Towards Understanding Why Lookahead Generalizes Better Than SGD and Beyond

Pan Zhou[*], Hanshu Yan[*], Xiao-Tong Yuan[†], Jiashi Feng[*], Shuicheng Yan[*]
[*] Sea AI Lab, Singapore [†] Nanjing University of Information Science & Technology, China
{zhoupan, yanhanshu, fengjs, yansc}@sea.com xtyuan@nuist.edu.cn

homepage

Problem Setup

```
Algorithm 1: SGD
Input  :Objective $F_{\mathcal{S}}(\boldsymbol{\theta})$, dataset $\mathcal{S}$, inner-loop optimizer $\mathcal{A}$,
        inner-loop step number $k$ and learning rate $\{\{\eta_\tau^{(t)}\}$,
        outer-loop learning rate $\alpha \in (0,1)$.
for $t = 1, 2, ..., T$ do
    $\boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;                Inner-loop optimization
    for $\tau = 1, 2, ..., k$ do
        $\boldsymbol{v}_\tau^{(t)} = \mathcal{A}(F_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{v}_{\tau-1}^{(t)}, \eta_{\tau-1}^{(t)}, \mathcal{S}) = \boldsymbol{v}_{\tau-1}^{(t)} - \eta_{\tau-1}^{(t)} \boldsymbol{g}_{\tau-1}^{(t)}$
    end
    $\boldsymbol{\theta}_t = \boldsymbol{v}_k^{(t)} = (1-1)\boldsymbol{\theta}_{t-1} + 1 * \boldsymbol{v}_k^{(t)}$ $(\alpha = 1)$
end                                              outer-loop optimization
Output :$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$
```

```
Algorithm 2: Lookahead
Input  :Objective $F_{\mathcal{S}}(\boldsymbol{\theta})$, dataset $\mathcal{S}$, inner-loop optimizer $\mathcal{A}$,
        inner-loop step number $k$ and learning rate $\{\{\eta_\tau^{(t)}\}$,
        outer-loop learning rate $\alpha \in (0,1)$.
for $t = 1, 2, ..., T$ do
    $\boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;                Inner-loop optimization
    for $\tau = 1, 2, ..., k$ do
        $\boldsymbol{v}_\tau^{(t)} = \mathcal{A}(F_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{v}_{\tau-1}^{(t)}, \eta_{\tau-1}^{(t)}, \mathcal{S}) = \boldsymbol{v}_{\tau-1}^{(t)} - \eta_{\tau-1}^{(t)} \boldsymbol{g}_{\tau-1}^{(t)}$
    end
    $\boldsymbol{\theta}_t = (1-\alpha)\boldsymbol{\theta}_{t-1} + \alpha \boldsymbol{v}_k^{(t)}$.
end                                              outer-loop optimization
Output :$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$
```

Key steps in SGD & LookAhead (LA):

- 1) **inner-loop optimization**: K steps forward in SGD & LA
- 2) **outer-loop optimization**: 1 step back in LA, while no step back in SGD

OPTIMIZER	CIFAR-10	CIFAR-100
SGD	$95.23 \pm .19$	$78.24 \pm .18$
POLYAK	$95.26 \pm .04$	$77.99 \pm .42$
ADAM	$94.84 \pm .16$	$76.88 \pm .39$
LOOKAHEAD	$95.27 \pm .06$	$78.34 \pm .05$

ResNet 18

OPTIMIZER	TRAIN	VAL.	TEST
SGD	43.62	66.0	63.90
LA(SGD)	35.02	65.10	63.04
ADAM	33.54	61.64	59.33
LA(ADAM)	**31.92**	**60.28**	**57.72**
POLYAK	-	61.18	58.79

LSTM

Important observations: LookAhead (LA) enjoys better test performance than SGD

Problem:

- 1) Why LA enjoys better test performance than SGD?
- 2) How to further improve LA?

Tools for Test Performance Analysis

Optimal solution to empirical risk on dataset $\mathcal{S}$:

$$\boldsymbol{\theta}_{\mathcal{S}}^* \in \operatorname{argmin}_{\boldsymbol{\theta}} F_{\mathcal{S}}(\boldsymbol{\theta}) \triangleq \frac{1}{n}\sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i; \boldsymbol{\theta}), \boldsymbol{y}_i),$$

Approximate solution to empirical risk when using algorithm $\mathcal{A}$ on dataset $\mathcal{S}$:

$$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} \approx \operatorname*{argmin}_{\boldsymbol{\theta}} F_{\mathcal{S}}(\boldsymbol{\theta}) \triangleq \frac{1}{n}\sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i; \boldsymbol{\theta}), \boldsymbol{y}_i)$$

Excess risk error to measure test performance:

$$\varepsilon_{\text{exc}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\textit{test error}} - \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\textit{best training error}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\textbf{generalization error}} + \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\textbf{optimization error}}$$

where $F(\boldsymbol{\theta}) \triangleq \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}[\ell(f(\boldsymbol{x}; \boldsymbol{\theta}), \boldsymbol{y})]$ is the population risk.

