Stochastic, Reshuffling vs Fixed-ordering Methods in Large-sum Minimization

Xiaochun Niu

Jun. - Sept. 2018

Abstract

In this report, I make a summary of stochastic gradient descent methods used to solve large finite-sum minimization problems, which are popular these days. I focus on the three ordering techniques, stochastic, fixed-ordering, and random reshuffling methods, and make a comparison of their convergence rates on different types of functions. Because of the importance of random reshuffling method in implementation, I have done some theoretical works on it. I take a review of the analysis techniques in its previous proofs and get some inspirations from the proofs of SGD. I also apply a powerful tool, Moreau envelope, in the weakly convex setting. An important work of this report is that I give a new proof of RR on smooth nonconvex functions at the convergence rate $O(T^{-\frac{2}{3}})$, which is faster than that of SGD at $O(T^{-\frac{1}{2}})$ in the same setting. Finally, I complete some empirical study to show that RR converges faster than SGD in expectation. Using Moreau envelope, the future works I plan to do will be some convergence analysis of RR on nonsmooth nonconvex functions.

Contents

1 Introduction

We consider the unconstrained large finite-sum minimization problem of the form:

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

where n is a large finite number and the component $f_i(x) : \mathbb{R}^d \to \mathbb{R}$ can be different types of functions. It arises as an increasingly important optimization problem recently on account of its frequent appearance in many applications in large-scale machine learning, deep learning and reinforcement learning, such as regression or more general parameter estimation problems, in which $f_i(x)$ is the loss function denoting the error between the output and the prediction [1–3]; minimization of an expected value of a function, with the expectation taken over a finite probability distribution or approximated by an n-sample average [4]; or distributed optimization in neural networks, where each $f_i(x)$ corresponds to a training data set [5].

A standard way to solve minimization problems is the batch gradient descent method, which uses full gradient information. However, when the data size n is large, with a per-iteration cost proportional to n, it will be very expensive. Another issue with the batch method is that it doesn't provide an easy way to incorporate new data in an online setting. So in this case, stochastic methods are most commonly used, which select only one component function at each iteration to determine the optimal direction.

1.1 Overview of Stochastic Gradient Methods

This subsection makes a brief overview of the stochastic gradient methods, of which there are two main categories: one computes a gradient of f_i and throws it away, and one called "aggregated" keeps the gradient in memory for reuse.

In the first category, when drawing a selection of $f_{i_k}(x)$ in the kth iteration, there are three most popular ordering techniques, stochastic gradient descent method (SGD), which randomly and uniformly selects one index from the set $\{1, 2, \ldots, n\}$; incremental gradient descent method (IG), with a fixed often cyclic order; and random reshuffling gradient descent method (RR), which draws a random permutation before an epoch and then follows this order. Here SGD is a with-replacement method while IG and RR are without-replacement methods which don't allow any repetitions in one epoch.

As for gradient aggregation methods, there are IAG [6], SAG [7], SVRG [8] and SAGA [9], which can also be viewed as variance reduction methods of the first category. By storing the gradient estimates computed in previous iterations, updating one of them in each iteration, and determining the optimal direction as a weighted average of these estimates, these aggregated methods make full use of gradient information of all the component functions to reduce variance and can achieve a linear rate of convergence.

In addition, there are some accelerated techniques such as iteration averaging schemes and momentum accelerations. Accelerated approaches based on the previous ones, such as SGD with some optimal averaging schemes [10] and Katyusha X, which adds a Nesterov-accelerated step to SVRG [11], can perform better even on functions with some relatively bad conditions.

In this report, we focus on the three ordering techniques of the gradient descent method in the first category, compare their analyses on different types of functions and especially study the random reshuffling method (RR), which is more popular in practice but with less theoretical results. Order techniques do matter because different orderings can behave diversely on varying types of functions.

2 Comparison of the Three Ordering Techniques

In this section, we collect and summarize all the related papers and their high-level results to compare the three ordering techniques. The following table shows a summary of the worst-case convergence rates of the three ordering techniques on different types of functions.

Orderings Types of Functions	SGD	IG	RR
convex quadratic	$O(\frac{1}{k})$ [4] [2]	$O(\frac{1}{k^2})$ [12] [2]	$O(\frac{1}{k^2})$ [13] [2]
smooth strongly convex	$O(\frac{1}{k})$ [4] [2]	$O(\frac{1}{k^{2s}})$, $s \in (\frac{1}{2}, 1)$ [12] [2]	$O(\frac{1}{k^2})$ [13] [2]
smooth general convex (ergodic)	$O(\frac{1}{k})$ [4] [2]		
smooth with PL condition [1]			$O(\frac{1}{k^2})$ [13] [2]
smooth nonconvex (weakly convex)	$O(\frac{1}{\sqrt{k}})$ [14] [3]		
sum of nonconvex but convex	$O(\frac{1}{k})$ [11] [2]		
nonsmooth strongly convex	$O(\frac{\log k}{k})$ [10] [2]	$O(\frac{1}{\sqrt{k}})$ [15] [2]	
nonsmooth general convex	$O(\frac{\log k}{\sqrt{k}})$ [10] [2]		
nonsmooth with PL condition [1]			
nonsmooth nonconvex (weakly convex)	$O(\frac{1}{\sqrt{k}})$ [16] [4]		

Table 1: Convergence Rate of Stochastic, Reshuffling, vs Fixed-ordering Methods

[1] A function f, also known as a gradient dominated function, satisfies the PL condition, short for the Polyak-Łojasiewicz condition, meaning that $\exists\ \mu > 0$ s.t.

$$\frac{1}{2}\|\nabla f(x)\|^2 \geq \mu(f(x) - f^*), \quad \forall x,$$

where f^* denotes the minimum function value of f.

[2] In convex setting, the results are the convergence rates of the objective error $\mathbb{E}[f(x_k)] - f^*$ or the square of distance from the minimizer $\mathbb{E}\|x_k - x^*\|^2$. Some of the results, such as those of SGD on smooth strongly convex or general convex functions, are tight and the worst-case examples have been shown. [17]

[3] In smooth setting, convergence rates are of the square of the gradient norm $\mathbb{E}\|\nabla f(x_k)\|^2$.

[4] In nonsmooth weakly convex setting, convergence rates are of the square of the gradient norm of Moreau envelope $\mathbb{E}[\|\nabla e_\lambda f(x_k)\|^2]$, the definition of which will be shown in Section 4.

As shown in the table, in present results, Random reshuffling method converges faster than SGD when they perform on the same types of functions. For example, in smooth strongly convex setting, SGD converges at a rate of $O(\frac{1}{k})$, while RR achieves a rate of

$O(\frac{1}{k^2})$. Maybe the reason is RR doesn't allow any repetitions in one epoch so that it can reduce the noises in the gradient estimates.

Moreover, we can learn that the theories of SGD are relatively complete, while there are still a lot of blanks in the column of RR. However, in fact, what is more commonly used in implement is RR. So in the coming sections, we take a careful look at the analyses of RR and try to get some new meaningful results.

3 Random Reshuffling Method

Random Reshuffling (RR) is a popular approach following an idea between SGD and IG, which samples the component functions randomly but doesn't allow repetitions in each epoch. Specifically, before each epoch t, we draw a permutation σ_t of $\{1, 2, ..., n\}$ randomly from the set $\Gamma = \{\sigma : \sigma \text{ is a permutation of } \{1, 2, ..., n\}\}$ and pass over the functions with this order:

$$x_k^t = x_{k-1}^t - \gamma_t \nabla f_{\sigma_t(k)}(x_{k-1}^t), \quad k = 1, 2..., n,$$

where $\gamma_t > 0$ is a stepsize. A key point is that the permutations σ_t are *i.i.d.* and uniformly distributed over Γ. We set $x_0^{t+1} = x_n^t$ and refer to $\{x_0^t\}$ as the outer iterates.

In the implementation of stochastic gradient algorithms, RR, a without-replacement sampling, is actually more popular. Intuitively, random reshuffling processes all component functions more equally and often leads to better empirical performance. Moreover, without-replacement sampling is often easier and faster to implement, as it requires sequential data access, as opposed to the random data access required by with-replacement sampling.

