牛顿法(续)

王尧

西安交通大学智能决策与机器学习中心
(Email: yao.s.wang@gmail.com)

2022. 5

回顾：牛顿法

Recall the motivation for gradient descent step at x: we minimize the quadratic approximation

$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t}\|y - x\|_2^2$$

over y, and this yields the update $x^+ = x - t\nabla f(x)$

Newton's method uses in a sense a better quadratic approximation

$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x)$$

and minimizes over y to yield $x^+ = x - (\nabla^2 f(x))^{-1} \nabla f(x)$

注：若Hessian为正定矩阵，则牛顿方向为下降方向

回顾：二次逼近的几何展示

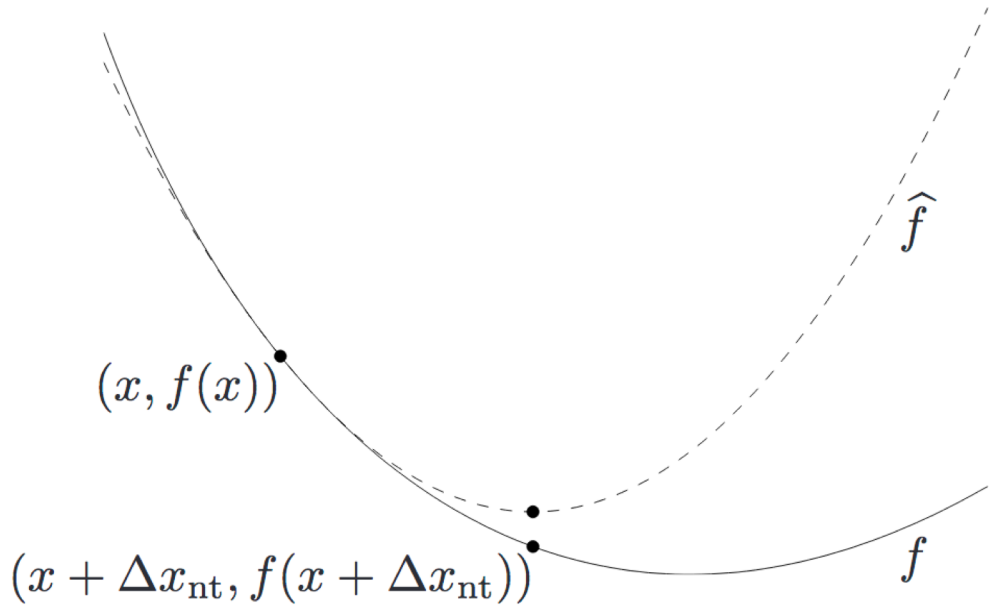


Figure 9.16 The function f (shown solid) and its second-order approximation $\widehat{f}$ at x (dashed). The Newton step Δx_{nt} is what must be added to x to give the minimizer of $\widehat{f}$.

Boyd版9.5.1节

回顾：牛顿减量

At a point x, we define the Newton decrement as

$$\lambda(x) = \left(\nabla f(x)^T \left(\nabla^2 f(x)\right)^{-1} \nabla f(x)\right)^{1/2}$$

This relates to the difference between $f(x)$ and the minimum of its quadratic approximation:

$$\begin{aligned} & f(x) - \min_y \left(f(x) + \nabla f(x)^T (y-x) + \frac{1}{2}(y-x)^T \nabla^2 f(x)(y-x)\right) \\ &= f(x) - \left(f(x) - \frac{1}{2}\nabla f(x)^T \left(\nabla^2 f(x)\right)^{-1} \nabla f(x)\right) \\ &= \frac{1}{2}\lambda(x)^2 \end{aligned}$$

注：牛顿减量可用于迭代算法的终止条件

回顾：后退线性搜索

In practice, we use damped Newton's method (typically just called Newton's method), which repeats

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

Note that the pure method uses $t = 1$

Step sizes here are chosen by backtracking search, with parameters $0 < \alpha \leq 1/2$, $0 < \beta < 1$. At each iteration, start with $t = 1$, while

$$f(x + tv) > f(x) + \alpha t \nabla f(x)^T v$$

we shrink $t = \beta t$, else we perform the Newton update. Note that here $v = -(\nabla^2 f(x))^{-1}\nabla f(x)$, so $\nabla f(x)^T v = -\lambda^2(x)$

注：下降算法均可与后退线性搜索结合使用

回顾：收敛性结果

assumptions

- f strongly convex on S with constant m
- $\nabla^2 f$ is Lipschitz continuous on S, with constant $L > 0$:

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L\|x - y\|_2$$

conclusion: number of iterations until $f(x) - p^\star \leq \epsilon$ is bounded above by

$$\frac{f(x^{(0)}) - p^\star}{\gamma} + \log_2 \log_2(\epsilon_0/\epsilon)$$

- γ, ϵ_0 are constants that depend on m, L, $x^{(0)}$
- second term is small (of the order of 6) and almost constant for practical purposes
- in practice, constants m, L (hence γ, ϵ_0) are usually unknown

