Second-Order Optimization with Lazy Hessians

Nikita Doikov

Joint work with El Mahdi Chayti and Martin Jaggi

EPFL, Switzerland

ICML 2023

$$\min_{x \in \mathbb{R}^d} f(x),$$

where f is twice differentiable, possibly non-convex

Gradient Method. Iterate, $k \geq 0$:

$$x_{k+1} \;=\; x_k - \alpha_k \nabla f(x_k)$$

+ **Cheap iterations:** $\mathcal{O}(d)$
+ **Convergence from arbitrary** x_0
− **Slow rate**

Let the gradient be Lipschitz continuous:

$$\|\nabla f(x) - \nabla f(y)\| \;\leq\; L_1 \|x - y\|, \qquad \forall x, y \in \mathbb{R}^d$$

Then, to find $\|\nabla f(\bar{x}_k)\| \leq \varepsilon$, the method needs

$$K \;=\; \mathcal{O}\Big(\tfrac{L_1(f(x_0) - f^\star)}{\varepsilon^2}\Big)$$

2^{nd}-order assumption. Let the Hessian be Lipschitz continuous:

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \quad \leq \quad L_2 \|x - y\|, \qquad \forall x, y \in \mathbb{R}^d$$

$\Rightarrow$ global upper model of the objective, for $H \geq L_2$:

$$f(y) \quad \leq \quad \Omega(x; y) + \frac{H}{6}\|y - x\|^3, \qquad \forall x, y \in \mathbb{R}^d,$$

where

$$\Omega(x; y) \quad \stackrel{\mathrm{def}}{=} \quad f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}\langle \nabla^2 f(x)(y - x), y - x \rangle$$

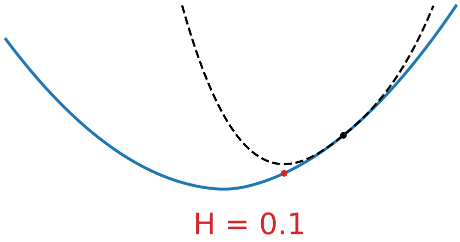
Cubic Newton [Nesterov-Polyak, 2006].
Iterate, $k \geq 0$:

$$x_{k+1} \quad = \quad \operatorname*{argmin}_{y \in \mathbb{R}^d} \Big\{ M_H(x; y) \quad \equiv \quad \Omega(x_k; y) + \frac{H}{6}\|y - x_k\|^3 \Big\}$$

Cubic Model

Regularized quadratic model of $f(y)$ at point $x \in \mathbb{R}^d$:

$$M_H(x; y) \;\; \equiv \;\; \Omega(x; y) + \frac{H}{6}\|y - x\|^3$$



$\Rightarrow$ global progress of the method.

$$x_{k+1} \;=\; \operatorname*{argmin}_{y \in \mathbb{R}^d} \Big\{ M_H(x_k; y) \;\equiv\; \Omega(x_k; y) + \tfrac{H}{6}\|y - x_k\|^3 \Big\}$$

Theorem. Let $H := L_2$. Then, to find $\|\nabla f(\bar{x}_k)\| \leq \varepsilon$, the Cubic Newton needs

$$K \;=\; \mathcal{O}\Big(\tfrac{\sqrt{L_2}(f(x_0) - f^\star)}{\varepsilon^{3/2}} \Big)$$

iterations.

- For the Gradient Method, we had $\mathcal{O}(\frac{1}{\varepsilon^2})$
- We also have convergence to a second-order stationary point for the Cubic Newton: $\nabla^2 f(\bar{x}_k) \succeq -\sqrt{L_2 \varepsilon} I$
- Adaptive strategy for H: ensure $f(x_{k+1}) \leq M_H(x_k; x_{k+1})$

[Nesterov-Polyak, 2006; Cartis-Gould-Toint, 2011; Grapiglia-Nesterov, 2017]

Cubic Newton step:

$$x_{k+1} \;=\; x_k - \left[\nabla^2 f(x_k) + \tau_k I\right]^{-1} \nabla f(x_k)$$

where τ_k is the solution of the dual

For convex functions we can use **Gradient Regularization:**

$$\tau_k \;=\; \sqrt{\frac{H\|\nabla f(x_k)\|}{2}}$$

[Ueda-Yamashita, 2014; Mishchenko, 2021; D-Nesterov, 2021]

- **Fast global rates**
- **High arithmetic cost**

$\Rightarrow$ **this work: Lazy Hessian updates**

It improves the total arithmetic cost of CN by a factor $\sqrt{d}$

Lazy Hessian updates

- Idea: use the same Hessian for $m \geq 1$ iterations

Lazy Hessian Updates: compute new Hessian once per m iterations.

