Towards Understanding Why LookAhead Generalizes Better Than SGD and Beyond

Pan Zhou

Joint work with Hanshu Yan, Xiaotong Yuan, Jiashi Feng, and Shuicheng Yan

SEA AI Lab

zhoupan@sea.com

Dec 06, 2021

Background: LookAhead

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 : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

[1] Lookahead Optimizer: k steps forward, 1 step back, NeurIPS'19

Background: LookAhead

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**

$\quad \boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;

Inner-loop optimization

$\quad$ **for** $\tau = 1, 2, ..., k$ **do**

$\quad\quad \boldsymbol{v}_\tau^{(t)} = \mathcal{A}(F_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{v}_{\tau-1}^{(t)}, \eta_{\tau-1}^{(t)}, \mathcal{S}) = \boldsymbol{v}_{\tau-1}^{(t)} - \eta_{\tau-1}^{(t)} \boldsymbol{g}_{\tau-1}^{(t)}$

$\quad$ **end**

$\quad \boldsymbol{\theta}_t = \boldsymbol{v}_k^{(t)} = (1-1)\boldsymbol{\theta}_{t-1} + 1 * \boldsymbol{v}_k^{(t)}$ $(\alpha = 1)$

end

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**

$\quad \boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;

Inner-loop optimization

$\quad$ **for** $\tau = 1, 2, ..., k$ **do**

$\quad\quad \boldsymbol{v}_\tau^{(t)} = \mathcal{A}(F_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{v}_{\tau-1}^{(t)}, \eta_{\tau-1}^{(t)}, \mathcal{S}) = \boldsymbol{v}_{\tau-1}^{(t)} - \eta_{\tau-1}^{(t)} \boldsymbol{g}_{\tau-1}^{(t)}$

$\quad$ **end**

$\quad \boldsymbol{\theta}_t = (1-\alpha)\boldsymbol{\theta}_{t-1} + \alpha \boldsymbol{v}_k^{(t)}$.

end

Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

Key steps in SGD & LookAhead (LA):

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

[1] Lookahead Optimizer: k steps forward, 1 step back, NeurIPS'19

Background: LookAhead

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**
  $\boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;
  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
  $\boxed{\boldsymbol{\theta}_t = \boldsymbol{v}_k^{(t)} = (1-1)\boldsymbol{\theta}_{t-1} + 1 * \boldsymbol{v}_k^{(t)} \ (\alpha = 1)}$
end
Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

outer-loop optimization

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**
  $\boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;
  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
  $\boxed{\boldsymbol{\theta}_t = (1-\alpha)\boldsymbol{\theta}_{t-1} + \alpha \boldsymbol{v}_k^{(t)}.}$
end
Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

outer-loop optimization

Key steps in SGD & LookAhead (LA):

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

[1] Lookahead Optimizer: k steps forward, 1 step back, NeurIPS'19

Observation: LookAhead Generalizes Better Than SGD

Important observations: **LookAhead (LA) [1] enjoys better test performance than 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 [1]

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 [1]

[1] Lookahead Optimizer: k steps forward, 1 step back, NeurIPS'19

Observation: LookAhead Generalizes Better Than SGD

Important observations: **LookAhead (LA) [1] enjoys better test performance than 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 [1]

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 [1]

Problems:

1. **Why LA enjoys better test performance than SGD?**
2. **How to further improve LA?**

[1] Lookahead Optimizer: k steps forward, 1 step back, NeurIPS'19

Tools for Test Performance Analysis

- **optimal solution** to empirical risk:

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

- **approximate solution** to empirical risk:

$$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} \approx \operatorname{argmin}_{\boldsymbol{\theta}} F_{\mathcal{S}}(\boldsymbol{\theta}) \triangleq \tfrac{1}{n} \sum_{i=1}^{n} \ell(f(\boldsymbol{x}_i; \boldsymbol{\theta}), \boldsymbol{y}_i), \tag{2}$$

Tools for Test Performance Analysis

- **optimal solution** to empirical risk:

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

- **approximate solution** to empirical risk:

$$\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} \approx \operatorname{argmin}_{\boldsymbol{\theta}} F_{\mathcal{S}}(\boldsymbol{\theta}) \triangleq \tfrac{1}{n}\sum\nolimits_{i=1}^{n} \ell(f(\boldsymbol{x}_i;\boldsymbol{\theta}), \boldsymbol{y}_i), \quad (2)$$

- **Excess risk error** to measure **test performance**:

$$\underbrace{\varepsilon_{\text{exc}}}_{\text{excess risk error}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\text{test error}} - \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\text{best training error}} = \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}})]}_{\text{generalization error}} + \underbrace{\mathbb{E}_{\mathcal{A},\mathcal{S}}[F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}}) - F_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S}}^*)]}_{\text{optimization error}} \quad (3)$$

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

Superiority of LA: Smaller Optimization & Generalization Errors

Excess risk error to measure test performance:

