Stability and Generalization of Asynchronous SGD: Sharper Bounds Beyond Lipschitz and Smoothness

Xiaoge Deng, Tao Sun, Shengwei Li, Dongsheng Li, and Xicheng Lu

National Laboratory for Parallel and Distributed Processing (PDL)
College of Computer, National University of Defense Technology (NUDT)
Changsha, Hunan, China

December 2024

Background

- Asynchronous stochastic gradient descent (ASGD) has evolved into an indispensable optimization algorithm for training modern large-scale distributed machine learning tasks.
- Generalizability is an important metric for evaluating machine learning algorithms. Therefore, it is imperative to explore the generalization performance of the ASGD algorithm.
- However, the existing results are either pessimistic and vacuous or restricted by strict assumptions that fail to reveal the intrinsic impact of asynchronous training on generalization.

Stability and Generalization

- Generalization error: expected difference between empirical risk on finite training data and population risk on unknown test examples
- Empirical risk: training dataset $\mathcal{S} = \{z_1, \cdots, z_n\}$

$$F_{\mathcal{S}}(w) = \frac{1}{n}\sum_{i=1}^{n} f(w; z_i)$$

- Population risk: unknown distribution $\mathcal{D}$

$$F(w) = \mathbb{E}_{z\sim\mathcal{D}}[f(w; z)]$$

- Generalization error: $w = A(\mathcal{S})$ denotes the output model obtained by minimizing the empirical risk on $\mathcal{S}$ using the stochastic algorithm A

$$\epsilon_{\mathrm{gen}} = \mathbb{E}_{\mathcal{S},A}\left[F(A(\mathcal{S})) - F_{\mathcal{S}}(A(\mathcal{S}))\right]$$

- Excess generalization error: w^* is the minimizer of F

$$\epsilon_{\mathrm{ex-gen}} = \mathbb{E}_{\mathcal{S},A}\left[F(A(\mathcal{S})) - F(w^*)\right]$$

Stability and Generalization

Stability: measures sensitivity to perturbations in the training dataset

$$\mathcal{S}' = \{z_1', .., z_n'\}, \quad \mathcal{S}^{(i)} = \{z_1, .., z_{i-1}, z_i', z_{i+1}, .., z_n\}.$$

On-average model stability

$$\mathbb{E}_{\mathcal{S},\mathcal{S}',A}\left[\frac{1}{n}\sum_{i=1}^{n}\left\|A(\mathcal{S}) - A(\mathcal{S}^{(i)})\right\|^2\right] \leq \epsilon_{\text{stab}}.$$

Generalization error via on-average model stability

Let $\gamma > 0$. Assume that $f(w;z)$ is non-negative and β-smooth, then

$$\mathbb{E}_{\mathcal{S},A}\left[F(A(\mathcal{S})) - F_{\mathcal{S}}(A(\mathcal{S}))\right] \leq \frac{\beta}{\gamma}\mathbb{E}_{\mathcal{S},A}[F_{\mathcal{S}}(A(\mathcal{S}))] + \frac{\beta+\gamma}{2}\epsilon_{\text{stab}}.$$

If $f(w;z)$ is non-negative, convex, and $\nabla f(w;z)$ is (α,β)-Hölder, then

$$\mathbb{E}_{\mathcal{S},A}\left[F(A(\mathcal{S})) - F_{\mathcal{S}}(A(\mathcal{S}))\right] \leq \frac{c_{\alpha,\beta}^2}{2\gamma}\mathbb{E}_{\mathcal{S},A}[F^{\frac{2\alpha}{1+\alpha}}(A(\mathcal{S}))] + \frac{\gamma}{2}\epsilon_{\text{stab}}.$$

Asynchronous SGD $w_{k+1} = w_k - \eta_k \nabla f(w_{k-\tau_k}; z_{i_k})$

Algorithm 1 Asynchronous SGD

Initialization: model parameter w
Input: learning rate η

```
// Worker m
repeat
    pull the current model w from the server
    compute gradient g^m = ∇f(w; z) with local data z
    push g^m to the server
until terminated
// Server
if server received gradient from any worker m then
    update the model as w ← w − ηg^m
    send w back to worker m
end if
```

Output: model w

Theoretical Analysis

Assumptions:

- The loss function is is non-negative and convex
- The parameter space is a bounded convex set.

