Final Project of W4995 Optimization of Machine Learning

Ryan Goldenberg Liyi Zhang ChengKuan Chen
Department of Computer Science, Columbia University New York, US

***Abstract*—Stochastic Variance Reduced Gradient (SVRG) is one of the popular method proposed to reduce the variance of gradient in the optimization. In this project, we evaluate its result on the nonconvex optimization. We study the analysis without convex assumption proposed in [1] and try to reproduce their experiment results to verify their claims. However, we can only partially reproduce their result and the advantage seems no longer hold compared to SGD when we use deeper network.**

I. Introduction

The method we study is Stochastic Variance Reduction on Nonconvex Optimization [1]. The main contribution of the paper is that it provide a new theoretical perspective of analyzing SVRG without convex assumption, and the result shows that it has better convergence compared to the Stochastic Gradient Descent (SGD) on noncovex case ($O(n+\frac{n^{\frac{2}{3}}}{\epsilon})$ versus $O(\frac{1}{\epsilon^2})$). In addition, the author also prove a mini-batch version of SVRG and use it to compare the SGD in the experiment. Note that the mini batch version has the same theoretical bound as non-batch version if we the bath is calculated in parallel way while the bound of SGD become $O(\frac{b}{\epsilon^2})$ where b is batch size. Throughout the analysis of nonconvex SVRG, the author rely on following conditions:

- **L-Smoothness**: There is a constant L such that

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

- **ϵ-accuracy**: There is a constant $\epsilon > 0$, we say the algorithm achieve ϵ-accuracy in t if

$$\mathbb{E}[\|\nabla f(x^t)\|^2] \leq \epsilon\ ,$$

  where the expectation is taken over the stochastic of the algorithm

The following pseudocode is from the original paper and we will implement this version in our study since it is the only algorithm that is comparable to the original work.

II. Experiment Setup

Our experiment contains two parts. First, we follow the same experiment setting to test whether the result is reproducible on CIFAR10[1], MNIST[2] and STL[3] dataset. Second, we conduct critical evaluation which include: using deeper network and new FMNIST dataset [2]. The goal in the second part is to see whether the claim can easily generalize to deeper network and other data. Note that although the theoretical analysis suggest compute batch in parallel, we follow the original setting in the paper without computing batch of SVRG in parallel in order to compare SGD fairly.

[1] www.cs.toronto.edu/~kriz/cifar.html
[2] http://yann.lecun.com/exdb/mnist/
[3] https://cs.stanford.edu/~acoates/stl10/

Algorithm 1 Mini-batch SVRG

Input: $\tilde{x}^0 = x_m^0 = x^0 \in \mathbb{R}^d$, epoch length m, step sizes $\{\eta_i > 0\}_{i=0}^{m-1}$, $S = \lceil T/m \rceil$, discrete probability distribution $\{p_i\}_{i=0}^m$, mini-batch size b
for $s = 0$ **to** $S-1$ **do**
  $x_0^{s+1} = x_m^s$
  $g^{s+1} = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\tilde{x}^s)$
  for $t = 0$ **to** $m-1$ **do**
    Choose a mini-batch (uniformly random with replacement) $I_t \subset [n]$ of size b
    $u_t^{s+1} = \frac{1}{b}\sum_{i_t \in I_t}(\nabla f_{i_t}(x_t^{s+1}) - \nabla f_{i_t}(\tilde{x}^s)) + g^{s+1}$
    $x_{t+1}^{s+1} = x_t^{s+1} - \eta_t u_t^{s+1}$
  end for
  $\tilde{x}^{s+1} = \sum_{i=0}^m p_i x_i^{s+1}$
end for
Output: Iterate x_a chosen uniformly random from $\{\{x_t^{s+1}\}_{t=0}^{m-1}\}_{s=0}^{S-1}$.

III. Experiment Result

A. Reproducibility

We use fixed step size of SVRG and SGD with learning rate that give the best performance in terms of training loss in our learning rate tuning experiment and batch size equals to 10. Please refer to implementation details for the learning rate tuning result for each dataset. Our result is shown in the Fig. 1. We only partially reproduce the result in the original paper. Our SVRG performs better on CIFAR10 and only slightly better on MNIST dataset. For all datasets, the gradient norm of SVRG is much smoother and lower compared to SGD which is expected since the main goal of SVRG is reducing variance of gradient norm.