This section makes a review of the convergence rates proved previously of RR on different types of functions and also the analysis techniques used before. These results are inspiring for further theoretical researches of RR on more complicated functions.

3.1 Summary of Convergence Rates

It is worth mentioning that in [13], the results are non-asymptotic solutions to this problem, while others are all asymptotic ones. Besides, a diminishing stepsize is used in [3], while a constant stepsize is used in [18].

From the above table, we can learn that all the present results require some relatively strong assumptions of smoothness and convexity. Therefore, there are still a lot of works we can do to complete the mathematical theories of RR.

3.2 Review of Proof Techniques

This subsection focuses on the proof of convergence analysis on RR in the case where function f is L-smooth and strongly convex, which is inspiring for our further study.

The key difficulty in analyzing without-replacement method is that it leads to statistically non-independent samples, which greatly complicates analysis. So instead of considering the improvement after each iteration, we always add up iterations in an epoch and take the epoch as a whole into account. In other words, we will consider the improvement after an epoch, the outer iteration.

	Random Reshuffling		
Stepsize Types of Functions	$\alpha_K = O(\frac{1}{K^s})$, $s \in (\frac{1}{2}, 1)$	$\alpha = O(\frac{\log K}{K})$	α
convex quadratic	$O(\frac{1}{K^{2s}})$, [3]	$O(\frac{1}{K^2} + \frac{n^3}{K^3})$ [13]	$O(\alpha^3)$ [18]
smooth strongly convex	$O(\frac{1}{K^{2s}})$ [3]	$O(\frac{1}{K^2} + \frac{n^3}{K^3})$ [13]	$O(\alpha^2)$ [18]
smooth general convex (ergodic)			
smooth with PL condition		$O(\frac{1}{K^2} + \frac{n^3}{K^3})$ [13]	
smooth nonconvex (weakly convex)			
sum of nonconvex but convex			
nonsmooth strongly convex			
nonsmooth general convex			
nonsmooth with PL condition			
nonsmooth nonconvex (weakly convex)			

Table 2: $\mathbb{E}\|x_K - x^*\|^2$-Convergence Rate of Random Reshuffling, $K \in n\mathbb{Z}^+$.

When analyzing the convergence rate of SGD, in convex setting, the central idea is to establish the following inequality using the convexity of f:

$$\begin{aligned}\mathbb{E}\|x_{k+1} - x^*\|^2 &= \mathbb{E}\|x_k - x^*\|^2 - 2\gamma_k\langle x_k - x^*, \mathbb{E}[g_k]\rangle + \gamma_k^2\mathbb{E}\|g_k\|^2 \\ &\leq \mathbb{E}\|x_k - x^*\|^2 - 2\gamma_k\mathbb{E}[f(x_k) - f^*] + \gamma_k^2\mathbb{E}\|g_k\|^2\end{aligned}$$

where $\mathbb{E}[g_k] = \nabla f(x_k)$ is an unbiased estimation of the gradient at x_k. Then there is always a bounded assumption of g_k, so that $\mathbb{E}[g_k] \leq G^2$ and the last variance term can be bounded by a constant. After accumulating the recursions, a desired convergence rate of the objective error will be obtained. [4]

When it comes to the analysis of RR, we find out that the main ideas are similar to that of SGD. The first step is also to establish the equality:

$$\mathbb{E}\|x_0^{t+1} - x^*\|^2 = \mathbb{E}\|x_0^t - x^*\|^2 - 2\gamma_t\mathbb{E}[\langle x_0^t - x^*, \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t)\rangle] + \gamma_t^2\mathbb{E}\|\sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t)\|^2. \quad (2)$$

To get a tighter bound than that of SGD and also noticing that $\sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t)$ is biased, an intuitive way is to split it into two parts. One behaves like the full gradient descent of f at x_0^t and the other one is the error term R^t capturing the effects of random sampling, which is a random variable dependent on $\sigma_t(\cdot)$, where

$$R^t := \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t) - \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_0^t) = \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t) - n\nabla f(x_0^t).$$

Substituting $\sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t) = R^t + n\nabla f(x_0^t)$ into (2) and using the the strong convexity of f, we have

$$\mathbb{E}\|x_0^{t+1} - x^*\|^2 \leq (1 - n\gamma_t C_1)\mathbb{E}\|x_0^t - x^*\|^2 - C_2\|\nabla f(x_0^t)\|^2 - 2\gamma_t\langle x_0^t - x^*, \mathbb{E}[R^t]\rangle + \gamma_t^2\mathbb{E}\|R^t\|^2.$$

What remains to be bounded is the terms $\mathbb{E}[R^t]$ and $\mathbb{E}\|R^t\|^2$, where the L-smooth assumption plays an important role. With the Lipschitz gradients, the error term R^t, a sum of the difference between gradients, can be bounded by the difference between iteration points. We have $\|R^t\| \leq \sum_{k=1}^n \sum_{j=1}^{k-1} \|\nabla f_{\sigma_t(k)}(x_j^t) - \nabla f_{\sigma_t(k)}(x_{j-1}^t)\| \leq \sum_{k=1}^n \sum_{j=1}^{k-1} L\|x_j^t - x_{j-1}^t\|$. Because of the definition of two adjacent iterations and the assumption of bounded gradients, we have $\|R^t\| \leq \sum_{k=1}^n \sum_{j=1}^{k-1} L\gamma_t \|\nabla f_{\sigma_t(k)}(x_{j-1}^t)\| \leq \frac{n(n-1)}{2} GL\gamma_t$, which means $\|R^t\|$ can be bounded by $O(\gamma_t)$. So there holds an estimation, $\|R^t\|^2 \leq O(\gamma_t^2)$.

A more accurate bound of $\mathbb{E}[R^t]$ can be obtained by introducing second-order information, a Lipschitz continuous Hessian, to each component function. [13] In this way, the randomness of permutation σ_t can be used sufficiently. The proof techniques are similar as above. With $H_i(x)$ denoting the Hessian of function f_i at x and H_i denoting $H_i(x^*)$, we split $R^t := \sum_{k=1}^n [\nabla f_{\sigma_t(k)}(x_{k-1}^t) - \nabla f_{\sigma_t(k)}(x_0^t)] = \sum_{k=1}^n [\int_{x_0^t}^{x_{k-1}^t} H_{\sigma_t(k)}(x)dx]$ into three parts, A^t, B^t and C^t, where $A^t + B^t = \sum_{k=1}^n [\int_{x_0^t}^{x_{k-1}^t} H_{\sigma_t(k)} dx] = \sum_{k=1}^n [H_{\sigma_t(k)} \sum_{i=1}^{k-1} (-\gamma_t \nabla f_{\sigma_t(i)}(x_{i-1}^t))]$ using the fixed Hessian at x^*, $A^t := \sum_{k=1}^n [H_{\sigma_t(k)} \sum_{i=1}^{k-1} (-\gamma_t \nabla f_{\sigma_t(i)}(x_0^t))]$ relies on gradients at the initial point x_0^t, $B^t := -\gamma_t \sum_{k=1}^n \{H_{\sigma_t(k)} \sum_{i=1}^{k-1} [\nabla f_{\sigma_t(i)}(x_{i-1}^t) - \nabla f_{\sigma_t(i)}(x_0^t)]\}$ reflects the difference between gradients, and $C^t := \sum_{k=1}^n [\int_{x_0^t}^{x_{k-1}^t} (H_{\sigma_t(k)}(x) - H_{\sigma_t(k)})dx]$ captures the error difference caused by the changing Hessian. The randomness of permutation plays a key role in bounding $\mathbb{E}[A^t]$, with which we have $\mathbb{E}[A^t] = -\frac{n(n-1)}{2}\gamma_t \mathbb{E}_{i\neq j}[H_i \nabla f_j(x_0^t)]$. Then using Lipschitz gradients to bound $\mathbb{E}[A^t]$ and $\|B^t\|$, and similarly, Lipschitz Hessian to bound $\|C^t\|$, will lead to a tighter estimation:

$$-2\gamma_t \langle x_0^t - x^*, \mathbb{E}[R^t]\rangle \leq \frac{1}{2}\gamma_t \mu(n-1)\|x_0^t - x^*\|^2 + \gamma_t^2 n^2 \|\nabla f(x_0^t)\|^2 + \gamma_t^3 n C_1 + \gamma_t^4 n^4 C_2 + \gamma_t^5 n^5 C_3,$$

where C_1, C_2 and C_3 are constants.