Hessians:	$\nabla^2 f(\mathbf{x}_0)$	$\xrightarrow{\text{reuse Hessian}}$			$\nabla^2 f(\mathbf{x}_m)$	$\xrightarrow{\text{reuse Hessian}}$	
Gradients:	$\nabla f(\mathbf{x}_0)$	$\nabla f(\mathbf{x}_1)$	$\ldots$	$\nabla f(\mathbf{x}_{m-1})$	$\nabla f(\mathbf{x}_m)$	$\nabla f(\mathbf{x}_{m+1})$	$\ldots$

Appeared first in [Shamanskii, 1967]

[Lampariello-Sciandrone, 2001; Wang-Chen-Du, 2006; Fan, 2013]

Define step of the method with Hessian at some previous point z:

$$T_H(x, z) \;=\; \operatorname*{argmin}_{y \in \mathbb{R}^d} \Big\{ \langle \nabla f(x), y - x \rangle + \tfrac{1}{2} \langle \nabla^2 f(z)(y - x), y - x \rangle + \tfrac{H}{6} \|y - x\|^3 \Big\}$$

Define

$$\pi(k) \;\stackrel{\mathrm{def}}{=}\; k - k \bmod m$$

Cubic Newton with Lazy Hessians

Iterate, $k \geq 0$:

1. Set last snapshot point $z_k = x_{\pi(k)}$
2. Compute lazy cubic step $x_{k+1} = T_H(x_k, z_k)$

Convergence Rate

Theorem. Let $H := 6mL_2$. Then, to find $\|\nabla f(\bar{x})\| \leq \varepsilon$, the method needs

$$K \quad = \quad \mathcal{O}\Bigl(\frac{\sqrt{mL_2}(f(x_0)-f^\star)}{\varepsilon^{3/2}}\Bigr)$$

lazy steps.

- Worse than the full Cubic Newton by the factor $\sqrt{m}$

Note: the total number of Hessian updates during these steps is

$$\frac{K}{m} \quad = \quad \mathcal{O}\Bigl(\frac{\sqrt{L_2}(f(x_0)-f^\star)}{\sqrt{m}\varepsilon^{3/2}}\Bigr)$$

Arithmetic Cost

» **Choice of** m**?** Optimize the total cost:

$$\texttt{Arithmetic complexity} \;=\; K \times \texttt{GradCost} + \frac{K}{m} \times \texttt{HessCost}$$

In many problems: $\boxed{\texttt{HessCost} \;=\; d \times \texttt{GradCost}}$

- Logistic Regression, Generalized Linear Models
- Neural Networks

$\Rightarrow$ optimal choice

$$\boxed{m \;:=\; d}$$

(update the Hessian once every d steps)

Total arithmetic complexity

- Gradient Method:

$$\mathcal{O}\left(\frac{L_1(f(x_0)-f^\star)}{\varepsilon^2}\right) \times \texttt{GradCost}$$

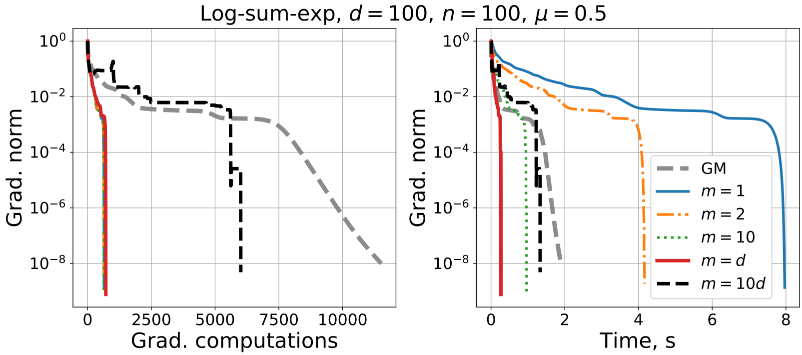
- Full Cubic Newton:

$$\mathcal{O}\left(\frac{\sqrt{L_2}(f(x_0)-f^\star)}{\varepsilon^{3/2}}\right) \times \texttt{GradCost} \times d$$

- Lazy Cubic Newton ($m = d$):

$$\mathcal{O}\left(\frac{\sqrt{L_2}(f(x_0)-f^\star)}{\varepsilon^{3/2}}\right) \times \texttt{GradCost} \times \sqrt{d}$$

$$\min_{x \in \mathbb{R}^d} f(x) \quad := \quad \mu \ln\Bigg(\sum_{i=1}^{n} \exp\Big(\frac{\langle a_i, x\rangle - b_i}{\mu}\Big)\Bigg) \quad \approx \quad \max_{1 \le i \le n} \big[\langle a_i, x\rangle - b_i\big].$$



Conclusions

- Using cubic regularization or gradient regularization for Newton's method we can establish global convergence

- With lazy Hessian updates we improve the total arithmetic complexity

Research directions:

- Convex optimization
- Stochastic methods (we have a follow-up work)
- Sparse problems (different schedules of updating the Hessian)

Thank you very much for your attention!