*. Springer Series in the Data Sciences. Springer International Publishing, 2020.

[30] Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex Finite-Sum Optimization Via SCSG Methods. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[31] Tadeusz Ležanski. Gradient methods for minimizing functionals. *Mathematische Annalen*, 152, 1963.

[32] Qunwei Li, Yi Zhou, Yingbin Liang, and Pramod K. Varshney. Convergence analysis of proximal gradient with momentum for nonconvex optimization. In *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2017.

[33] Xiao Li, Andre Milzarek, and Junwen Qiu. Convergence of random reshuffling under the Kurdyka-Łojasiewicz inequality. *arXiv preprint arXiv:2110.04926*, 2021.

[34] Zhize Li, Hongyan Bao, Xiangliang Zhang, and Peter Richtárik. Page: A simple and optimal probabilistic gradient estimator for nonconvex optimization. *arXiv preprint arXiv:2008.10898*, 2021.

[35] Zhize Li and Jian Li. A Simple Proximal Stochastic Gradient Method for Nonsmooth Nonconvex Optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[36] Stanisław Łojasiewicz. Sur le probleme de la division. *Studia Mathematica*, 18(1):87–136, 1959.

[37] Stanislaw Łojasiewicz. A topological property of real analytic subsets. *Coll. du CNRS, Les équations aux dérivées partielles*, 117(87-89):2, 1963.

[38] Jincheng Mei, Yue Gao, Bo Dai, Csaba Szepesvari, and Dale Schuurmans. Leveraging non-uniformity in first-order non-convex optimization. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[39] Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the Global Convergence Rates of Softmax Policy Gradient Methods. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020.

[40] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. *SIAM Review*, 1983.

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

[42] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pages 2613–2621. PMLR, 2017.

[43] Lam M. Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient. *arXiv preprint arXiv:1703.00102*, 2017.

[44] Lam M. Nguyen, Marten van Dijk, Dzung T. Phan, Phuong Ha Nguyen, Tsui-Wei Weng, and Jayant R. Kalagnanam. Finite-Sum Smooth Optimization with SARAH. *arXiv preprint arXiv:1901.07648*, 2019.

[45] Matteo Papini, Damiano Binaghi, Giuseppe Canonaco, Matteo Pirotta, and Marcello Restelli. Stochastic variance-reduced policy gradient. In *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2018.

[46] Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki*, 3(4):643–653, 1963.

[47] Sashank J Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323. PMLR, 2016.

[48] Nicolas Roux, Mark Schmidt, and Francis Bach. A Stochastic Gradient Method with an Exponential Convergence _Rate for Finite Training Sets. In *Advances in Neural Information Processing Systems*, 2012.

[49] Kevin Scaman, Cedric Malherbe, and Ludovic Dos Santos. Convergence rates of non-convex stochastic gradient descent under a generic lojasiewicz condition and local smoothness. In *International Conference on Machine Learning*, pages 19310–19327. PMLR, 2022.

[50] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic Policy Gradient Algorithms. In *Proceedings of the 31st International Conference on Machine Learning*, Proceedings of Machine Learning Research, 2014.

[51] Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1195–1204. PMLR, 2019.

[52] Stephen A Vavasis. Complexity issues in global optimization: a survey. In *Handbook of global optimization*, pages 27–41. Springer, 1995.

[53] Zhe Wang, Kaiyi Ji, Yi Zhou, Yingbin Liang, and Vahid Tarokh. SpiderBoost and Momentum: Faster Variance Reduction Algorithms. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[54] Rui Yuan, Robert M Gower, and Alessandro Lazaric. A general sample complexity analysis of vanilla policy gradient. *arXiv preprint arXiv:2107.11433*, 2021.

[55] Jinshan Zeng, Shikang Ouyang, Tim Tsz-Kit Lau, Shaobo Lin, and Y. Yao. Global convergence in deep learning with variable splitting via the Kurdyka-Łojasiewicz property. *arXiv: Optimization and Control*, 2018.

[56] Dongruo Zhou, Pan Xu, and Quanquan Gu. Stochastic nested variance reduction for nonconvex optimization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2018.

[57] Yi Zhou, Yingbin Liang, and Huishuai Zhang. Understanding generalization error of SGD in nonconvex optimization. *Machine Learning*, pages 1–31, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] We accurately state the main contributions in introduction.
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] We provide all the technical proofs in Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]