Convex Optimization

Part 5: Quasi-Newton methods

Namhoon Lee

POSTECH

23 Nov 2022

Newton method

$$\text{minimize } f(x)$$

f convex, twice continuously differentiable

Newton method

$$x_{k+1} = x_k - t_k \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

- advantages: fast convergence, robustness, affine invariance
- disadvantages: requires second derivatives and solution of linear equation

can be too expensive for large scale applications

Variable metric methods

$$x_{k+1} = x_k - t_k H_k^{-1} \nabla f(x_k)$$

the positive definite matrix H_k is an approximation of the Hessian at x_k, chosen to:

- avoid calculation of second derivatives
- simplify computation of search direction

"Variable metric" interpretation

$$\Delta x = -H^{-1} \nabla f(x)$$

is the steepest descent direction at x for the quadratic norm

$$\|z\|_H = \left(z^\top H z\right)^{1/2}$$

Quasi-Newton methods

given: starting point $x_0 \in \operatorname{dom} f$, $H_0 \succ 0$

---

for $k = 0, 1, \ldots$

1. compute quasi-Newton direction $\Delta x_k = -H_k^{-1} \nabla f(x_k)$
2. determine step size t_k (*e.g.*, by backtracking line search)
3. compute $x_{k+1} = x_k + t_k \Delta x_k$
4. compute H_{k+1}

- different update rules exist for H_{k+1} in step 4
- can also propagate H_k^{-1} or a factorization of H_k to simplify calculation of Δx_k

Broyden-Fletcher-Goldfarb-Shanno (BFGS) update

BFGS update

$$H_{k+1} = H_k + \frac{yy^\top}{y^\top s} - \frac{H_k s s^\top H_k}{s^\top H_k s}$$

where

$$s = x_{k+1} - x_k, \quad y = \nabla f(x_{k+1}) - \nabla f(x_k)$$

Inverse update

$$H_{k+1}^{-1} = \left(I - \frac{sy^\top}{y^\top s}\right) H_k^{-1} \left(I - \frac{ys^\top}{y^\top s}\right) + \frac{ss^\top}{y^\top s}$$

- note that $y^\top s > 0$ for strictly convex f
- cost of update or inverse update is $\mathcal{O}(n^2)$ operations

Positive definiteness

- if $y^\top s > 0$, BFGS update preserves positive definiteness of H_k
- this ensures that $\Delta x = -H_k^{-1}\nabla f(x_k)$ is a descent direction

Proof: from inverse update formula,

$$v^\top H_{k+1}^{-1} v = \left(v - \frac{s^\top v}{s^\top y} y\right)^\top H_k^{-1} \left(v - \frac{s^\top v}{s^\top y} y\right) + \frac{(s^\top v)^2}{y^\top s}$$

- if $H_k \succ 0$, both terms are nonnegative for all v
- second term is zero only if $s^\top v = 0$; then first term is zero only if $v = 0$

Secant condition

the BFGS update satisfies the secant condition

$$H_{k+1}s = y$$

where $s = x_{k+1} - x_k$ and $y = \nabla f(x_{k+1}) - \nabla f(x_k)$

Interpretation: we define a quadratic approximation of f around x_{k+1}

$$\tilde{f}(x) = f(x_{k+1}) + \nabla f(x_{k+1})^\top (x - x_{k+1}) + \frac{1}{2}(x - x_{k+1})^\top H_{k+1}(x - x_{k+1})$$

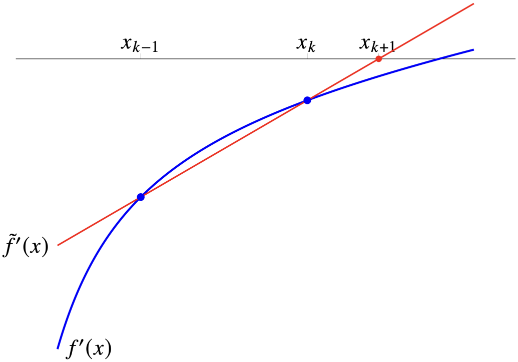
- by construction $\nabla \tilde{f}(x_{k+1}) = \nabla f(x_{k+1})$
- secant condition implies that also $\nabla \tilde{f}(x_k) = \nabla f(x_k)$:

$$\begin{aligned} \nabla \tilde{f}(x_k) &= \nabla f(x_{k+1}) + H_{k+1}(x_k - x_{k+1}) \\ &= \nabla f(x_k) \end{aligned}$$