回顾：收敛性结果

damped Newton phase ($\|\nabla f(x)\|_2 \geq \eta$)

- most iterations require backtracking steps
- function value decreases by at least γ
- if $p^\star > -\infty$, this phase ends after at most $(f(x^{(0)}) - p^\star)/\gamma$ iterations

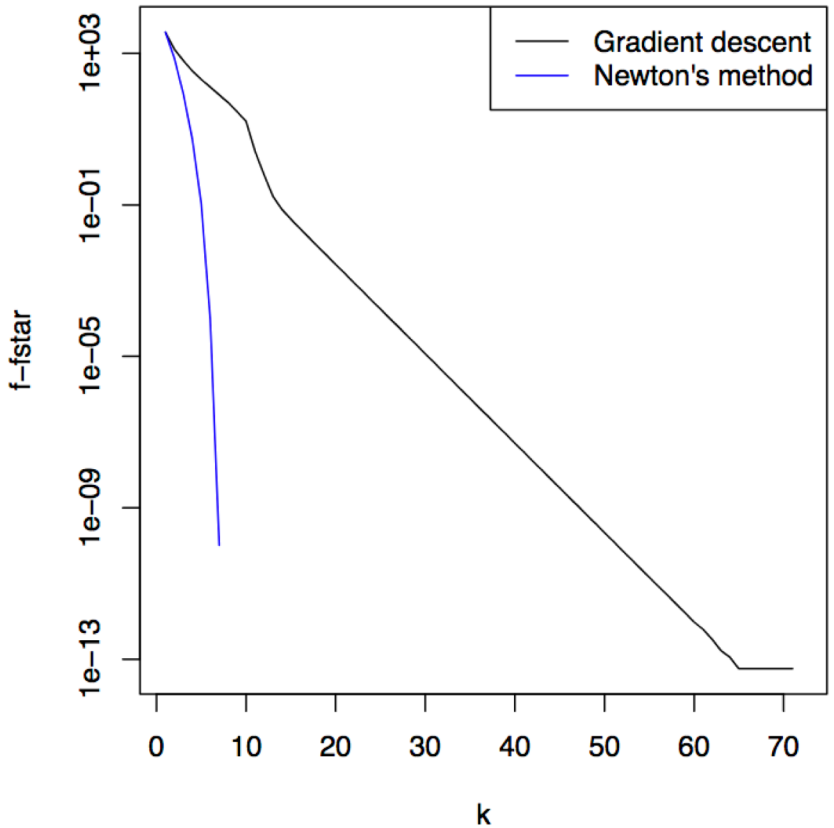
quadratically convergent phase ($\|\nabla f(x)\|_2 < \eta$)

- all iterations use step size $t = 1$
- $\|\nabla f(x)\|_2$ converges to zero quadratically: if $\|\nabla f(x^{(k)})\|_2 < \eta$, then

$$\frac{L}{2m^2}\|\nabla f(x^l)\|_2 \leq \left(\frac{L}{2m^2}\|\nabla f(x^k)\|_2\right)^{2^{l-k}} \leq \left(\frac{1}{2}\right)^{2^{l-k}}, \qquad l \geq k$$

Ridge Logistic Regression

Logistic regression example, with $n = 500$, $p = 100$: we compare gradient descent and Newton's method, both with backtracking



相比梯度法的线性收敛率，牛顿法呈现出更快的超线性收敛率

注：感兴趣的同学可加入加速梯度法的对比

Self-concordant函数

definition

- convex $f : \mathbf{R} \to \mathbf{R}$ is self-concordant if $|f'''(x)| \leq 2f''(x)^{3/2}$ for all $x \in \mathbf{dom}\, f$
- $f : \mathbf{R}^n \to \mathbf{R}$ is self-concordant if $g(t) = f(x + tv)$ is self-concordant for all $x \in \mathbf{dom}\, f$, $v \in \mathbf{R}^n$

examples on R

- linear and quadratic functions
- negative logarithm $f(x) = -\log x$
- negative entropy plus negative logarithm: $f(x) = x \log x - \log x$

Self-concordant函数计算

properties

- preserved under positive scaling $\alpha \geq 1$, and sum
- preserved under composition with affine function
- if g is convex with $\mathbf{dom}\, g = \mathbf{R}_{++}$ and $|g'''(x)| \leq 3g''(x)/x$ then

$$f(x) = \log(-g(x)) - \log x$$

is self-concordant

examples: properties can be used to show that the following are s.c.

