Optimization for Machine Learning
机器学习中的优化方法

陈 程

华东师范大学 软件工程学院

chchen@sei.ecnu.edu.cn

Outline

(Proximal) Gradient Methods

Iteration complexities of (proximal) gradient methods

- strongly convex and smooth problems

$$O\left(\kappa \log \frac{1}{\epsilon}\right)$$

- convex and smooth problems

$$O\left(\frac{1}{\epsilon}\right)$$

Can we have better convergence rate?

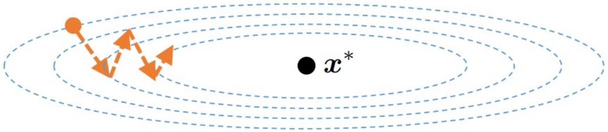
Polyak's Heavy-ball Method

Heavy ball Method (HB):

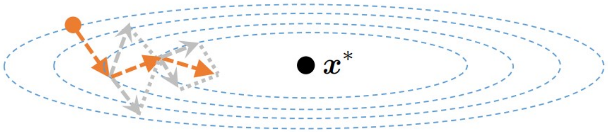
$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \nabla f(\mathbf{x}_t) + \underbrace{\theta_t(\mathbf{x}_t - \mathbf{x}_{t-1})}_{\text{momentum term}}$$

- add inertia to the "ball" (i.e. include a momentum term) to mitigate zigzagging

Polyak's Heavy ball Method



gradient descent



heavy-ball method

Polyak's Heavy ball Method

Theorem (Convergence of heavy ball methods)

Suppose f is a L-smooth and μ-strongly convex quadratic function. If we choose $\eta_t = 4/(\sqrt{L} + \sqrt{\mu})^2$, $\theta_t = \max\{|1 - \sqrt{\eta_t L}|, |1 - \sqrt{\eta_t \mu}|\}^2$ *and* $\kappa = L/\mu$, *then*

$$\left\| \begin{bmatrix} \mathbf{x}_{t+1} - \mathbf{x}^* \\ \mathbf{x}_t - \mathbf{x}^* \end{bmatrix} \right\|_2 \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^t \left\| \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}^* \\ \mathbf{x}_0 - \mathbf{x}^* \end{bmatrix} \right\|_2$$

- only have convergence guarantee for quadratic function
- significant improvement over GD: $O\left(\sqrt{\kappa} \log \frac{1}{\epsilon}\right)$ v.s. $O\left(\kappa \log \frac{1}{\epsilon}\right)$

Can we obtain improvement for more general convex cases as well?

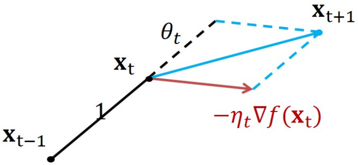
Nesterov's idea

Nesterov's accelerated gradient (NAG) method:

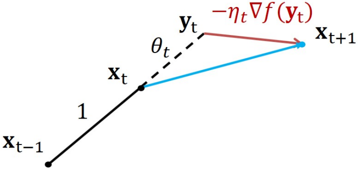
$$\mathbf{y}_t = \mathbf{x}_t + \theta_t(\mathbf{x}_t - \mathbf{x}_{t-1})$$
$$\mathbf{x}_{t+1} = \mathbf{y}_t - \eta_t \nabla f(\mathbf{y}_t)$$

- alternates between gradient updates and proper extrapolation
- not a descent method (i.e. we may not have $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t)$)
- one of the most beautiful and mysterious results in optimizatio

Comparison between HB and NAG



Heavy ball



NAG

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \nabla f(\mathbf{x}_t) + \theta_t(\mathbf{x}_t - \mathbf{x}_{t-1})$$

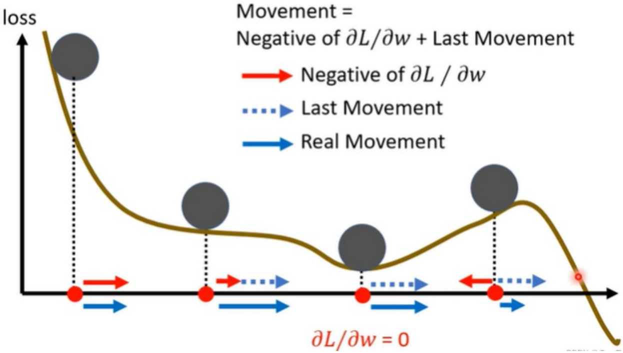
$$\begin{cases} \mathbf{y}_t = \mathbf{x}_t + \theta_t(\mathbf{x}_t - \mathbf{x}_{t-1}) \\ \mathbf{x}_{t+1} = \mathbf{y}_t - \eta_t \nabla f(\mathbf{y}_t) \end{cases}$$

History

- Polyak invented HB momentum in 1964 (and discussed the physics analogy)
- Nesterov invented NAG in 1983
  - Even though Nesterov was Polyak's student, he seems not to have mentioned the physics analogy
- Sutskever et al. (2013)[1] popularized momentum methods in machine learning and revived the momentum interpretation.