From this point of view, comparing with SGD, RR can make a tighter bound of the last variance term than just a constant. It means that RR reduces the variance of the stochastic gradient estimates, therefore, it converges faster.

Another important theorem in [13] is that, as a non-asymptotic solution, its result $O(\frac{1}{K^2} + \frac{n^3}{K^3})$ will be unavoidably dependent on n. A special constructed instance illustrates that a n-free convergence rate $O(\frac{1}{K^{1+\delta}})$ with some $\delta > 0$, a faster one than SGD, is unlikely to hold for RR. Therefore, when considering the nonconvex setting in section 5, we also try to find a non-asymptotic result, but may be dependent on n as well.

3.3 Inspiration in smooth nonconvex setting

Noticing that the present results of RR ask for strong assumptions of convexity and smoothness but the convexity seems not so important in the proofs, a faster convergence rate of RR in the case where f is L-smooth nonconvex may be possible to be obtained by referring to the previous proofs and using some new skills. Here is another inspiration.

In the analysis of the convergence rate of SGD in smooth setting, the main idea is to establish the following inequality:

$$\mathbb{E}[f(x_{k+1})] \leq \mathbb{E}[f(x_k)] - \frac{\gamma_k}{2}\mathbb{E}\|\nabla f(x_k)\|^2 + \frac{L\gamma_k^2}{2}\mathbb{E}\|g_k - \nabla f(x_k)\|^2$$

where $\mathbb{E}[g_k] = \nabla f(x_k)$ is an unbiased estimation of $\nabla f(x_k)$. After adding up the recursions, a desired convergence rate of the gradient norm will be obtained. [19]

Therefore, a possible solution to deal with RR in smooth case is to establish a similar inequality as this and bound the variance term tighter. The proof will be shown is section 5. But we may need some more skills because in our case, f is nonconvex. So the next section will introduce a powerful tool, the Moreau Envelope, in weakly convex setting.

4 Weakly Convex Functions and Moreau Envelope

In this section, we introduce the weakly convex functions and a powerful tool in nonsmooth nonconvex setting, the Moreau envelope, which we will use later.

4.1 Background

Weakly Convex Functions:

A function $f : \mathbb{R}^d \to \mathbb{R}$ is ρ-weakly convex, meaning that the assignment $x \to f(x) + \frac{\rho}{2}\|x\|^2$ is convex. For example, any function of the form $g = h \circ c$, with h convex and Lipschitz and c a smooth map with Lipschitz Jacobian [20], is weakly convex. Minimization problems on weakly convex functions often appear in machine learning tasks, such as nonlinear least squares, exact penalty formulations of nonlinear programs, robust phase retrieval, and matrix factorization problems like NMF [21]

Proximal map:

$$prox_{\lambda\varphi} := \underset{y}{\operatorname{argmin}}\{\varphi(y) + \frac{1}{2\lambda}\|y - x\|^2\}$$

Moreau envelope:

$$e_\lambda\varphi(x) := \min_y\{\varphi(y) + \frac{1}{2\lambda}\|y - x\|^2\},$$

where $\lambda > 0$. Standard results show that as long as $\lambda < \rho^{-1}$, the envelope φ_λ is C^1-smooth with the gradient given by

$$\nabla e_\lambda\varphi(x) := \lambda^{-1}(x - prox_{\lambda\varphi}(x)).$$

Moreover, the norm of the gradient $\|\nabla e_\lambda\varphi(x)\|$ has an intuitive interpretation in terms of near-stationarity for the target problem (1.1). Namely, the definition of the Moreau envelope directly implies that for any $x \to R^d$, the proximal point $\widehat{x} := prox_{\lambda\varphi}(x)$ satisfies

$$\begin{cases} \|\widehat{x} - x\| = & \lambda\|\nabla e_\lambda\varphi(x)\|, \\ \varphi(\widehat{x}) \leq & \varphi(x), \\ dist(0; \partial\varphi(\widehat{x})) \leq & \|\nabla e_\lambda\varphi(x)\|. \end{cases}$$

Thus a small gradient $\|\nabla e_\lambda\varphi(x)\|$ implies that x is near some nearly stationary point $\widehat{x}$ for the objective function f.

Besides, some even better properties of Moreau envelope hold for $0 < \lambda\rho < 1$ [22] :

- $\nabla e_\lambda f(x) := \lambda^{-1}(x - prox_{\lambda f}(x))$ is L-Lipschitz with

$$L = \begin{cases} \frac{\rho}{1-\rho\lambda} & if \;\; \frac{1}{2} \leq \rho\lambda < 1 \\ \frac{1}{\rho} & if \;\; 0 < \rho\lambda < \frac{1}{2} \end{cases}$$

- $\inf f$=$\inf e_\lambda f$.
- $0 \in \partial f(x)$ if and only if $\nabla f_\lambda(x) = 0$, i.e. the stationary points of f and $e_\lambda f$ coincide.
- $\operatorname{argmin} f$ = $\operatorname{argmin} e_\lambda f$.

The properties show that the Moreau envelope has a Lipschitz gradient and that the optimal value and minimizers, respectively, of f and its Moreau envelope coincide.

Moreau envelope is a key construction based on smooth approximations to measure the complexity of minimizing weakly convex functions. On the one hand, it has Lipschitz gradients even though the original function may be nonsmooth, which makes it possible to bound the differences between gradients by the iterators. On the other hand, after introducing Moreau envelope, there are some additional useful information, such as the strong convexity of $f(y) + \frac{1}{2\lambda}\|y - x\|^2$, and the distance bound of $\|x_0^t - \widehat{x}_0^t$, which will be used in our proof in the next section.

4.2 SGD on Nonsmooth Weakly Convex Functions

This subsection considers the stochastic subgradient descent method on nonsmooth weakly convex functions. Due to the lack of smoothness, gradients are replaced by subgradients in the stochastic descent algorithms. A convergence rate at $O(k^{-1/2})$ of the square of gradient norm of Moreau envelope has been proved [16], which is the same as that of SGD on L-smooth nonconvex functions [14].

In weakly convex setting, to analyze the convergence rate of SGD, the most important step is to establish the following inequality using weakly convexity and Moreau envelope [16]:

$$\mathbb{E}[e_{1/2\rho} f(x_{k+1})] \leq \mathbb{E}[e_{1/2\rho} f(x_k)] - \rho\gamma_k \mathbb{E}\|\nabla e_{1/2\rho} f(x_k)\|^2 + 2\rho\gamma_k^2 \mathbb{E}\|g_k\|^2.$$

Considering the proof techniques used in RR, it inspires us with an idea to reduce the last variance term in the above inequality when analyzing the convergence rate of random reshuffling method. RR may perform a better worst-case convergence rate than SGD on weakly convex functions. A proof is given in the next section.

5 RR on smooth Nonconvex Functions

This section gives a proof that random reshuffling method, applied to a L-smooth nonconvex function, achieves an asymptotic convergence rate at $O(T^{-\frac{2}{3}})$ when run with the proper step size schedule, which is faster than the convergence rate at $O(T^{-\frac{1}{2}})$ of SGD in the same setting [14].

5.1 Motivation

Random reshuffling is a more popular sampling technique in stochastic gradient methods because of its convenience in practical use. In implementation, passing over component functions with a randomly drawn permutation is much easier than generating a stochastic number at each iteration. In addition, in machine learning, sometimes when the datasize n of a training set is too large to fit in main memory, we are unable to deal with them in a single machine. If separating the dataset into different groups and using several networked processors to compute together, stochastic selection from the whole dataset is impractical, but random reshuffling still does a good work.