B. Critical evaluation

In this part, we conduct further evaluation of SGD and SVRG by using deeper network on MNIST dataset and on another new FashionMNIST dataset with original network. The details of network structure can be found in implementation details. The result of deeper network on MNIST is shown in Fig 2. We found that the SVRG performance similar to the SGD in terms of training loss. In other word, the advantage of SVRG on MNIST in shallow network could diminish once

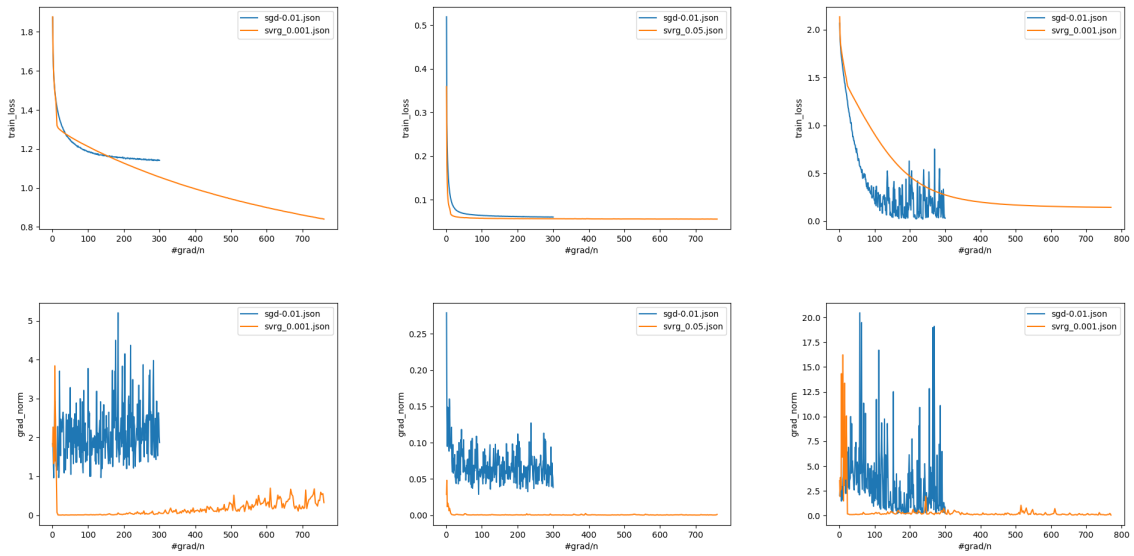

Fig. 1. The comparison of training loss and gradient norm on each dataset. Left: The result on CIFAR10 dataset. Middle: The result on the MNIST dataset. Right: The result on STL dataset. SVRG has lower training loss only on CIFAR10 and MNIST data. For the gradient norm, SVRG is always lower and more stable compared to SGD, which suggest the claim of SVRG.

Dataset	warm-up learning rate	normal learning rate
CIFAR10	0.01	0.001
MNIST	0.03	0.05
STL	0.001	0.001
Deep MNIST	0.03	0.025*
FashionMNIST	0.03	0.005

TABLE I

THE FINAL LEARNING RATE WE USED FOR EACH DATASET. THE NORMAL LEARNING RATE MEANS THE LEARNING PHASE AFTER THE WARM-UP STAGE. NOTE THAT THE * SYMBOL MEANINGS WE DO NOT FIND HUGE DIFFERENCE WHEN LEARNING RATE IN THIS SCALE SO WE JUT ARBITRARY CHOOSE ONE. PLEASE REFER TO FIG 4 FOR MORE DETAILS.

the network become deeper. On the other hand, we found that SVRG performs better than SGD on FashionMNIST dataset. Again, the gradient norm is smoother and smaller compared to the SGD which is expected result of SVRG.