Necessary definitions:

λ-strongly convex: $\forall \theta_1, \theta_2, f(\theta_1) \geq f(\theta_2) + \langle \nabla f(\theta_2), \theta_1 - \theta_2 \rangle + \frac{\lambda}{2}\|\theta_1 - \theta_2\|^2$

convex: $\forall \theta_1, \theta_2, f(\theta_1) \geq f(\theta_2) + \langle \nabla f(\theta_2), \theta_1 - \theta_2 \rangle$

G-Lipschitz continuous: $\|f(\theta_1) - f(\theta_2)\|_2 \leq G\|\theta_1 - \theta_2\|_2$

L-smooth: $\|\nabla f(\theta_1) - \nabla f(\theta_2)\|_2 \leq L\|\theta_1 - \theta_2\|_2$ $(\forall \theta_1, \theta_2)$

Polyak-Łojasiewicz (PŁ) Condition: $2\mu(f(\theta) - f(\theta^*)) \leq \|\nabla f(\theta)\|^2$ $(\forall \theta)$ where $\theta^* \in \operatorname{argmin}_\theta f(\theta)$.

Weakly Quasi-Convexity: $\langle \nabla f(\theta), \theta - \theta^* \rangle \geq \rho(f(\theta) - f(\theta^*))$

Main Results

Excess risk error to measure test performance:

$$\varepsilon_{\text{exc}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\textit{test error}} - \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\textit{best training error}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\textbf{generalization error}} + \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\textbf{optimization error}}$$

Convex problems. Under proper assumptions, by setting conventional learning rate $\eta = \frac{1}{\sqrt{kT}}$, on convex problem we have

$$\text{optimization error} \leq \mathcal{O}\left(\frac{1}{\alpha\sqrt{kT}}\right), \qquad \text{generalization error} \leq \mathcal{O}\left(\frac{\alpha\sqrt{kT}}{n}\right)$$

where kT is total training iteration number, n is training sample number.

Remark: Since (1) optimum of α is $\alpha = \mathcal{O}\left(1 \cap \sqrt{n/kT}\right)$ and (2) SGD = LA with $\alpha = 1$

Lookahead enjoys smaller excess risk error (test error) than SGD

λ-strongly convex problems. Under proper assumptions, by setting conventional learning rate $\eta = \frac{1}{\sqrt{kT}}$, on convex problem we have

$$\text{optimization error} \leq \begin{cases} O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2 (kT)^{2\alpha}(1-2\alpha)}\right), & 0 < \alpha < \frac{1}{2}, \\ O\left(\frac{1}{T} + \frac{\log(Tk)}{\lambda^2 kT}\right), & \alpha = \frac{1}{2}, \\ O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2(2\alpha-1)kT}\right), & \frac{1}{2} < \alpha \leq 1. \end{cases},$$

$$\text{generalization error} \leq \mathcal{O}\left(\frac{G^2}{n\lambda}\frac{(Tk+1)^\alpha - 1}{((T+1)k+2)^\alpha}\right).$$

When problem is large-scale and iteration number T is not large,

$$\frac{\ln T}{T^\alpha} > \mathcal{O}\left(\frac{1}{n\lambda}\frac{\ln(Tk)}{k^\alpha}\right) \quad \text{and} \quad \frac{1}{\lambda(\alpha-1)^2 Tk} > \mathcal{O}\left(\frac{1}{n}\frac{\ln(Tk)}{T^\alpha k^\alpha}\right)$$

the optimum α is not 1.

Remark: This explains **smaller excess risk error of LA over SGD.**

Nonconvex problems under μ-PL condition. Under proper assumptions, we have

$$\text{optimization error} \leq \mathcal{O}\left(\frac{1}{(Tk+1)^{2\alpha}} + \frac{2\alpha L G^2(\alpha + 2(1-\alpha)(k-1))}{\mu^2 (Tk+1)^{2\alpha-1}}\right),$$

$$\text{generalization error} \leq \mathcal{O}\left(\frac{\xi}{n-1}\alpha^{\frac{1}{1+\gamma}}(Tk)^{\frac{\gamma}{\gamma+1}}\right).$$

where $\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu}$ and $\xi = \ell_{\max}^{\frac{\gamma}{1+\gamma}}\left[\frac{2G^2}{\mu}\right]^{\frac{1}{1+\gamma}}$ in which $\ell_{\max} = \max_{\boldsymbol{\theta},(\boldsymbol{x},\boldsymbol{y})} \ell(f(\boldsymbol{x}; \boldsymbol{\theta}), \boldsymbol{y})$.