Lemma: Smooth case (β-smooth)

$$\left\| w_k - \eta_k \nabla f(w_{k-\tau_k}; z_{i_k}) - \left(w_k^{(i)} - \eta_k \nabla f(w_{k-\tau_k}^{(i)}; z_{i_k}) \right) \right\|^2$$

$$\leq \left\| w_k - w_k^{(i)} \right\|^2 + 2\eta_k \beta^2 r^2 \sum_{j=1}^{\tau_k} \eta_{k-j}.$$

Lemma: Non-Smooth case ((α, β)-Hölder continuous gradient)

$$\left\| w_k - \eta_k \nabla f(w_{k-\tau_k}; z_{i_k}) - \left(w_k^{(i)} - \eta_k \nabla f(w_{k-\tau_k}^{(i)}; z_{i_k}) \right) \right\|^2$$

$$= \| w_k - w_k^{(i)} \|^2 + \mathcal{O}(\eta_k \sum_{j=1}^{\tau_k} \eta_{k-j} + \eta_k^{\frac{2}{1-\alpha}}).$$

Stability and Generalization Bounds: Smooth Case

On-average model stability (non-increasing learning rate $\eta_k \leq 1/2\beta$)

$$\epsilon_{\mathrm{stab}} = \frac{16\beta \mathrm{e}(1+k/n)}{n}\Big[\eta_1\|w_1 - w^*\|^2 + \big(4\beta r^2 + 2F(w^*)\big)\sum_{l=1}^{k}\eta_l^2\Big] + 2\beta^2 r^2 \mathrm{e}\sum_{l=1}^{k}\eta_l\sum_{j=1}^{\tau_l}\eta_{l-j}$$

Generalization error bounds ($F(w^*)=0$, $K \asymp n$, $\eta_k = c(\overline{\tau}\sqrt{K})^{-1}$)

generalization error $\quad \mathbb{E}\left[F(w_K) - F_{\mathcal{S}}(w_K)\right] = \mathcal{O}\Big(\frac{1}{\overline{\tau}} + \frac{1}{\sqrt{K}}\Big)$

excess generalization error $\quad \mathbb{E}\left[F(\overline{w}_K) - F(w^*)\right] = \mathcal{O}\Big(\frac{1}{\overline{\tau}} + \frac{\|w_1 - w^*\|^2}{n}\Big)$

Stability and Generalization Bounds: Non-smooth Case

On-average model stability

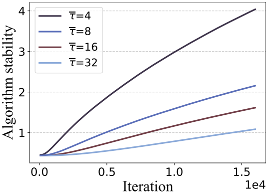
$$\epsilon_{\mathrm{stab}} = \mathcal{O}\left(\frac{1+k/n}{n}\sum_{l=1}^{k}\eta_l^2\mathbb{E}_{\mathcal{S},A}\left[F_{\mathcal{S}}^{\frac{2\alpha}{1+\alpha}}(w_{l-\tau_l})\right]+\sum_{l=1}^{k}\eta_l\sum_{j=1}^{\tau_l}\eta_{l-j}+\sum_{l=1}^{k}\eta_l^{\frac{2}{1-\alpha}}\right)$$

Excess generalization error $(F(w^*)=0,\ K\asymp n,\ \eta_k=c(\overline{\tau}\sqrt{K})^{-1})$

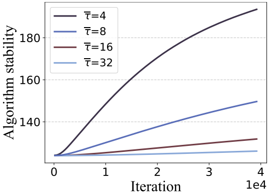
$$\mathbb{E}_{\mathcal{S},A}\left[F(\overline{w}_K)-F(w^*)\right]=\mathcal{O}\left(\frac{1}{\sqrt{\overline{\tau}}}+\frac{\|w_1-w^*\|^{\frac{4\alpha}{1+\alpha}}}{\sqrt{n}^{1+\alpha}}\right)$$

- Sharper and non-vacuous generalization bounds
- Appropriately increasing the asynchronous delay can improve the generalization performance of ASGD

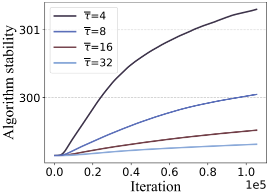
Experiment



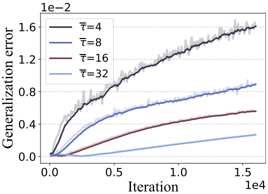
(a) Convex task on RCV1



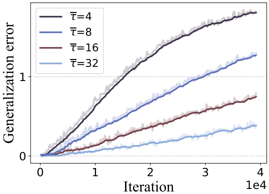
(b) CV task on CIFAR100



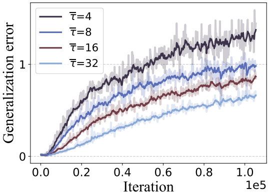
(c) NLP task on SST-2



(d) Convex task on RCV1



(e) CV task on CIFAR100



(f) NLP task on SST-2

Fig: Stability and generalization of ASGD in training various machine learning tasks with learning rate $\eta_k = 0.1/\overline{\tau}$.

Conclusion

- Increasing the asynchronous delay can enhance the stability of the ASGD algorithm at an appropriate learning rate, thereby reducing its generalization error.
- Our generalization results are non-vacuous and applicable to the general convex case.
- The theoretical results in this paper are applicable to non-smooth settings.
- The asynchronous generalization properties of this paper are applicable to the fixed learning rate setting.
- The asynchronous generalization properties of this paper are applicable to non-convex settings.

- Future work: exploring tighter generalization error bounds of ASGD in non-convex settings.

Thanks!