IV. IMPLEMENTATION DETAILS

Dataset We use MNIST, CIFAR10, STL and FashionMNIST in our study. We use default train/test split provide in the dataset to carry on all experiments. All datasets are normalized to [0,1] before feeding into any model.

Learning rate tuning For SVRG, We apply grid search of learning rate in the range [0.5, 0.25, 0.1, 0.03, 0.01, 0.001] for warm-up stage and in the range of [0.5, 0.25, 0.1, 0.05, 0.025, 0.01, 0.005, 0.001] for later stage. The results are shown in Fig. 3 for reproducibility experiment and in Fig. 4 for critical evaluation. The table I shows the summary of final learning rate we pick for each dataset

For SGD, we do not fine-tune the learning rate as mentioned in the original paper due to computation and time constraint. Instead, we use the warm-up stage of SVRG as a proxy of SGD results to find the best learning rate for SGD. We find learning rate = 0.01 for SGD is good enough across all dataset.

Network structure and training details

We follow the original paper to use MLP with one hidden layer and 100 hidden unit for reproducibility experiment and FashionMNIST. All For the experiment of deeper network, we use two hidden layer with 600 and 300 hidden units in each layer. The activate function of hidden unit are relu and softmax in the last layer. All models minimize the cross-entropy loss between the prediction and ground truth label.

The training epoch of SGD is 300 epoch and 250 epochs for SVRG on all datasets. We use the same warmp-up epoch of SVRG and regularization in the original paper. Precisely, the warm-up epoch of SVRG, is 10 epochs for CIFAR10 and MNIST and 20 epochs for STL. The L2-weight decay equals to 1e-3 for MNIST and CIFAR10 and 1e-4 for STL. The setting of using deeper network on MNIST and FashionMNIST experiment is the same as MNIST setting in the original paper.

V. CONCLUSION

In our study, we compare SGD and SVRG in the nonconvex optimization and found that the performance of SVRG does not always better than SGD in terms of training loss. Moreover, we haven't carefully fine-tuned SGD learning rate so there would be a room for SGD to improve. Therefore, to systematically verify the advantage of SVRG, we believe a fine-tuned SGD and averaging experiment results across multiple random seeds and more dataset are necessary.

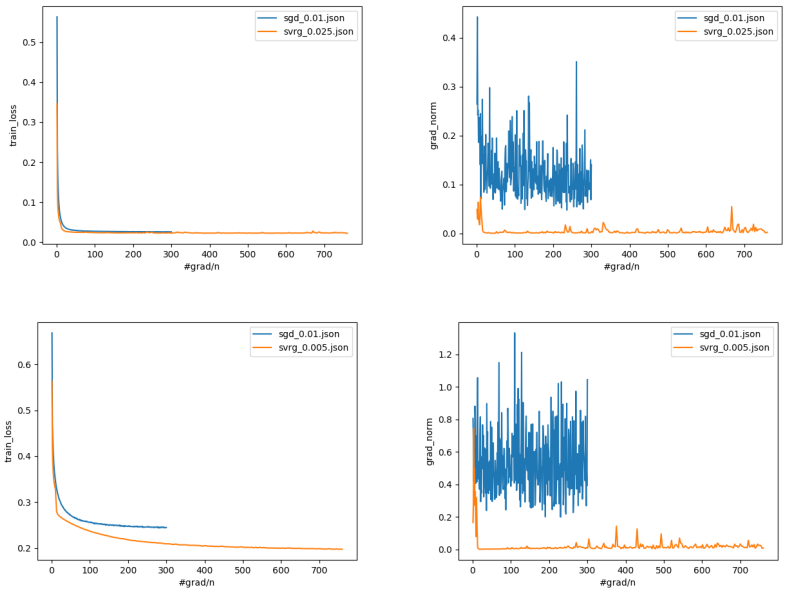

Fig. 2. The SVRG result of training loss and gradient norm. The first row contain the result of MNIST with deeper network and the second row contain the result of FMNIST. The gradient norm of SVRG is always smoother and smaller than SGD especially in FMNIST dataset. Note there are gradient norm values in x-axis but it is pretty close to zero