Remark: with properly α, **lookahead may have smaller test error than SGD**

An Improved LookAhead: Stagewise Locally-regularized Lookahead

```
Algorithm 3: Stagewise Locally-Regularized LookAhead (SLRLA)
Input  :Loss $F_{\mathcal{S}}(\boldsymbol{\theta})$, constant $\{\beta_q\}_{q=1}^{Q}$
for $q = 1, 2, ..., Q$ do
    $F_q(\boldsymbol{\theta}) = F_{\mathcal{S}}(\boldsymbol{\theta}) + \frac{\beta_q}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_{q-1}\|^2$;
    $\boldsymbol{\theta}_q = \text{Look-ahead}(F_q(\boldsymbol{\theta}), \eta_q, T_q, \alpha_q, k_q, \boldsymbol{\theta}_{q-1}, \mathcal{A}, \mathcal{S})$.
end                                          Vanilla LookAhead
Output :$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_Q$.
```

An Improved LookAhead: Stagewise Locally-regularized Lookahead

Strategy: divide optimization into several stages and use lookahead to solve locally-regularized loss

$$F_q(\boldsymbol{\theta}) = F_{\mathcal{S}}(\boldsymbol{\theta}) + \frac{\beta_q}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_{q-1}\|_2^2$$

Advantages:

- Local regularization improves loss convexity, e.g. ill-conditioned loss —¿ well-conditioned one
- Local regularization helps avoid overfitting

Optimization error. Under proper assumptions, to obtain optimization error

$$\text{optimization error} \leq \epsilon,$$

the stochastic gradient complexity (stochastic gradient evaluation number, a.k.a. IFO) is

stochastic gradient complexity	LookAhead (LA) $\alpha \in (0, \frac{1}{2})$	 $\alpha = \frac{1}{2}$	 $\alpha \in (\frac{1}{2}, 1]$	SLRLA $\alpha \in (0, 1]$
λ-strongly-convex problems	$\mathcal{O}\Big((\frac{1}{\epsilon})^{\frac{1}{2\alpha}} + (\frac{1}{(1-2\alpha)\lambda^2\epsilon})^{\frac{1}{2\alpha}}\Big)$	$\mathcal{O}\Big(\frac{\log\frac{1}{\epsilon}}{\lambda^2\epsilon}\Big)$	$\mathcal{O}\Big(\frac{1}{(2\alpha-1)\lambda^2\epsilon}\Big)$	$\mathcal{O}(\frac{1}{\lambda\alpha\epsilon})$
nonconvex problems with μ-PL	$\mathcal{O}\big((\frac{1}{\mu^2\epsilon})^{1/\alpha}\big)$			$\mathcal{O}(\frac{1}{\mu\alpha\epsilon})$

Remark: By observing factors α, λ and μ, SLRLA has smaller computational complexity than LA, meaning

SLRLA has smaller optimization error than LA under a given computational budget

Generalization error. Under proper assumptions, to obtain optimization error

$$\text{optimization error} \leq \epsilon,$$

the generalization error is

generalization error	LookAhead (LA) $\alpha \in (0, 1]$	SLRLA $\alpha \in (0, 1]$
λ-strongly-convex problems	$\mathcal{O}(\frac{1}{n\lambda})$	$\mathcal{O}(\frac{1}{n(\beta/\alpha+\lambda)})$
nonconvex problems with μ-PL	$\mathcal{O}(\frac{1}{n}(Tk)^{\frac{\gamma}{\gamma+1}})$ $(\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu})$	$\mathcal{O}(\frac{1}{n}/(\frac{c}{\alpha}+\mu))$ $(c \geq 0)$

Remark:

SLRLA has smaller generalization error than LA

Experiments

Table 3: Classification accuracy (%). $^\diamond$, $*$, $\dagger$, $\ddagger$ are respectively reported in [1], [15], [49], [50].

optimizer	CIFAR10 ResNet18	 VGG16	 WRN-16-10	CIFAR100 ResNet18	 VGG16	 WRN-16-10	ImageNet ResNet18
Adam [11]	$94.84^\diamond$	91.08	93.54	$76.88^\diamond$	64.07	74.81	66.54^*
Adabound [51]	92.56	91.35	91.68	71.43	64.74	71.64	$68.13^\dagger$
RAdam [15]	93.85	90.84	94.16	74.30	63.99	75.92	67.62^*
AdamW [52]	94.95	90.75	95.95	77.30	63.40	79.63	$67.93^\dagger$
AdaBelief [50]	$95.20^\ddagger$	92.25	95.71	$77.02^\ddagger$	68.63	77.93	$70.08^\ddagger$
Stagewise SGD [13]	$95.23\pm0.19^\diamond$	92.13 ± 0.02	95.51 ± 0.02	$78.24\pm0.18^\diamond$	69.97 ± 0.02	78.95 ± 0.03	$70.23^\dagger$
SLA [1]	$95.27\pm0.06^\diamond$	92.38 ± 0.02	95.73 ± 0.02	$78.34\pm0.05^\diamond$	70.20 ± 0.04	79.54 ± 0.02	70.30 ± 0.09
SLRLA	**95.47**±0.20	**92.63**±0.03	**96.08**±0.07	**78.58**±0.15	**70.63**±0.02	**79.85**±0.05	**70.47**±0.12

Remark: **SLRLA has better test performance than stagewise LA (SLA)**