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

Theorem 1 (informal) Under proper assumptions, by setting conventional learning rate $\eta = 1/\sqrt{kT}$, on **convex problem** we have

$$\text{optimization error} \leq O\left(\frac{1}{\alpha\sqrt{kT}}\right) \qquad \text{generalization error} \leq O\left(\frac{\alpha\sqrt{kT}}{n}\right) \tag{5}$$

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

Superiority of LA: Smaller Optimization & Generalization Errors

Excess risk error to measure test performance:

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

Theorem 1 (informal) Under proper assumptions, by setting conventional learning rate $\eta = 1/\sqrt{kT}$, on **convex problem** we have

$$\text{optimization error} \leq O(\frac{1}{\alpha\sqrt{kT}}) \qquad \text{generalization error} \leq O(\frac{\alpha\sqrt{kT}}{n}) \tag{5}$$

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

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

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

Superiority of LA: Smaller Optimization & Generalization Errors

Theorem 2 (informal). $\lambda-$**strongly-convex problem** with proper assumptions:

$$\text{optimization error} \leq \begin{cases} O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2 (kT)^{2\alpha}(1-2\alpha)}\right), & 0<\alpha<\frac{1}{2}, \\ O\left(\frac{1}{T} + \frac{\log(Tk)}{\lambda^2 kT}\right), & \alpha = \frac{1}{2}, \\ O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2 (2\alpha-1)kT}\right), & \frac{1}{2}<\alpha\leq 1. \end{cases} \quad \text{generalization error} \leq O\left(\frac{1}{n\lambda}\frac{(Tk+1)^{\alpha}-1}{((T+1)k+2)^{\alpha}}\right) \tag{6}$$

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

$$\frac{\ln T}{T^{\alpha}} > O\left(\frac{1}{n\lambda}\frac{\ln(Tk)}{k^{\alpha}}\right) \quad \text{or} \quad \frac{1}{\lambda(\alpha-1)^2 Tk} > O\left(\frac{1}{n}\frac{\ln(Tk)}{T^{\alpha}k^{\alpha}}\right) \tag{7}$$

the optimum α is not 1.

Superiority of LA: Smaller Optimization & Generalization Errors

Theorem 2 (informal). $\lambda-$**strongly-convex problem** with proper assumptions:

$$\text{optimization error} \le \begin{cases} O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2 (kT)^{2\alpha}(1-2\alpha)}\right), & 0<\alpha<\frac{1}{2}, \\ O\left(\frac{1}{T} + \frac{\log(Tk)}{\lambda^2 kT}\right), & \alpha = \frac{1}{2}, \\ O\left(\frac{1}{T^{2\alpha}} + \frac{1}{\lambda^2 (2\alpha-1)kT}\right), & \frac{1}{2}<\alpha\le 1. \end{cases} \quad \text{generalization error} \le O\left(\frac{1}{n\lambda}\frac{(Tk+1)^{\alpha}-1}{((T+1)k+2)^{\alpha}}\right) \tag{6}$$

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

$$\frac{\ln T}{T^{\alpha}} > O\left(\frac{1}{n\lambda}\frac{\ln(Tk)}{k^{\alpha}}\right) \quad \text{or} \quad \frac{1}{\lambda(\alpha-1)^2 Tk} > O\left(\frac{1}{n}\frac{\ln(Tk)}{T^{\alpha}k^{\alpha}}\right) \tag{7}$$

the optimum α is not 1.

This also explains **why Lookahead enjoys smaller excess risk error (test error) than SGD**

Superiority of LA: Smaller Optimization & Generalization Errors

Theorem 3 (informal). **On nonconvex problem with PL condition**

$$2\mu(f(w) - f(w^*)) \leq \|\nabla f(w)\|^2 \ (\forall w,\ w^* \in \mathrm{argmin}_w f(w))$$

then we have

$$\text{optimization error} \leq O\Big(\frac{1}{(Tk+1)^{2\alpha}} + \frac{\alpha\,(\alpha + 2(1-\alpha)(k-1))}{\mu^2 (Tk+1)^{2\alpha-1}}\Big)$$

$$\text{generalization error} \leq O\Big(\frac{1}{n-1}\alpha^{\frac{1}{1+\gamma}}(Tk)^{\frac{\gamma}{\gamma+1}}\Big) \quad \Big(\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu}\Big)$$

Superiority of LA: Smaller Optimization & Generalization Errors

Theorem 3 (informal). **On nonconvex problem with PL condition**

$$2\mu(f(w) - f(w^*)) \leq \|\nabla f(w)\|^2 \ (\forall w,\ w^* \in \mathrm{argmin}_w f(w))$$

then we have

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

$$\text{generalization error} \leq O\Big(\frac{1}{n-1}\alpha^{\frac{1}{1+\gamma}}(Tk)^{\frac{\gamma}{\gamma+1}}\Big) \quad \Big(\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu}\Big)$$

By properly choosing α ,

Lookahead can also enjoys smaller excess risk error (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

Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_Q$.