However, most of the present results of random reshuffling require strong assumptions of convexity and smoothness, which a lot of problems in practice could not satisfy. Therefore, we do a further study on random reshuffling method and achieve some better theoretical results on nonconvex functions.

5.2 Problem and Assumptions

Problem

$$\min_x f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{3}$$

where each component $f_i : R^d \to R$ is a nonconvex function with L-Lipschitz gradients, meaning that $\exists L > 0$, s.t. $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$, $\forall x, y$, $\forall 1 \leq i \leq n$.

Assumptions
(A1) L-smoothness: Each component function f_i is L-smooth, so that for $i = 1, ..., n$, there exists a constant L such that $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$.
(A2) Lipschitz Hessian: Component functions are second-order differentiable with a Lipschitz continuous Hessian, that is, using $H_i(x)$ to denote the Hessian of function f_i at x, there exists a constant L_H such that $\|H_i(x) - H_i(y)\| \leq \|x - y\|$, $\forall x, y$, $\forall\ 1 \leq i \leq n$.
(A3) Bounded Gradients: There is a real $G \geq 0$ such that the inequality $\|\nabla f_i(x)\| \leq G$ holds for each f_i $(1 \leq i \leq n)$.

Here is a comment. Because each f_i has a L-Lipschitz gradient, the function f is also L-smooth. Then it can be proved that with the L-smoothness, for $\forall \rho > L$, f is a ρ-weakly convex function, meaning that $f(x) + \frac{\rho}{2}\|x\|^2$ is convex.

Proof. Define $f_{1/\rho}(x) := f(x) + \frac{\rho}{2}\|x\|^2$.

$$\begin{aligned}
\langle \nabla f_{1/\rho}(x) - \nabla f_{1/\rho}(y),\ x - y\rangle &= \langle \nabla f(x) + \rho x - \nabla f(y) - \rho y,\ x - y\rangle \\
&= \rho\|x - y\|^2 + \langle \nabla f(x) - \nabla f(y),\ x - y\rangle \\
&\geq \rho\|x - y\|^2 - \|\nabla f(x) - \nabla f(y)\|\|x - y\| \\
&\geq \rho\|x - y\|^2 - L\|x - y\|^2 \\
&= (\rho - L)\|x - y\|^2 \geq 0
\end{aligned}$$

$\therefore f_{1/\rho}(x) := f(x) + \frac{\rho}{2}\|x\|^2$ is convex. □

In the following analysis, the L-smooth function f is also regarded as a ρ-weakly convex function, where $\rho > L$.

5.3 Algorithm RR

```
Algorithm RR: Random reshuffling method
Initial: $x_0^0 \in R^d$, epoch count $T$.
Step: Epoch $t = 0, 1, ..., T$
        Pick $\sigma_t(\cdot) \in \Gamma = \{\sigma : \sigma \text{ is a permutation of } \{1, 2, ..., n\}\}$ independently uniformly.
        Visit each of the components sequentially and perform the update:
                $x_k^t = x_{k-1}^t - \gamma_t \nabla f_{\sigma_t(k)}(x_{k-1}^t)$,
      where $x_k^t$ denotes the point in the $k$th iteration of the $t$th epoch, $1 \leq k \leq n$. Define $x_0^t = x_n^{t-1}$.
        Sample $T^* \in \{0, 1, ..., T\}$ according to the probability distribution $\mathbb{P}(T^* = t) = \dfrac{\gamma_t}{\sum_{t=0}^T \gamma_t}$.
Return: $x_0^{T^*}$.
```

5.4 Convergence Analysis

Theorem (Random reshuffling method). Let $x_0^{T^*}$ be the point returned by Algorithm RR. Let Assumptions (A1) and (A2) hold. And suppose the stepsize γ_t be such that $\sum\limits_{t=0}^{\infty} \gamma_t = \infty, \sum\limits_{t=0}^{\infty} \gamma_t^p < \infty,\ p = 2, 3, 4$. Then in terms of any constant $\rho > L$, $\widehat{\rho} > \rho$, the estimate holds:

$$\mathbb{E}[\|\nabla e_{1/\widehat{\rho}} f(x_0^{T^*})\|^2] \leq \frac{(f_{1/\widehat{\rho}}(x_0) - \min f) + C_2 n \sum_{t=1}^T {\gamma_t}^3 + C_3 n^5 \sum_{t=1}^T {\gamma_t}^5 + C_4 n^4 \sum_{t=1}^T {\gamma_t}^4}{\widehat{\rho}^{-2} C_1 n \sum_{t=1}^T \gamma_t},$$

where C_1, C_2, C_3, C_4 are constants defined in the proof.

Proof. Notate x_k^t as the kth iterator generated by Algorithm RR in the tth epoch. Set $\widehat{x}_0^t := prox_{f/\widehat{\rho}}(x_0^t)$. We have $\widehat{x}_0^t := prox_{f/\widehat{\rho}}(x_0^t) = \operatorname{argmin}_y\{f(y) + \frac{\widehat{\rho}}{2}\|y - x_0^t\|^2\} = \operatorname{argmin}_y f_{1/\widehat{\rho}}(y)$, so $\nabla f_{1/\widehat{\rho}}(\widehat{x}_0^t) = \nabla f(\widehat{x}_0^t) + \widehat{\rho}(\widehat{x}_0^t - x_0^t) = 0$, from which there is a bound we will use later:

$$\|x_0^t - \widehat{x}_0^t\| = \frac{1}{\widehat{\rho}}\|\nabla f(\widehat{x}_0^t)\| \leq \frac{G}{\widehat{\rho}} \tag{4}$$

For each epoch t, define error term:

$$R^t := \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t) - \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_0^t) = \sum_{k=1}^n \nabla f_{\sigma_t(k)}(x_{k-1}^t) - n\nabla f(x_0^t).$$

We successively deduce:

$$e_{1/\widehat{\rho}} f(x_0^{t+1}) \leq f(\widehat{x}_0^t) + \frac{\widehat{\rho}}{2}\|\widehat{x}_0^t - x_0^{t+1}\|^2$$

$$
\begin{aligned}
&= f(\widehat{x}_0^t) + \frac{\widehat{\rho}}{2}\|\widehat{x}_0^t - x_0^t + \gamma_t \sum_{k=1}^{n} \nabla f_{\sigma_t(k)}(x_{k-1}^t)\|^2 \\
&= f(\widehat{x}_0^t) + \frac{\widehat{\rho}}{2}\|\widehat{x}_0^t - x_0^t + \gamma_t n \nabla f(x_0^t) + \gamma_t R^t\|^2 \\
&= f(\widehat{x}_0^t) + \frac{\widehat{\rho}}{2}\|\widehat{x}_0^t - x_0^t\|^2 - \widehat{\rho} n \gamma_t \langle x_0^t - \widehat{x}_0^t, \nabla f(x_0^t)\rangle - \widehat{\rho}\gamma_t \langle \widehat{x}_0^t - x_0^t, R^t\rangle + \frac{\widehat{\rho}}{2}{\gamma_t}^2 \|n\nabla f(x_0^t) + R^t\|^2 \\
&\leq e_{1/\widehat{\rho}} f(x_0^t) - \widehat{\rho} n \gamma_t \langle x_0^t - \widehat{x}_0^t, \nabla f(x_0^t)\rangle - \widehat{\rho}\gamma_t \langle \widehat{x}_0^t - x_0^t, R^t\rangle + \widehat{\rho}{\gamma_t}^2 n^2 \|\nabla f(x_0^t)\|^2 + \widehat{\rho}{\gamma_t}^2\|R^t\|^2,
\end{aligned} \tag{5}
$$

where the first inequality is due to the definition of Moreau envelope and the last one is by Jensen's inequality.