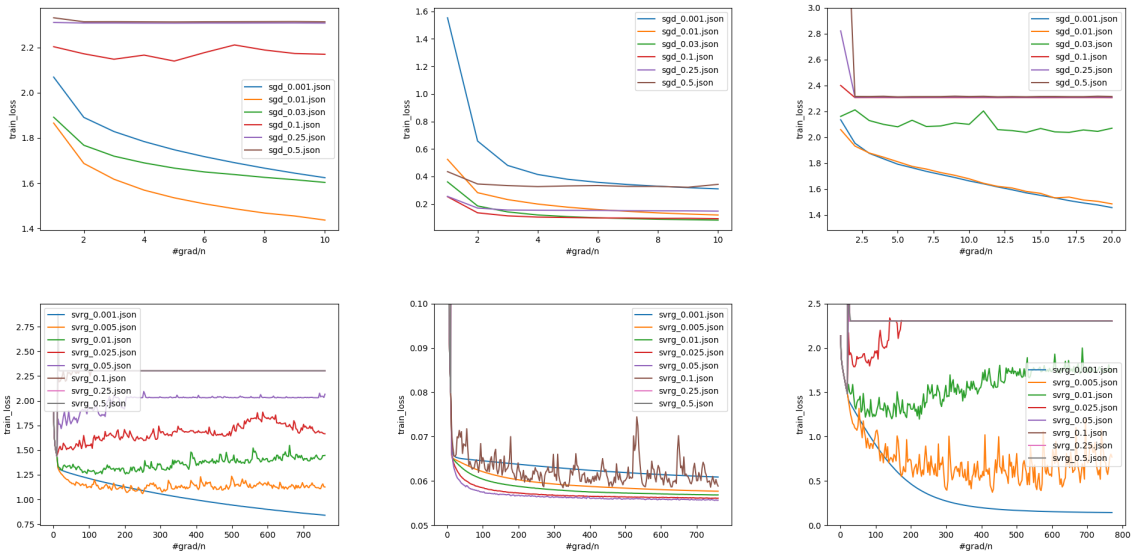

Fig. 3. The training loss under different learning settings for SVRG on CIFAR10, MNIST and STL dataset. The first row contain the result of the warm-up stage and the second row show the result after the warm-up.

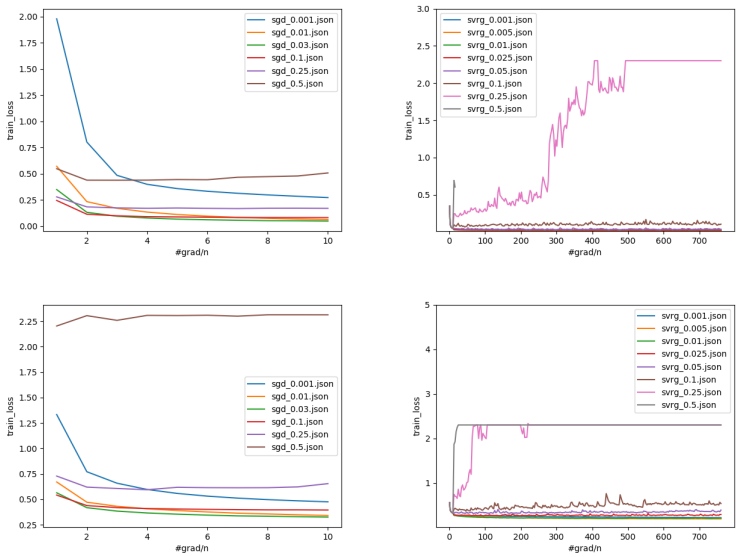

Fig. 4. The training loss under different learning settings for SVRG on MNIST with deeper network and FMNIST. The first row contain the result of deeper network on MSNIT and FMNIST and the second row contain the result after warm-up stage.

REFERENCES

[1] S. J. Reddi, A. Hefny, S. Sra, B. Poczos, and A. Smola, "Stochastic variance reduction for nonconvex optimization," in *International conference on machine learning*, 2016, pp. 314–323.

[2] H. Xiao, K. Rasul, and R. Vollgraf. (2017) Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.