SMG: A Shuffling Gradient-Based Method with Momentum

Trang H. Tran[1] · Lam M. Nguyen[2] · Quoc Tran-Dinh[3]

[1] Cornell University, School of Operations Research and Information Engineering
[2] IBM Research, Thomas J. Watson Research Center [3] The University of North Carolina at Chapel Hill

Problem Statement

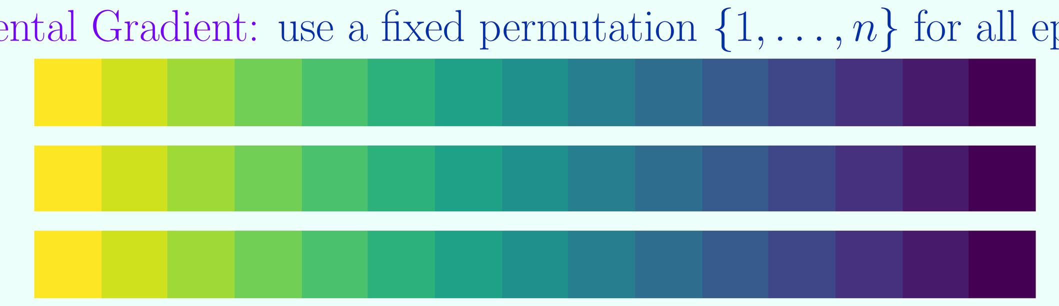
We consider the following finite-sum minimization:

$$\min_{w\in\mathbb{R}^d}\left\{F(w) := \frac{1}{n}\sum_{i=1}^{n} f(w;i)\right\}, \tag{1}$$

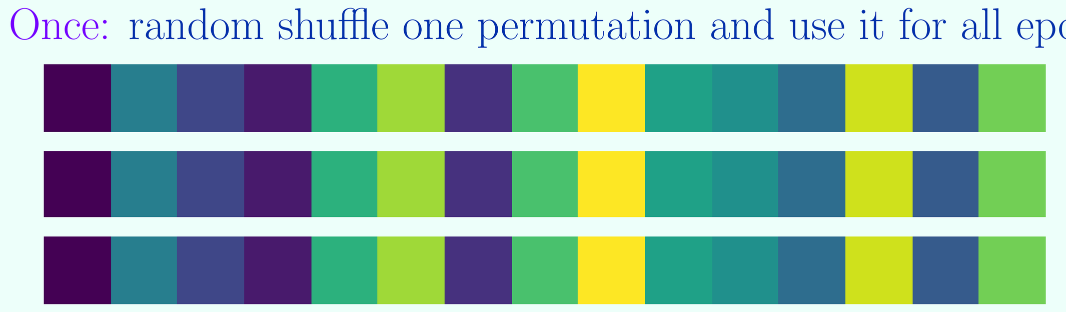
where $f(\cdot;i):\mathbb{R}^d\to\mathbb{R}$ is a given smooth function for $i\in[n]:=\{1,\ldots,n\}$.

Assume that we have access to the first order oracle of $f(\cdot;i)$. Below are some common sampling schemes:

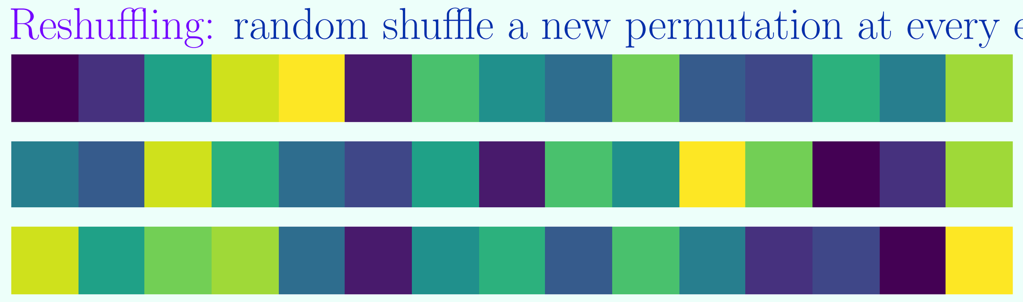
1. **Regular (Standard) Scheme:** Uniformly at random: at each iteration i_t of epoch t, sample an index uniformly at random from $[n]$.
2. **Shuffling Schemes:**
   - Incremental Gradient: use a fixed permutation $\{1,\ldots,n\}$ for all epochs.