Take the expectation of (5) over randomness of permutation $\sigma_t(\cdot)$, we have

$$
\begin{aligned}
\mathbb{E}_t[e_{1/\widehat{\rho}} f(x_0^{t+1})] \leq e_{1/\widehat{\rho}} f(x_0^t) &- \widehat{\rho} n \gamma_t \langle x_0^t - \widehat{x}_0^t, \nabla f(x_0^t)\rangle \ (a) - \widehat{\rho}\gamma_t \langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[R^t]\rangle \ (b) \\
&+ \widehat{\rho}{\gamma_t}^2 n^2 \|\nabla f(x_0^t)\|^2 + \widehat{\rho}{\gamma_t}^2 \mathbb{E}_t[\|R^t\|^2] \ (c).
\end{aligned} \tag{6}
$$

Bound the terms (a), (b) and (c) separately.

(a)

Define $f_{1/\widehat{\rho}}(y) := f(y) + \frac{\widehat{\rho}}{2}\|y - x_0^t\|^2$,

$\because$ f is L-smooth, ρ-weakly convex

$\therefore$ $f_{1/\widehat{\rho}}(y) := f(y) + \frac{\widehat{\rho}}{2}\|y - x_0^t\|^2$ is $(L+1)$-smooth, $\widehat{\rho} - \rho$ strongly convex.

$\because$ $\widehat{x}_0^t := prox_{f/\widehat{\rho}}(x_0^t) = \operatorname{argmin}_y\{f(y) + \frac{\widehat{\rho}}{2}\|y - x_0^t\|^2\} = \operatorname{argmin}_y f_{1/\widehat{\rho}}(y)$

$\therefore$ $\nabla f_{1/\widehat{\rho}}(\widehat{x}_0^t) = \nabla f(\widehat{x}_0^t) + \widehat{\rho}(\widehat{x}_0^t - x_0^t) = 0$

$\because$ $\nabla f_{1/\widehat{\rho}}(x_0^t) = \nabla f(x_0^t)$

$\therefore$

$$
\nabla f(x_0^t) = \nabla f(x_0^t) - 0 = \nabla f_{1/\widehat{\rho}}(x_0^t) - \nabla f_{1/\widehat{\rho}}(\widehat{x}_0^t) \tag{7}
$$

Using Theorem 2.1.12 in [23], there is:

$$
\begin{aligned}
-\langle x_0^t - \widehat{x}_0^t, \nabla f(x_0^t)\rangle &= -\langle x_0^t - \widehat{x}_0^t, \nabla f_{1/\widehat{\rho}}(x_0^t) - \nabla f_{1/\widehat{\rho}}(\widehat{x}_0^t)\rangle \\
&\leq -\frac{(\widehat{\rho} - \rho)(L+1)}{\widehat{\rho} - \rho + L + 1}\|\widehat{x}_0^t - x_0^t\|^2 - \frac{1}{\widehat{\rho} - \rho + L + 1}\|\nabla f_{1/\widehat{\rho}}(\widehat{x}_0^t) - \nabla f_{1/\widehat{\rho}}(x_0^t)\|^2 \\
&= -\frac{(\widehat{\rho} - \rho)(L+1)}{\widehat{\rho} - \rho + L + 1}\|\widehat{x}_0^t - x_0^t\|^2 - \frac{1}{\widehat{\rho} - \rho + L + 1}\|\nabla f(x_0^t)\|^2.
\end{aligned} \tag{8}
$$

(c)

$$
\begin{aligned}
\|R^t\| &= \|\sum_{k=1}^{n} \nabla f_{\sigma_t(k)}(x_{k-1}^t) - \sum_{k=1}^{n} \nabla f_{\sigma_t(k)}(x_0^t)\| \\
&\leq \sum_{k=1}^{n} \|\nabla f_{\sigma_t(k)}(x_{k-1}^t) - \nabla f_{\sigma_t(k)}(x_0^t)\| \\
&= \sum_{k=1}^{n} \|\sum_{j=1}^{k-1}(\nabla f_{\sigma_t(k)}(x_j^t) - \nabla f_{\sigma_t(k)}(x_{j-1}^t))\|
\end{aligned}
$$

$$\begin{aligned}
&\le \sum_{k=1}^{n}\sum_{j=1}^{k-1} \|\nabla f_{\sigma_t(k)}(x_j^t) - \nabla f_{\sigma_t(k)}(x_{j-1}^t)\| \\
&\le \sum_{k=1}^{n}\sum_{j=1}^{k-1} L\|x_j^t - x_{j-1}^t\| \\
&= \sum_{k=1}^{n}\sum_{j=1}^{k-1} L\| - \gamma_t \nabla f_{\sigma_t(k)}(x_{j-1}^t)\| \\
&\le \sum_{k=1}^{n}\sum_{j=1}^{k-1} L\gamma_t G \\
&= \frac{n(n-1)}{2}\gamma_t G L,
\end{aligned}$$

where the first and second inequality is by triangle inequality of vector norm, the third inequality is by definition of L-smooth and the fourth inequality is by definition of G. By this result, we have

$$\mathbb{E}_t[\|R^t\|^2] \le \frac{n^4}{4}{\gamma_t}^2 G^2 L^2. \tag{9}$$

(b)

As for the $\mathbb{E}_t[R^t]$ term, to make a relatively tight bound, we need to use the continuous Hessian. Let H_i denote $H_i(\widehat{x}_0^t)$. Define vector value directional function $dir(v) = \frac{v}{\|v\|}$, $v \neq 0$ and $dir(0) = 0$. Then for $\forall a, b \in \mathbb{R}^d$, and a matrix function $g(\cdot) : \mathbb{R}^d \to \mathbb{R}^{d\times d}$, define line integral:

$$\int_a^b g(x)dx := \int_0^{\|b-a\|} g(a + t\frac{b-a}{\|b-a\|} dir(b-a))dt,$$

where the integral on the left side represents integrating the matrix values function along the line from a to b and that on the right side is integral of a vector valued function over a real number interval. We make a decomposition of R^t:

$$\begin{aligned}
R^t &:= \sum_{k=1}^{n}[\nabla f_{\sigma_t(k)}(x_{k-1}^t) - \nabla f_{\sigma_t(k)}(x_0^t)] \\
&= \sum_{k=1}^{n}[\int_{x_0^t}^{x_{k-1}^t} H_{\sigma_t(k)}(x)dx] \\
&= \sum_{k=1}^{n}[\int_{x_0^t}^{x_{k-1}^t} H_{\sigma_t(k)}dx] + \sum_{k=1}^{n}[\int_{x_0^t}^{x_{k-1}^t} (H_{\sigma_t(k)}(x) - H_{\sigma_t(k)})dx] \\
&= \sum_{k=1}^{n} H_{\sigma_t(k)}(x_{k-1}^t - x_0^t) + \sum_{k=1}^{n}[\int_{x_0^t}^{x_{k-1}^t} (H_{\sigma_t(k)}(x) - H_{\sigma_t(k)})dx] \\
&= \sum_{k=1}^{n}[H_{\sigma_t(k)} \sum_{i=1}^{k-1}(-\gamma_t \nabla f_{\sigma_t(i)}(x_{i-1}^t))] + \sum_{k=1}^{n}[\int_{x_{k-1}^t}^{x_{k-1}^t} (H_{\sigma_t(k)}(x) - H_{\sigma_t(k)})dx] \\
&= -\gamma_t \sum_{k=1}^{n}[H_{\sigma_t(k)} \sum_{i=1}^{k-1} \nabla f_{\sigma_t(i)}(x_0^t)] - \gamma_t \sum_{k=1}^{n}\{H_{\sigma_t(k)} \sum_{i=1}^{k-1}[\nabla f_{\sigma_t(i)}(x_{i-1}^t) - \nabla f_{\sigma_t(i)}(x_0^t)]\}
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{k=1}^{n}[\int_{x_{k-1}^t}^{x_{k-1}^t}(H_{\sigma_t(k)}(x)-H_{\sigma_t(k)})dx] \\
&= A^t + B^t + C^t, \qquad (10)
\end{aligned}
$$

where

$$
\begin{aligned}
A^t &:= -\gamma_t \sum_{k=1}^{n}[H_{\sigma_t(k)} \sum_{i=1}^{k-1} \nabla f_{\sigma_t(i)}(x_0^t)], \\
B^t &:= -\gamma_t \sum_{k=1}^{n}\{H_{\sigma_t(k)} \sum_{i=1}^{k-1}[\nabla f_{\sigma_t(i)}(x_{i-1}^t) - \nabla f_{\sigma_t(i)}(x_0^t)]\}, \\
C^t &:= \sum_{k=1}^{n}[\int_{x_{k-1}^t}^{x_{k-1}^t}(H_{\sigma_t(k)}(x)-H_{\sigma_t(k)})dx].
\end{aligned}
$$