- $f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)$ on $\{x \mid a_i^T x < b_i,\ i = 1, \ldots, m\}$
- $f(X) = -\log\det X$ on $\mathbf{S}_{++}^n$
- $f(x) = -\log(y^2 - x^T x)$ on $\{(x, y) \mid \|x\|_2 < y\}$

课后作业：请完成上述命题与实例的证明

Self-concordant函数的收敛性

summary: there exist constants $\eta \in (0, 1/4]$, $\gamma > 0$ such that

- if $\lambda(x) > \eta$, then

$$f(x^{(k+1)}) - f(x^{(k)}) \leq -\gamma$$

- if $\lambda(x) \leq \eta$, then

$$2\lambda(x^{(k+1)}) \leq \left(2\lambda(x^{(k)})\right)^2$$

(η and γ only depend on backtracking parameters α, β)

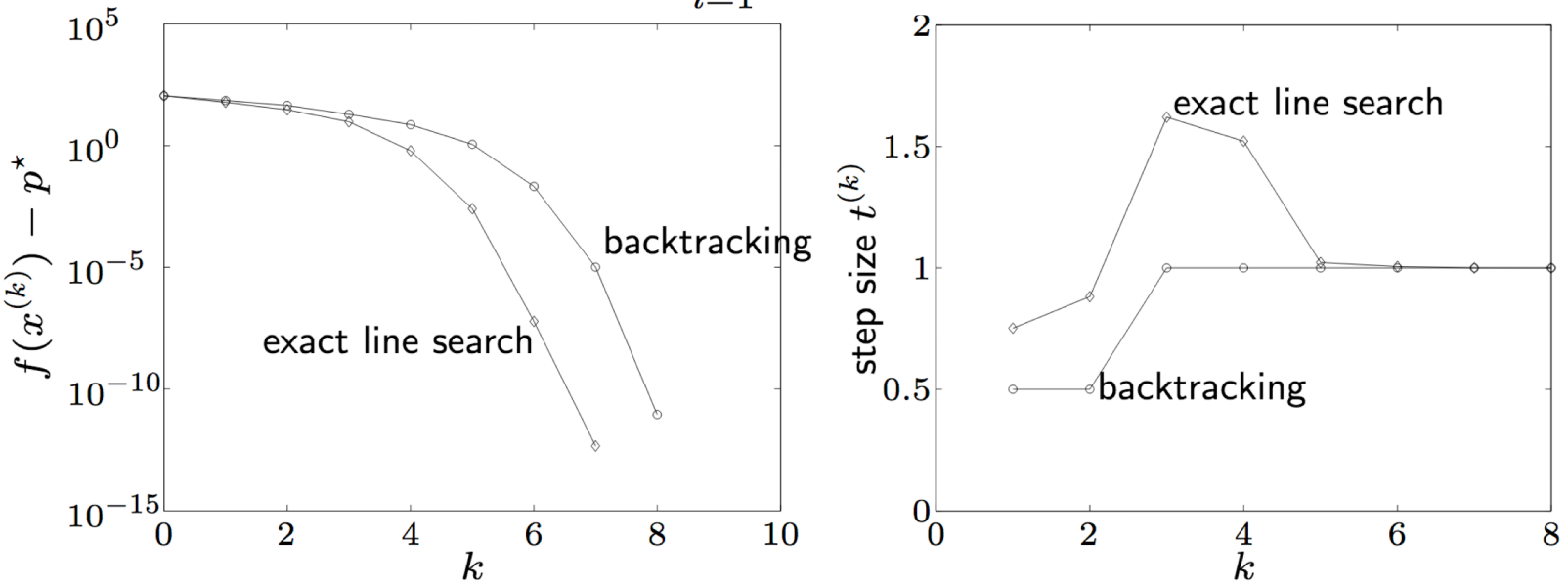
complexity bound: number of Newton iterations bounded by

$$\frac{f(x^{(0)}) - p^\star}{\gamma} + \log_2 \log_2(1/\epsilon)$$

细节见Boyd版9.6.4节

一个高维例子

$$f(x) = c^T x - \sum_{i=1}^{500} \log(b_i - a_i^T x)$$



- backtracking parameters $\alpha = 0.01$, $\beta = 0.5$
- backtracking line search almost as fast as exact l.s. (and much simpler)
- clearly shows two phases in algorithm

拟牛顿法

Two main steps in Newton iteration:

- Compute Hessian $\nabla^2 f(x)$
- Solve the system $\nabla^2 f(x)s = -\nabla f(x)$

Each of these two steps could be expensive

Quasi-Newton methods repeat updates of the form

$$x^+ = x + ts$$

where direction s is defined by linear system

$$Bs = -\nabla f(x)$$

for some approximation B of $\nabla^2 f(x)$. We want B to be easy to compute, and $Bs = g$ to be easy to solve

拟牛顿法

Let $x^{(0)} \in \mathbb{R}^n$, $B^{(0)} \succ 0$. For $k = 1, 2, 3, \ldots$, repeat:

1. Solve $B^{(k-1)} s^{(k-1)} = -\nabla f(x^{(k-1)})$
2. Update $x^{(k)} = x^{(k-1)} + t_k s^{(k-1)}$
3. Compute $B^{(k)}$ from $B^{(k-1)}$

Different quasi-Newton methods implement Step 3 differently.

Basic idea: as $B^{(k-1)}$ already contains info about the Hessian, use suitable matrix update to form $B^{(k)}$

Reasonable requirement for $B^{(k)}$ (motivated by secant method):

$$\nabla f(x^{(k)}) = \nabla f(x^{(k-1)}) + B^{(k)} s^{(k-1)}$$

Broyden-Fletcher-Goldfarb-Shanno (BFGS)

Thank you for your attentions !