   - Shuffle Once: random shuffle one permutation and use it for all epochs.



   - Random Reshuffling: random shuffle a new permutation at every epoch.



Assumptions

Assumption 1. Problem (1) satisfies:

(a) (Boundedness from below) $F_* := \inf_{w\in\mathbb{R}^d} F(w) > -\infty$, and $\mathrm{dom}(F)\neq\emptyset$.

(b) (L-smoothness) $f(\cdot;i)$ is L-smooth for all $i\in[n]$:

$$\|\nabla f(w;i)-\nabla f(w';i)\|\le L\|w-w'\|,\ \text{for all } w,w'\in\mathrm{dom}(F)$$

(c) (Generalized bounded variance) There exist two finite constants $\Theta,\sigma\ge 0$ such that for any $w\in\mathrm{dom}(F)$:

$$\frac{1}{n}\sum_{i=1}^{n}\|\nabla f(w;i)-\nabla F(w)\|^2\le\Theta\|\nabla F(w)\|^2+\sigma^2.$$

Assumption 2. (Bounded gradient) There exists $G>0$ such that $\|\nabla f(x;i)\|\le G$, $\forall x\in\mathrm{dom}(F)$ and $i\in[n]$.

Key References

[1] Nguyen, L. M., Tran-Dinh, Q., Phan, D. T., Nguyen, P. H., and van Dijk, M. A unified convergence analysis for shuffling-type gradient methods. *arXiv preprint arXiv:2002.08246, 2020.*

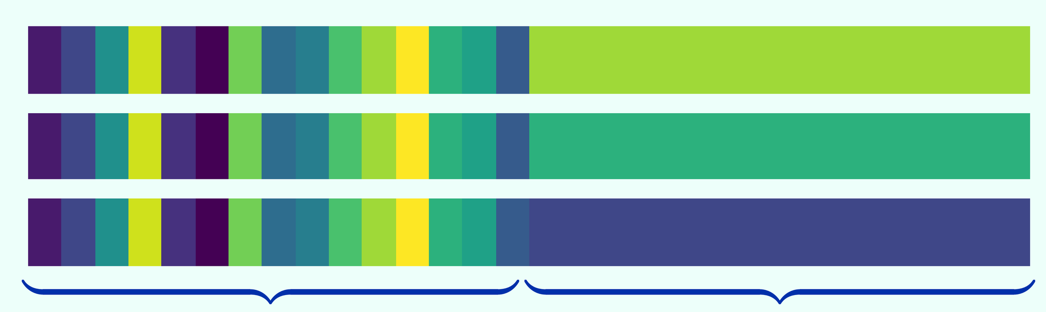
[2] Mishchenko, K., Khaled Ragab Bayoumi, A., and Richtárik P. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems, 33, 2020.*

Shuffling Momentum Gradient (SMG)

Algorithm 1: Shuffling Momentum Gradient (SMG)