Secant method

for $f : \mathbb{R} \to \mathbb{R}$, BFGS with unit step size gives the secant method

$$x_{k+1} = x_k - \frac{f'(x_k)}{H_k}, \quad H_k = \frac{f'(x_k) - f'(x_{k-1})}{x_k - x_{k-1}}$$



On secant condition

The secant condition admits infinite number of solutions; *i.e.*, $n(n+1)/2$ degree of freedom with n equations. To determine H_{k+1} uniquely, we impose additional condition that is H_{k+1} and H_k are close to each other:

$$\begin{aligned} \text{minimize} \quad & \|H - H_k\| \\ \text{subject to} \quad & H = H^\top, \quad Hs = y \end{aligned}$$

- different matrix norms can be used, and each norm gives rise to a different quasi–Newton method

BFGS update in detail

BFGS takes a rank-two update

$$H_{k+1} = H_k + auu^\top + bvv^\top$$

the secant equation $H_{k+1}s = y$ yields

$$H_{k+1}s = H_k s + auu^\top s + bvv^\top s = y$$

with $u = y$ and $v = H_k s$ we have

$$H_k s + ayy^\top s + bH_k ss^\top H_k^\top s = y$$

to solve for a and b

$$y(1 - ay^\top s) = H_k s(1 + bs^\top H_k^\top s)$$

which yields $a = \frac{1}{y^\top s}$ and $b = -\frac{1}{s^\top H_k s}$, and plugging this in gives the BFGS update

applying the Sherman-Morrison-Woodbury formula further gives the inverse update

Symmetric rank-one (SR1) update

An update of the form:

$$H_{k+1} = H_k + auu^\top$$

The secant equation $H_{k+1}s = y$ yields

$$(au^\top s)u = y - H_k s$$

This holds if u is a multiple of $y - H_k s$. Thus, with $u = y - H_k s$, we have $a = 1/(y - H_k s)^\top s$, which leads to

$$H_{k+1} = H_k + \frac{(y - H_k s)(y - H_k s)^\top}{(y - H_k s)^\top s}$$

- SR1 is simple but does not preserve positive definiteness

Convergence

Global result

if f is strongly convex, BFGS with backtracking line search converges from any x_0, $H_0 \succ 0$

Local convergence

if f is strongly convex and $\nabla^2 f(x)$ is Lipschitz continuous, local convergence is superlinear: for sufficiently large k,

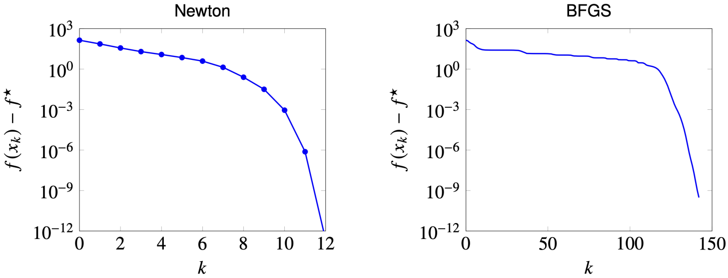
$$\|x_{k+1} - x^\star\|_2 \leq c_k \|x_k - x^\star\|_2$$

where $c_k \to 0$
(quadratic local convergence of Newton method)

Example

$$\text{minimize}\; c^\top x - \sum_{i=1}^{m} \log(b_i - a_i^\top x)$$

$n = 100,\ m = 500$



- cost per Newton iteration: $\mathcal{O}(n^3)$ plus computing $\nabla^2 f(x)$
- cost per BFGS iteration: $\mathcal{O}(n^2)$

Limited memory quasi-Newton methods

main disadvantage of quasi-Newton method is need to store H_k or H_k^{-1}

Limited-memory BFGS (L-BFGS): do not store H_k^{-1} explicitly

- instead we store up to m (*e.g.*, $m = 30$) values of

$$s_j = x_{j+1} - x_j, \quad y_j = \nabla f(x_{j+1}) - \nabla f(x_j)$$

- we evaluate $\Delta x_k = H_k^{-1} \nabla f(x_k)$ recursively, using

$$H_{j+1}^{-1} = \left(I - \frac{s_j y_j^\top}{y_j^\top s_j} \right) H_j^{-1} \left(I - \frac{y_j s_j^\top}{y_j^\top s_j} \right) + \frac{s_j s_j^\top}{y_j^\top s}$$

  for $j = k-1, \dots, k-m$, assuming, for example, $H_{k-m} = I$
- an alternative is to restart after m iterations
- cost per iteration is $\mathcal{O}(nm)$, storage is $\mathcal{O}(nm)$

Further reading

- Numerical Optimization by J. Nocedal and S. J. Wright

Any questions?