Then we give a bound to each term respectively.

$$
\begin{aligned}
\|B^t\| &\le \gamma_t \sum_{k=1}^{n} H_{\sigma_t(k)} \sum_{i=1}^{k-1} \|\nabla f_{\sigma_t(i)}(x_{i-1}^t) - \nabla f_{\sigma_t(i)}(x_0^t)\| \\
&\le \gamma_t \sum_{k=1}^{n} L \sum_{i=1}^{k-1} (i-1)\gamma_t G L \\
&= \gamma_t^2 L^2 G \sum_{k=1}^{n} \frac{(k-1)(k-2)}{2} \\
&\le \frac{1}{2}\gamma_t^2 L^2 G n^3. \\
\|C^t\| &\le \sum_{k=1}^{n}[\int_{)}^{\|x_{k-1}^t - x_0^t\|} \|H_{\sigma_t(k)}(x_0^t + t\frac{x_{k-1}^t - x_0^t}{\|x_{k-1}^t - x_0^t\|}) - H_{\sigma_t(k)}\|]dt \\
&\le \sum_{i=1}^{n}[L_H \max\{\|x_{k-1}^t - \widehat{x}_0^t\|, \|x_0^t - \widehat{x}_0^t\|\}\|x_{k-1}^t - x_0^t\|] \\
&\le n[(\|x_0^t - \widehat{x}_0^t\| + n\gamma_t G) L_H n \gamma_t G] \\
&= n^2 \gamma_t L_H G \|x_0^t - \widehat{x}_0^t\| + n^3 \gamma_t^2 L_H G^2 \\
\mathbb{E}_t[A^t] &= -\frac{n(n-1)}{2}\gamma_t \mathbb{E}_{i\neq j}[H_i \nabla f_j(x_0^t)].
\end{aligned}
$$

Then (b) can be decomposed as following:

$$
\begin{aligned}
-\widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[R^t]\rangle &= -\widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[A^t] + \mathbb{E}_t[B^t] + \mathbb{E}_t[C^t]\rangle \\
&= -\widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[A^t]\rangle - \widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[B^t]\rangle - \widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[C^t]\rangle \\
&= \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_{i\neq j}[H_i \nabla f_j(x_0^t)]]\rangle - \widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[B^t]\rangle \\
&\quad - \widehat{\rho}\gamma_t\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_t[C^t]\rangle \qquad (11)
\end{aligned}
$$

For the first term, with $f_{i_{1/\widehat{\rho}}}(x) := f_i(x) + \frac{\widehat{\rho}}{2}\|x - x_0^t\|^2$ and (7), we have further bound

$$
\begin{aligned}
&\frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_{i\neq j} H_i \nabla f_j(x_0^t)\rangle \\
&= \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\left[\mathbb{E}_{i\neq j}\langle H_i(x_0^t - \widehat{x}_0^t), \nabla f_{j_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\rangle + \langle x_0^t - \widehat{x}_0^t, \mathbb{E}_{i\neq j} H_i \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\right] \\
&\leq \frac{\widehat{\rho}}{2}\gamma_t^2 n^2 \mathbb{E}_{i,j}\langle \nabla f_{i_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{i_{1/\widehat{\rho}}}(\widehat{x}_0^t), \nabla f_{j_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\rangle + \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)[\frac{\lambda_1}{2}\|x_0^t - \widehat{x}_0^t\|^2 \\
&\quad + \frac{1}{2\lambda_1}\|\Delta\|^2] + \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\mathbb{E}_{i\neq j}\langle H_i(x_0^t - \widehat{x}_0^t) - (\nabla f_{i_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{i_{1/\widehat{\rho}}}(\widehat{x}_0^t)), \nabla f_{j_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\rangle \\
&\leq \frac{\widehat{\rho}}{2}\gamma_t^2 n^2\|\nabla f(x_0^t)\|^2 + \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)[\frac{\lambda_1}{2}\|x_0^t - \widehat{x}_0^t\|^2 + \frac{1}{2\lambda_1}\|\mathbb{E}_{i\neq j} H_i \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\|^2] \\
&\quad + \frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)L[(\frac{L_H G}{\widehat{\rho}} + \widehat{\rho})\|x_0^t - \widehat{x}_0^t\|^2] \qquad (12)
\end{aligned}
$$

The first inequality uses AM-GM inequality with a $\lambda_1 > 0$ to be determined later. And the last inequality is because of the Lipschitz Hessian and the bound (4) in the beginning:

$$
\begin{aligned}
\|H_i(x_0^t - \widehat{x}_0^t) - (\nabla f_{i_{1/\widehat{\rho}}}(x_0^t) - \nabla f_{i_{1/\widehat{\rho}}}(\widehat{x}_0^t))\| &= \left\| H_i(x_0^t - \widehat{x}_0^t) - \int_{\widehat{x}_0^t}^{x_0^t} (H_i(x) + \widehat{\rho})dx \right\| \\
&= \left\| \int_{\widehat{x}_0^t}^{x_0^t} (H_i - H_i(x) - \widehat{\rho})dx \right\| \\
&\leq \int_0^{\|x_0^t - \widehat{x}_0^t\|} \left\| H_i - H_i\left(\widehat{x}_0^t + t\frac{x_0^t - \widehat{x}_0^t}{\|x_0^t - \widehat{x}_0^t\|}\right) - \widehat{\rho} \right\| dt \\
&\leq L_H\|x_0^t - \widehat{x}_0^t\|^2 + \widehat{\rho}\|x_0^t - \widehat{x}_0^t\| \\
&\leq \left(\frac{L_H G}{\widehat{\rho}} + \widehat{\rho}\right)\|x_0^t - \widehat{x}_0^t\|.
\end{aligned}
$$

Then we make a bound for $\|\mathbb{E}_{i\neq j} H_i \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\|$, with i, j uniformly selected from all pairs of different indices. Here shows the important of random sampling.

$$
\begin{aligned}
\|\mathbb{E}_{i\neq j} H_i \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t)\| &= \left\| \frac{1}{n(n-1)} \sum_{i\neq j} H_i \nabla f_{j_{1/\widehat{\rho}}}(\widehat{x}_0^t) \right\| \\
&= \left\| \frac{-1}{n(n-1)} \sum_{i} H_i \nabla f_{i_{1/\widehat{\rho}}}(\widehat{x}_0^t) \right\| \\
&= \frac{1}{n-1}\|\mathbb{E}[H_i \nabla f_i(\widehat{x}_0^t)]\| \\
&\leq \frac{1}{n-1} LG \qquad (13)
\end{aligned}
$$

After combining (12) and (13), for the first term, we get a bound

$$
\frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\langle \widehat{x}_0^t - x_0^t, \mathbb{E}_{i\neq j} H_i \nabla f_j(x_0^t)\rangle \leq \frac{\widehat{\rho}}{2}\gamma_t^2 n^2 \|\nabla f(x_0^t)\|^2
$$

$$+\frac{\widehat{\rho}}{2}\gamma_t^2 n(n-1)\left[\frac{\lambda_1}{2}+L(\frac{L_HG}{\widehat{\rho}}+\widehat{\rho})\right]\|x_0^t-\widehat{x}_0^t\|^2+\frac{\widehat{\rho}\gamma_t^2L^2G^2n}{4\lambda_1(n-1)} \quad (14)$$

For the second term in (11), we bound it using the AM–GM inequality with a $\lambda_2 > 0$

$$\begin{aligned}-\widehat{\rho}\gamma_t\langle\widehat{x}_0^t-x_0^t,\mathbb{E}_t[B^t]\rangle &\leq \widehat{\rho}\gamma_t\left[\frac{\lambda_2}{2}\|x_0^t-\widehat{x}_0^t\|^2+\frac{1}{2\lambda_2}\|\mathbb{E}_t[B^t]\|^2\right]\\ &\leq \widehat{\rho}\gamma_t\left[\frac{\lambda_2}{2}\|x_0^t-\widehat{x}_0^t\|^2+\frac{1}{8\lambda_2}\gamma_t^4L^4G^2n^6\right]. \quad (15)\end{aligned}$$