[1] On the importance of initialization and momentum in deep learning. ICML 2013.

Momentum methods for nonconvex problems



Convergence Rate of NAG

Theorem

Suppose f is μ-strongly convex and L-smooth. If we choose $\eta_t = \eta = 1/L$ and $\theta_t = \theta = \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$, then

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1} \left[f(\mathbf{x}_1) - f(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2\right]$$

Convergence Rate of NAG

Theorem

Suppose f is convex and L-smooth. If we choose $\eta_t = \eta = 1/L$ and $\theta_t = \frac{\lambda_t - 1}{\lambda_{t+1}}$ where $\lambda_0 = 1$ and $\lambda_{t+1} = \frac{1+\sqrt{1+4\lambda_t^2}}{2}$. Then

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{t-1} \left[f(\mathbf{x}_1) - f(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}_1 - \mathbf{x}^*\|_2^2\right]$$

- A simpler choice the θ_t is $\theta_t = \frac{t}{t+3}$.

Extension to Composite Models

$$min_{\mathbf{x}} F(\mathbf{x}) = f(\mathbf{x}) + h(\mathbf{x})$$

- f is convex and smooth
- h is convex (may not be differentiable)
- Let $F^* = \min_{\mathbf{x}} F(\mathbf{x})$ be the optimal value

FISTA (Beck & Teboulle '09)

Fast iterative shrinkage-thresholding algorithm:

$$
\begin{aligned}
\mathbf{y}_t &= \mathrm{prox}_{\eta_t h}(\mathbf{x}_t + \theta_t(\mathbf{x}_t - \mathbf{x}_{t-1})) \\
\mathbf{x}_{t+1} &= \mathbf{y}_t - \eta_t \nabla f(\mathbf{y}_t)
\end{aligned}
$$

- has same convergence property as the convex problems
- fast if prox can be efficiently implemented

Outline

Lower Bounds

Interestingly, no first-order methods can improve upon Nesterov's results in general.

More precisely, there exists convex and L-smooth function f s.t.

$$f(\mathbf{x}) - f^* \geq \frac{3L \left\|\mathbf{x}_0 - \mathbf{x}^*\right\|_2^2}{32(t+1)^2}$$

as long as $\underbrace{\mathbf{x}_k \in \mathbf{x}_0 + \text{span}\{\nabla f(\mathbf{x}_0), \dots, \nabla f(\mathbf{x}_{k-1})\}}_{\text{definition of first-order methods}}$ for all $1 \leq k \leq t$.

Example

$$\min_{\mathbf{x}\in\mathbb{R}^{2n+1}} f(\mathbf{x}) = \frac{L}{4}\left(\frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{e}_1^\top\mathbf{x}\right)$$

$$\text{where } \mathbf{A} = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)}$$

- f is convex and smooth
- the optima $\mathbf{x}^*$ is given by $x_i^* = 1 - \frac{i}{2n+2}(1 \leq i \leq n)$.

Outline

Newton's Method

Recall that optimizing smooth function $f(\mathbf{x})$ by gradient descent

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \frac{1}{L}\nabla f(\mathbf{x}_t)$$

is achieved by minimizing

$$\min_{\mathbf{x}} f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle + \frac{L}{2}\left\|\mathbf{x} - \mathbf{x}_t\right\|_2^2.$$

If we can compute Hessian matrix, we can minimize

$$\min_{\mathbf{x}} f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle + \frac{1}{2}\langle \mathbf{x} - \mathbf{x}_t, \nabla^2 f(\mathbf{x}_t)(\mathbf{x} - \mathbf{x}_t)\rangle.$$

Suppose $\nabla^2 f(\mathbf{x}_t)$ is non-singular, then we achieve Newton's method

$$\mathbf{x}_{t+1} = \mathbf{x}_t - (\nabla^2 f(\mathbf{x}_t))^{-1}\nabla f(\mathbf{x}_t).$$

Quadratic Convergence

Theorem

Suppose the twice differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ *has* L_2*-Lipschitz continuous Hessian and local minimizer* $\mathbf{x}^*$ *with* $\nabla^2 f(\mathbf{x}^*) \succeq \mu \mathbf{I}$*, then the Newton's method*

$$\mathbf{x}_{t+1} = \mathbf{x}_t - (\nabla^2 f(\mathbf{x}_t))^{-1} \nabla f(\mathbf{x}_t)$$

with $\|\mathbf{x}_0 - \mathbf{x}^*\|_2 \leq \mu/(2L_2)$ *holds that*

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|_2 \leq \frac{L_2}{\mu} \|\mathbf{x}_t - \mathbf{x}^*\|_2^2 .$$

Newton's method has local quadratic convergence, which requires

$$T = \mathcal{O}(\log \log(1/\epsilon))$$

iterations to achieve $\|\mathbf{x}_T - \mathbf{x}^*\|_2 \leq \epsilon$.