1: **Initialization:** Choose $\tilde{w}_0\in\mathbb{R}^d$ and set $\tilde{m}_0 := \mathbf{0}$.
2: **for** $t := 1,2,\cdots,T$ **do**
3: Set $w_0^{(t)} := \tilde{w}_{t-1}$; $m_0^{(t)} := \tilde{m}_{t-1}$; and $v_0^{(t)} := \mathbf{0}$;
4: Generate a deterministic or random permutation $\pi^{(t)}$ of $[n]$;
5: **for** $i := 0,\cdots,n-1$ **do**
6: Query $g_i^{(t)} := \nabla f(w_i^{(t)};\pi^{(t)}(i+1))$;
7: Choose $\eta_i^{(t)} := \frac{\eta_t}{n}$ and update $\begin{cases} m_{i+1}^{(t)} := \beta m_0^{(t)} + (1-\beta)g_i^{(t)} \\ v_{i+1}^{(t)} := v_i^{(t)} + \frac{1}{n}g_i^{(t)} \\ w_{i+1}^{(t)} := w_i^{(t)} - \eta_i^{(t)} m_{i+1}^{(t)}; \end{cases}$
8: **end for**
9: Set $\tilde{w}_t := w_n^{(t)}$ and $\tilde{m}_t := v_n^{(t)}$;
10: **end for**
11: **Output:** Choose $\hat{w}_T\in\{\tilde{w}_0,\cdots,\tilde{w}_{T-1}\}$ at random with probability $\mathbb{P}[\hat{w}_T=\tilde{w}_{t-1}] = \frac{\eta_t}{\sum_{t=1}^{T}\eta_t}$.

- β is a hyperparameter and $\beta=0.5$ works best in our experiments.
- $m_0^{(t)}$ is an average of all the gradients computed at different points in the previous epoch $t-1$.
- Illustration: the update term $m_{i+1}^{(t)}$ in SMG when $\beta=0.5$:



$\beta m_0^{(t)}$ gradients from previous epoch (equally weighted)

$(1-\beta)g_i^{(t)}$ gradient from current epoch

Nonconvex Convergence Results

Theorem 1

Suppose that Assumption 1 holds for (1). Let $\{w_i^{(t)}\}_{t=1}^{T}$ be generated by Algorithm 1 with a fixed momentum weight $0\le\beta<1$ and an epoch learning rate $\eta_i^{(t)} := \frac{\eta_t}{n}$ for every $t\ge 1$. Assume that $\eta_0=\eta_1$, $\eta_t\ge\eta_{t+1}$, and $0<\eta_t\le\frac{1}{L\sqrt{K}}$ for $t\ge 1$, where $K:=\max\left\{\frac{5}{2},\frac{9(5-3\beta)(\Theta+1)}{1-\beta}\right\}$. Then

$$\mathbb{E}\left[\|\nabla F(\hat{w}_T)\|^2\right]\le\frac{4[F(\tilde{w}_0)-F_*]}{(1-\beta)\sum_{t=1}^{T}\eta_t}+\frac{9\sigma^2L^2(5-3\beta)}{(1-\beta)}\left(\frac{\sum_{t=1}^{T}\eta_{t-1}^3}{\sum_{t=1}^{T}\eta_t}\right).$$

- With a constant LR, the convergence rate of SMG is expressed as $\mathcal{O}\left(\frac{[F(\tilde{w}_0)-F_*]+\sigma^2}{T^{2/3}}\right)$, which matches the best known rate in the literature for general shuffling strategies.
- This rate also hold for exponential and cosine scheduled LR schemes, as well as diminishing LR (up to a logarithmic factor).
- With a randomized reshuffling strategy and constant learning rates, the convergence rate of SMG is improved to $\mathcal{O}\left(\frac{[F(\tilde{w}_0)-F_*]+\sigma^2}{n^{1/3}T^{2/3}}\right)$.