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**

  $\boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;

  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

Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

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

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

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**

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

$\quad \boldsymbol{\theta}_q = \text{Look-ahead}(F_q(\boldsymbol{\theta}), \eta_q, T_q, \alpha_q, k_q, \boldsymbol{\theta}_{q-1}, \mathcal{A}, \mathcal{S})$.

end

Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_Q$.

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**

$\quad \boldsymbol{v}_0^{(t)} = \boldsymbol{\theta}_{t-1}$;

$\quad$ **for** $\tau = 1, 2, ..., k$ **do**

$\quad\quad \boldsymbol{v}_\tau^{(t)} = \mathcal{A}(F_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{v}_{\tau-1}^{(t)}, \eta_{\tau-1}^{(t)}, \mathcal{S}) = \boldsymbol{v}_{\tau-1}^{(t)} - \eta_{\tau-1}^{(t)}\boldsymbol{g}_{\tau-1}^{(t)}$

$\quad$ **end**

$\quad \boldsymbol{\theta}_t = (1-\alpha)\boldsymbol{\theta}_{t-1} + \alpha \boldsymbol{v}_k^{(t)}$.

end

Output : $\boldsymbol{\theta}_{\mathcal{A},\mathcal{S}} = \boldsymbol{\theta}_T$

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

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

Advantages:

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

Superiority of SLRLA: Smaller Optimization & Generalization Errors

Theorem 4 (informal). 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) $\alpha \in (0, \frac{1}{2})$	LookAhead (LA) $\alpha = \frac{1}{2}$	LookAhead (LA) $\alpha \in (\frac{1}{2}, 1]$	SLRLA $\alpha \in (0, 1]$
λ-strongly-convex problems	$\mathcal{O}\left((\frac{1}{\epsilon})^{\frac{1}{2\alpha}} + (\frac{1}{(1-2\alpha)\lambda^2\epsilon})^{\frac{1}{2\alpha}}\right)$	$\mathcal{O}\left(\frac{\log\frac{1}{\epsilon}}{\lambda^2\epsilon}\right)$	$\mathcal{O}\left(\frac{1}{(2\alpha-1)\lambda^2\epsilon}\right)$	$\mathcal{O}(\frac{1}{\lambda\alpha\epsilon})$
nonconvex problems with μ-PL	$\mathcal{O}((\frac{1}{\mu^2\epsilon})^{1/\alpha})$			$\mathcal{O}(\frac{1}{\mu\alpha\epsilon})$

Superiority of SLRLA: Smaller Optimization & Generalization Errors

Theorem 4 (informal). 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) $\alpha \in (0, \frac{1}{2})$	LookAhead (LA) $\alpha = \frac{1}{2}$	LookAhead (LA) $\alpha \in (\frac{1}{2}, 1]$	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	$\mathcal{O}((\frac{1}{\mu^2\epsilon})^{1/\alpha})$			$\mathcal{O}(\frac{1}{\mu\alpha\epsilon})$

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

Superiority of SLRLA: Smaller Optimization & Generalization Errors

Theorem 4 (informal). 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	$\mathcal{O}(\frac{1}{n}(Tk)^{\frac{\gamma}{\gamma+1}})$ $(\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu})$	$\mathcal{O}(\frac{1}{n}/(\frac{c}{\alpha}+\mu))$ $(c \geq 0)$

Superiority of SLRLA: Smaller Optimization & Generalization Errors

Theorem 4 (informal). 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	$\mathcal{O}(\frac{1}{n}(Tk)^{\frac{\gamma}{\gamma+1}})$ $(\gamma = (1-\frac{1}{n})\frac{\alpha L}{\mu})$	$\mathcal{O}(\frac{1}{n}/(\frac{c}{\alpha}+\mu))$ $(c \geq 0)$

By comparison,

SLRLA has smaller generalization error than LA

Experimental Results

Table 3: Classification accuracy (%). ◇, ∗, †, ‡ are respectively reported in [1], [15], [49], [50].

optimizer	CIFAR10 ResNet18	CIFAR10 VGG16	CIFAR10 WRN-16-10	CIFAR100 ResNet18	CIFAR100 VGG16	CIFAR100 WRN-16-10	ImageNet ResNet18
Adam [11]	94.84◇	91.08	93.54	76.88◇	64.07	74.81	66.54*
Adabound [51]	92.56	91.35	91.68	71.43	64.74	71.64	68.13†
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†
AdaBelief [50]	95.20‡	92.25	95.71	77.02‡	68.63	77.93	70.08‡
Stagewise SGD [13]	95.23±0.19◇	92.13±0.02	95.51±0.02	78.24±0.18◇	69.97±0.02	78.95±0.03	70.23†
SLA [1]	95.27±0.06◇	92.38±0.02	95.73±0.02	78.34±0.05◇	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

SLRLA has better test performance than SGD and (stagewise) LA

Conclusion

- **Problems:**

(1) Why Lookahead enjoys better test performance than SGD?

LookAhead enjoys smaller excess risk error than SGD

(2) How to further improve LookAhead?

we propose a stagewise locally-regularized Lookahead with provable performance

Thanks !