Standard Newton's Method

Strengths:

1. The quadratic convergence is very fast (even for ill-conditioned case).

Weakness:

1. The convergence guarantee is local.
2. Each iteration requires $O(d^3)$ time.

Secant Condition

For quadratic function

$$Q(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbf{b}^\top \mathbf{x},$$

we have $\nabla Q(\mathbf{x}_{t+1}) - \nabla Q(\mathbf{x}_t) = \nabla^2 Q(\mathbf{x}_{t+1})(\mathbf{x}_{t+1} - \mathbf{x}_t)$.

For general $f(\mathbf{x})$ with Lipschitz continuous Hessian, we have

$$\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t) = \nabla^2 f(\mathbf{x}_{t+1})(\mathbf{x}_{t+1} - \mathbf{x}_t) + o(\|\mathbf{x}_{t+1} - \mathbf{x}_t\|_2),$$

which leads to

$$\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t) \approx \nabla^2 f(\mathbf{x}_{t+1})(\mathbf{x}_{t+1} - \mathbf{x}_t).$$

Classical Quasi-Newton Methods

Motivated by

$$\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t) \approx \nabla^2 f(\mathbf{x}_{t+1})(\mathbf{x}_{t+1} - \mathbf{x}_t),$$

classical Quasi-Newton methods target to find $\mathbf{G}_{t+1}$ such that

$$\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t) = \mathbf{G}_{t+1}(\mathbf{x}_{t+1} - \mathbf{x}_t)$$

and update the variable as

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \mathbf{G}_t^{-1}\nabla f(\mathbf{x}_t).$$

We typically take $\mathbf{G}_0 = \delta_0 \mathbf{I}$ with some $\delta_0 > 0$.

For given $\mathbf{G}_t$ or $\mathbf{G}_t^{-1}$, we hope

1. $\{\mathbf{x}_t\}$ converges to $\mathbf{x}^*$ efficiently;
2. $\mathbf{G}_{t+1}$ is close to $\mathbf{G}_t$;
3. $\mathbf{G}_{t+1}$ or $\mathbf{G}_{t+1}^{-1}$ can be constructed/stored efficiently.

Woodbury Matrix Identity

The Woodbury matrix identity is

$$(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1},$$

where $\mathbf{A} \in \mathbb{R}^{d\times d}$, $\mathbf{C} \in \mathbb{R}^{k\times k}$, $\mathbf{U} \in \mathbb{R}^{d\times k}$ and $\mathbf{V} \in \mathbb{R}^{k\times d}$.

For $\mathbf{A} = \mathbf{G}_t$, $\mathbf{U} = \mathbf{Z}_t$, $\mathbf{V} = \mathbf{Z}_t^\top$ and $\mathbf{C} = \mathbf{I}$, we let

$$\mathbf{G}_{t+1} = \mathbf{G}_t + \mathbf{Z}_t\mathbf{Z}_t^\top,$$

then

$$\mathbf{G}_{t+1}^{-1} = \mathbf{G}_t^{-1} - \mathbf{G}_t^{-1}\mathbf{Z}_t(\mathbf{I} + \mathbf{Z}_t^\top\mathbf{G}_t^{-1}\mathbf{Z}_t)^{-1}\mathbf{Z}_t^\top\mathbf{G}_t^{-1}$$

can be computed within $\mathcal{O}(kd^2)$ flops for given $\mathbf{G}_t^{-1}$.

Classical SR1 Method

We consider secant condition and the symmetric rank one (SR1) update

$$\begin{cases} \mathbf{y}_t = \mathbf{G}_{t+1}\mathbf{s}_t, \\ \mathbf{G}_{t+1} = \mathbf{G}_t + \mathbf{z}_t\mathbf{z}_t^\top. \end{cases}$$

where $\mathbf{s}_t = \mathbf{x}_{t+1} - \mathbf{x}_t$ and $\mathbf{y}_t = \nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t)$.