For the third term in (11), we use another form of AM–GM inequality to bound

$$\begin{aligned}-\widehat{\rho}\gamma_t\langle\widehat{x}_0^t-x_0^t,\mathbb{E}_t[C^t]\rangle &\leq \widehat{\rho}\gamma_t\|x_0^t-\widehat{x}_0^t\|\|\mathbb{E}[C^t]\|\\ &\leq \widehat{\rho}\gamma_t\|x_0^t-\widehat{x}_0^t\|[n^2\gamma_tL_HG\|x_0^t-\widehat{x}_0^t\|+n^3\gamma_t^2L_HG^2]\\ &= \widehat{\rho}\gamma_t^2L_HGn^2\|x_0^t-\widehat{x}_0^t\|^2+\widehat{\rho}\gamma_t^3n^3L_HG^2\|x_0^t-\widehat{x}_0^t\|\\ &\leq \frac{3}{2}\widehat{\rho}\gamma_t^2L_HGn^2\|x_0^t-\widehat{x}_0^t\|^2+\frac{1}{2}\widehat{\rho}\gamma_t^4L_HG^3n^4. \quad (16)\end{aligned}$$

Substituting (14) (15) (16) back to (11), we have

$$\begin{aligned}-\widehat{\rho}\gamma_t\langle\widehat{x}_0^t-x_0^t,\mathbb{E}_t[R^t]\rangle \leq{}& \frac{\widehat{\rho}}{2}\gamma_t^2n^2\|\nabla f(x_0^t)\|^2+\left[\frac{\widehat{\rho}\gamma_t^2L^2G^2n}{4\lambda_1(n-1)}+\frac{1}{8\lambda_2}\gamma_t^5L^4G^2n^6+\frac{1}{2}\widehat{\rho}\gamma_t^4L_HG^3n^4\right]\\ &+\left\{\frac{\widehat{\rho}}{2}\gamma_t^2n(n-1)\left[\frac{\lambda_1}{2}+L(\frac{L_HG}{\widehat{\rho}}+\widehat{\rho})\right]+\frac{\lambda_2}{2}\widehat{\rho}\gamma_t+\frac{3}{2}\widehat{\rho}\gamma_t^2L_HGn^2\right\}\|x_0^t-\widehat{x}_0^t\|^2. \quad (17)\end{aligned}$$

Then we replace (a) (b) (c) in (4) with their bounds (8) (9) (17) to get a recursion bound for one epoch. Let $\lambda_1=\frac{(\widehat{\rho}-\rho)(L+1)}{\widehat{\rho}-\rho+L+1}\gamma_t^{-1}(n-1)^{-1}$ and $\lambda_2=\frac{(\widehat{\rho}-\rho)(L+1)}{2(\widehat{\rho}-\rho+L+1)}n$, we have

$$\begin{aligned}\mathbb{E}_t[e_{1/\widehat{\rho}}f(x_0^{t+1})] \leq{}& e_{1/\widehat{\rho}}f(x_0^t)-(\frac{\widehat{\rho}n\gamma_t}{\widehat{\rho}-\rho+L+1}-\frac{3}{2}\widehat{\rho}{\gamma_t}^2n^2)\|\nabla f(x_0^t)\|^2\\ &-\left\{\frac{\widehat{\rho}(\widehat{\rho}-\rho)(L+1)}{2(\widehat{\rho}-\rho+L+1)}\gamma_tn-\left[\frac{1}{4}\widehat{\rho}L(\frac{L_HG}{\widehat{\rho}}+\widehat{\rho})+\frac{3}{2}\widehat{\rho}L_HG\right]\gamma_t^2n^2\right\}\|x_0^t-\widehat{x}_0^t\|^2\\ &+\frac{\widehat{\rho}L^2G^2(\widehat{\rho}-\rho+L+1)}{4(\widehat{\rho}-\rho)(L+1)}\gamma_t^3n+\frac{L^4G^2(\widehat{\rho}-\rho+L+1)}{4(\widehat{\rho}-\rho)(L+1)}\gamma_t^5n^5+(\frac{1}{2}\widehat{\rho}L_HG^3+\frac{1}{4}\widehat{\rho}L^2G^2)\gamma_t^4n^4 \quad (18)\end{aligned}$$

Now assume $\frac{\widehat{\rho}n\gamma_t}{\widehat{\rho}-\rho+L+1}>\frac{3}{2}\widehat{\rho}{\gamma_t}^2n^2 \Leftrightarrow \gamma_t<\frac{2}{3n(\widehat{\rho}-\rho+L+1)}$ (A4); $\frac{\widehat{\rho}(\widehat{\rho}-\rho)(L+1)}{4(\widehat{\rho}-\rho+L+1)}\gamma_tn>\left[\frac{1}{4}\widehat{\rho}L(\frac{L_HG}{\widehat{\rho}}+\widehat{\rho})+\frac{3}{2}\widehat{\rho}L_HG\right]\gamma_t^2n^2 \Leftrightarrow \gamma_t<\frac{(\widehat{\rho}-\rho)(L+1)}{n(\widehat{\rho}-\rho+L+1)}[L(\frac{L_HG}{\widehat{\rho}}+\widehat{\rho})+6L_HG]^{-1}$ (A5), then inequality (18) can be further turned into:

$$\mathbb{E}_t[e_{1/\widehat{\rho}}f(x_0^{t+1})]\leq e_{1/\widehat{\rho}}f(x_0^t)-C_1\gamma_tnE[\|\widehat{x}_0^t-x_0^t\|^2]+C_2\gamma_t^3n+C_3\gamma_t^5n^5+C_4\gamma_t^4n^4, \quad (19)$$

where $C_1=\frac{\widehat{\rho}(\widehat{\rho}-\rho)(L+1)}{4(\widehat{\rho}-\rho+L+1)}$, $C_2=\frac{\widehat{\rho}L^2G^2(\widehat{\rho}-\rho+L+1)}{4(\widehat{\rho}-\rho)(L+1)}$, $C_3=\frac{L^4G^2(\widehat{\rho}-\rho+L+1)}{4(\widehat{\rho}-\rho)(L+1)}$, $C_4=\frac{1}{2}\widehat{\rho}L_HG^3+\frac{1}{4}\widehat{\rho}L^2G^2$.

Using the law of total expectation to unfold this recursion (19) yields:

$$\mathbb{E}[e_{1/\widehat{\rho}}f(x_0^{T+1})] \le e_{1/\widehat{\rho}}f(x_0^0) - C_1 n \sum_{t=1}^{T} \gamma_t \mathbb{E}[\|\widehat{x}_0^t - x_0^t\|^2] + C_2 n \sum_{t=1}^{T} {\gamma_t}^3 + C_3 n^5 \sum_{t=1}^{T} {\gamma_t}^5 + C_4 n^4 \sum_{t=1}^{T} {\gamma_t}^4.$$

Lower-bounding the left-hand side by $\min f$ and rearranging, we obtain the bound:

$$\frac{1}{\sum_{i=1}^{n} \gamma_t} \sum_{t=0}^{T} \gamma_t \mathbb{E}[\|\widehat{x}_0^t - x_0^t\|^2] \le \frac{(f_{1/\widehat{\rho}}(x_0) - \min f) + C_2 n \sum_{t=1}^{T} {\gamma_t}^3 + C_3 n^5 \sum_{t=1}^{T} {\gamma_t}^5 + C_4 n^4 \sum_{t=1}^{T} {\gamma_t}^4}{C_1 n \sum_{t=1}^{T} \gamma_t} \tag{20}$$