Single Shuffling Momentum Gradient

- Replacing the update in Step 7 of SMG by a traditional momentum update $m_{i+1}^{(t)} := \beta m_i^{(t)} + (1-\beta)g_i^{(t)}$, we get Algorithm 2: Single Shuffling Momentum Gradient.
- Illustration: the update term $m_{i+1}^{(t)}$ in Alg 2 when $\beta=0.9$:

$\beta m_i^{(t)}$ gradients from previous iterations (exponentially weighted)

$(1-\beta)g_i^{(t)}$ gradient from current iteration

Theorem 2

Let $\{w_i^{(t)}\}_{t=1}^{T}$ be generated by Algorithm 2, using a single shuffling strategy (IG or SO) with $\eta_i^{(t)} := \frac{\eta_t}{n}$ and $0<\eta_t\le\frac{1}{L}$ for $t\ge 1$. Then, under Assumption 1(a)-(b) and Assumption 2, we have

$$\mathbb{E}\left[\|\nabla F(\hat{w}_T)\|^2\right]\le\frac{\Delta_1}{\left(\sum_{t=1}^{T}\eta_t\right)(1-\beta^n)}+L^2G^2\left(\frac{\sum_{t=1}^{T}\xi_t^3}{\sum_{t=1}^{T}\eta_t}\right)+\frac{4\beta^nG^2}{1-\beta^n},$$

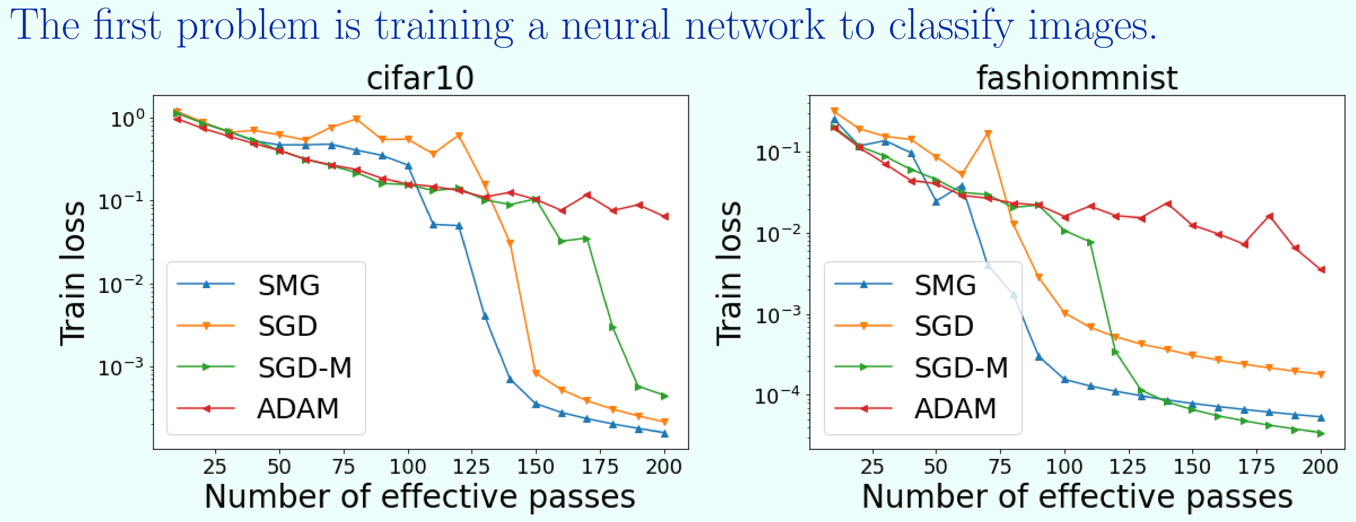
where $\xi_t := \max(\eta_t,\eta_{t-1})$ for $t\ge 2$, $\xi_1=\eta_1$, and

$$\Delta_1 := 2[F(\tilde{w}_0)-F_*]+\left(\frac{1}{L}+\eta_1\right)\|\nabla F(\tilde{w}_0)\|^2+2L\eta_1^2G^2.$$

Applying the previous LR schemes, this theorem leads to the same convergence rate $\mathcal{O}(T^{-2/3})$ for the traditional momentum update.

Experiments

We test SMG method with SGD algorithm, ADAM and SGD with momentum. The first problem is training a neural network to classify images.



For the second experiment, we test four methods on a non-convex logistic regression problem. Our tests have shown encouraging results for SMG.