It implies

$$\mathbf{G}_{t+1} = \mathbf{G}_t + \frac{(\mathbf{y}_t - \mathbf{G}_t\mathbf{s}_t)(\mathbf{y}_t - \mathbf{G}_t\mathbf{s}_t)^\top}{(\mathbf{y}_t - \mathbf{G}_t\mathbf{s}_t)^\top\mathbf{s}_t}.$$

and the corresponding update to inverse of Hessian estimator is

$$\mathbf{G}_{t+1}^{-1} = \mathbf{G}_t^{-1} + \frac{(\mathbf{s}_t - \mathbf{G}_t^{-1}\mathbf{y}_t)(\mathbf{s}_t - \mathbf{G}_t^{-1}\mathbf{y}_t)^\top}{(\mathbf{s}_t - \mathbf{G}_t^{-1}\mathbf{y}_t)^\top\mathbf{y}_t}.$$

Classical DFP Method

Let $\mathbf{G}_{t+1}$ be the solution of following matrix optimization problem

$$\min_{\mathbf{G}\in\mathbb{R}^{d\times d}} \|\mathbf{G}-\mathbf{G}_t\|_{\bar{\mathbf{G}}_t^{-1}}$$
$$\text{s.t} \quad \mathbf{G}=\mathbf{G}^\top, \ \ \mathbf{G}\mathbf{s}_t=\mathbf{y}_t,$$

where the weighted norm $\|\cdot\|_{\bar{\mathbf{G}}_t}$ is defined as

$$\|\mathbf{A}\|_{\bar{\mathbf{G}}_t} = \left\|\bar{\mathbf{G}}_t^{-1/2}\mathbf{A}\bar{\mathbf{G}}_t^{-1/2}\right\|_F \quad \text{with} \quad \bar{\mathbf{G}}_t = \int_0^1 \nabla^2 f(\mathbf{x}_t + \tau(\mathbf{x}_{t+1}-\mathbf{x}_t))\,\mathrm{d}\tau.$$

It implies DFP update

$$\mathbf{G}_{t+1} = \left(\mathbf{I} - \frac{\mathbf{y}_t\mathbf{s}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}\right)\mathbf{G}_t\left(\mathbf{I} - \frac{\mathbf{s}_t\mathbf{y}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}\right) + \frac{\mathbf{y}_t\mathbf{y}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}.$$

The corresponding update to inverse of Hessian estimator is

$$\mathbf{G}_{t+1}^{-1} = \mathbf{G}_t^{-1} - \frac{\mathbf{G}_t^{-1}\mathbf{y}_t\mathbf{y}_t^\top\mathbf{G}_t^{-1}}{\mathbf{y}_t^\top\mathbf{G}_t^{-1}\mathbf{y}_t} + \frac{\mathbf{s}_t\mathbf{s}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}.$$

Classical BFGS Method

Let $\mathbf{G}_{t+1}^{-1}$ be the solution of the following matrix optimization problem

$$\begin{aligned} \min_{\mathbf{H}\in\mathbb{R}^{d\times d}} \; & \|\mathbf{H}-\mathbf{H}_t\|_{\bar{\mathbf{G}}_t} \\ \text{s.t} \quad & \mathbf{H}=\mathbf{H}^\top, \;\; \mathbf{H}\mathbf{y}_t=\mathbf{s}_t, \end{aligned}$$

where $\mathbf{H}_t = \mathbf{G}_t^{-1}$ and the weighted norm $\|\cdot\|_{\bar{\mathbf{G}}_t}$ is defined as

$$\|\mathbf{A}\|_{\bar{\mathbf{G}}_t} = \left\|\bar{\mathbf{G}}_t^{1/2}\mathbf{A}\bar{\mathbf{G}}_t^{1/2}\right\|_F \quad \text{with} \quad \bar{\mathbf{G}}_t = \int_0^1 \nabla^2 f(\mathbf{x}_t + \tau(\mathbf{x}_{t+1}-\mathbf{x}_t))\,\mathrm{d}\tau.$$

It implies BFGS update

$$\mathbf{G}_{t+1}^{-1} = \left(\mathbf{I} - \frac{\mathbf{s}_t\mathbf{y}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}\right)\mathbf{G}_t^{-1}\left(\mathbf{I} - \frac{\mathbf{y}_t\mathbf{s}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}\right) + \frac{\mathbf{s}_t\mathbf{s}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}.$$

The corresponding update to Hessian estimator is

$$\mathbf{G}_{t+1} = \mathbf{G}_t - \frac{\mathbf{G}_t\mathbf{s}_t\mathbf{s}_t^\top\mathbf{G}_t}{\mathbf{s}_t^\top\mathbf{G}_t\mathbf{s}_t} + \frac{\mathbf{y}_t\mathbf{y}_t^\top}{\mathbf{y}_t^\top\mathbf{s}_t}.$$

Local superlinear convergence

Theorem (informal)

Suppose f is strongly convex and has Lipschitz-continuous Hessian. Under mild conditions, BFGS achieves

$$\lim_{t\to\infty} \frac{\left\|\mathbf{x}_{t+1} - \mathbf{x}^*\right\|_2}{\left\|\mathbf{x}_t - \mathbf{x}^*\right\|_2} = 0$$

- iteration complexity: larger than Newton methods but smaller than gradient methods
- asymptotic result: holds when $t \to \infty$

Questions