Notice that the left-hand side of (20) is precisely $\mathbb{E}[\|\widehat{x}_0^{T^*} - x_0^{T^*}\|^2]$. So finally, we can complete the proof:

$$\begin{aligned}\mathbb{E}[\|\nabla e_{1/\widehat{\rho}}f(x_0^{T^*})\|^2] &= \widehat{\rho}^2 \mathbb{E}[\|\widehat{x}_0^{T^*} - x_0^{T^*}\|^2] \\ &\le \frac{(f_{1/\widehat{\rho}}(x_0) - \min f) + C_2 n \sum_{t=1}^{T} {\gamma_t}^3 + C_3 n^5 \sum_{t=1}^{T} {\gamma_t}^5 + C_4 n^4 \sum_{t=1}^{T} {\gamma_t}^4}{\widehat{\rho}^{-2} C_1 n \sum_{t=1}^{T} \gamma_t}\end{aligned}$$

□

5.5 Convergence Rate

Fix an index $T > 0$ and set the constant stepsize $\gamma_t \equiv \gamma = \frac{\alpha}{(T+1)^{1/3}}$ for some real $\alpha > 0$ and $\rho = 2L$, $\widehat{\rho} = 4L$. Then the point $x_0^{T^*}$ returned by Algorithm RR satisfies:

$$E[\|\nabla f_{1/4L}(x_0^{T^*})\|^2] \le \frac{(f_{1/(4L)}(x_0) - \min f) + C_2 n + C_3 n^5 T^{-2/3} + C_4 n^4 T^{-1/3}}{(4L)^{-2} C_1 n (T+1)^{2/3}}. \tag{21}$$

The inequality (11) yeilds a result of the form $O(\frac{1}{T^{2/3}} + \frac{n^4}{T} + \frac{n^5}{T^{4/3}})$. What remains to determine is to satisfy the assumptions (A4) and (A5). So $T > O(n^3)$ is needed. This result shows that random reshuffling method achieves an asymptotic convergence rate at $O(T^{-\frac{2}{3}})$ on weakly convex functions.

5.6 Further Discussions

In future researches, it is possible to try some other analysis techniques to improve the n dependence in the result and show a better convergence rate of RR on smooth nonconvex functions with a small number of epochs T.

Besides, we can take a look at the nonsmooth nonconvex setting. The Moreau envelope always has Lipschitz gradients, regardless of the nonsmoothness of the original functions. We have shown a proof of the smooth nonconvex setting. So it is worthwhile for us to use the powerful tool, Moreau envelope, to prove a convergence rate of RR on nonsmooth weakly convex functions and get some better results than that of SGD at $O(k^{-1/4})$ [16].

6 Experiments

In this section, we present some empirical results to study and compare the behaviors of SGD and RR on nonconvex functions.

First, we use SGD and RR to solve some self-made smooth nonconvex minimization problems. An example is to minimize $f(x) := \sum_{i=1}^{n} \left[\ln(1 + \exp(-y_i \boldsymbol{\theta_i}^T x)) + \lambda(x^2 + \epsilon)^{1/4}\right]$, where $y_i \in \mathbb{R}$ and $\boldsymbol{\theta_i} \in \mathbb{R}^d$, which is a logistic regression problem with a nonconvex regularization item. We randomly generate y_i and $\boldsymbol{\theta_i}$ from standard normal distribution. And we set the parameters as $n = 1000$, $d = 40$, $\lambda = 1$ and $\epsilon = 10$. Results in the following Figure 1 shows that RR converges slightly faster than SGD.

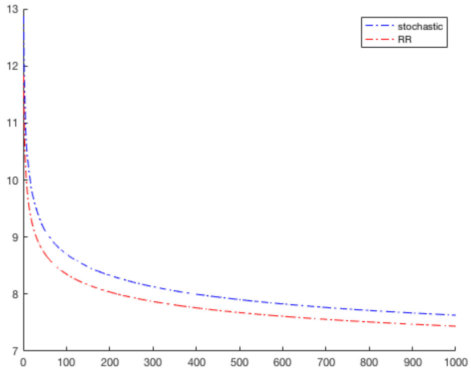

Figure 1

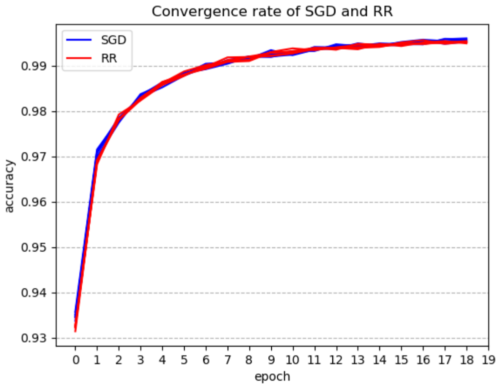

Figure 2

Then we train neural networks on some standard datasets such as CIFAR and MNIST. Here our objective function is the cross-entropy between the ground truth and the prediction vectors, a typical nonsmooth nonconvex function. The average results of SGD and RR for MNIST after five repeats are in the above Figure 2, from which we can learn that they converge almost at the same speed.

During our numerical simulations, we have found that sometimes RR converges a little bit faster than SGD, sometimes they are at the same speed, but there are almost no slower cases for RR, which means with an expectation, RR may perform better than SGD in such nonconvex setting.

The experiments also inspire us to do some further theoretical study on random reshuffling methods in a more complicated setting like nonconvex functions.

Acknowledgments

I would like to express my deepest appreciation to my mentor Professor Wotao Yin from the Department of Mathematics, UCLA, who helped me a lot in doing research, without whose knowledge and assistance this study would not have been successful.

I am also grateful to UCLA CSST program, who gave me the golden opportunity to do this wonderful project and provided me with many supports.

References

[1] Dimitri P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: A survey. *Optimization for Machine Learning*, 2011.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning,*, 2011.

[3] Mert Gürbüzbalaban, Asu Ozdaglar, and Pablo Parrilo. Why random reshuffling beats stochastic gradient descent. *eprint arXiv:1510.08560*, 2015.

[4] L. Bottou, F. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[5] D. Blatt, A. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.

[6] D. Blatt, A. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.

[7] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. 2013.

[8] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. pages 315–323, 2013.

[9] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. pages 1646–1654, 2014.

[10] O. Shamir and T. Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. *International Conference on Machine Learning*, 2013.

[11] Zeyuan Allen-Zhu. Katyusha X: practical momentum method for stochastic sum-of-conconvex optimization. *CoRR*, abs/1802.03866, 2018.

[12] Mert Gürbüzbalaban, Asu Ozdaglar, and Pablo Parrilo. Convergence rate of incremental gradient and newton methods. *eprint arXiv:1510.08562*, 2015.

[13] Jeffery Z. HaoChen and Suvrit Sra. Random shuffling beats SGD after finite epochs. *eprint arXiv:1806.10077*, 2018.

[14] J. Sashank, Ahmed Hefny Reddi, Suvrit Sra, Poczos Barnabas, and Smola Alex. Stochastic variance reduction for nonconvex optimization. *eprint arXiv:1603.06160*, 2016.

[15] A. Nedić and D. Bertsekas. Convergence rate of incremental subgradient algorithms. *Stochastic Optimization: Algorithms and Applications. Applied Optimization, vol 54. Springer, Boston, MA*, 2001.

[16] Damek Davis and Dmitriy Drusvyatskiy. Stochastic mubgradient method converges at the rate $O(k^{-1/4})$ on weakly convex functions. *eprint arXiv:1802.02988*, 2018.

[17] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation rpproach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[18] Bicheng Ying, Kun Yuan, Stefan Vlaski, and Ali H. Sayed. Stochastic learning under random reshuffling. *eprint arXiv:1803.07964*, 2018.

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

[20] Dmitriy Drusvyatskiy, Courtney Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *eprint arXiv:1605.00125*, 2016.

[21] Damek Davis, Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *eprint arXiv:1803.06523*, 2018.

[22] Tim Hoheisel, Maxime Laborde, and Adam Oberman. On proximal point-type algorithms for weakly convex functions and their connection to the backward euler method. *Mathematics Subject Classification*, 2010.

[23] Yurii Nesterov. Introductory lectures on nonvex optimization: A basic course. *Springer US*, 87(1), 2004.