A brief introduction to quasi-Newton methods

杨俊锋

南京大学数学系

December 9, 2014

Outline

Notation

Throughout this lecture, we use the following notation

$$
\begin{aligned}
g_k &:= \nabla f(x_k), \quad & G_k &:= \nabla^2 f(x_k), \\
B_k &\approx \nabla^2 f(x_k), \quad & H_k &\approx [\nabla^2 f(x_k)]^{-1}, \\
s_k &= x_{k+1} - x_k, \quad & y_k &= g_{k+1} - g_k.
\end{aligned}
$$

Here "$\approx$" is not necessarily componentwise approximation.

For $k = 0, 1, 2, \ldots$, the iteration scheme is

$$x_{k+1} = x_k + s_k.$$

- Pure Newton's method: $s_k = -G_k^{-1} g_k$.
- Truncated Newton's method: $s_k = -\alpha_k G_k^{-1} g_k$.
- General descent method:

$$s_k = \alpha_k d_k, \quad \text{where} \quad g_k^T d_k < 0$$

and $\alpha_k > 0$ is a step length satisfying certain conditions.

Outline

Outline

Disadvantages of Newton's method

As we have learned before, Newton's method for $\min_{x \in R^n} f(x)$ has the following disadvantages

1. needs to compute G_k, very costly for large problems;
2. needs to solve a large linear system at each iteration;
3. G_k could be (nearly) singular when x_k is far away;
4. if G_k is not positive definite, then Newton direction is not necessarily a descent direction;
5. global convergence is not guaranteed.

- **Can we avoid computing G_k and only use an approximation, say B_k, of it at each iteration?**
- **What are the desirable properties of B_k and how to define B_k at each iteration?**
- **Global convergence is desired!**
- **Fast local convergence should be maintained!**

Outline

Quasi-Newton equation

Consider unconstrained problem $\min_{x \in R^n} f(x)$. Started at x_0, suppose we have already obtained $x_1, x_2, \ldots, x_k$. How to generate the next point x_{k+1}?

We approximate $f(x)$ at x_k by a quadratic function $q_k(x)$:

$$q_k(x) := f_k + g_k^T(x - x_k) + \frac{1}{2}(x - x_k)^T B_k(x - x_k).$$

- Newton's method: $B_k = G_k$. Thus, $q_k(x)$ satisfies

$$q_k(x_k) = f_k, \nabla q_k(x_k) = g_k, \nabla^2 q_k(x_k) = G_k.$$

- quasi-Newton method: B_k approximates G_k in some way. We choose B_k such that

$$\nabla q_k(x_{k-1}) = g_{k-1},$$

  or, equivalently, $B_k s_{k-1} = y_{k-1}$, which is usually referred to as the quasi-Newton equation.

We can also directly approximate G_k^{-1} by H_k, in which case H_k satisfies the quasi-Newton equation $H_k y_{k-1} = s_{k-1}$.

For general nonlinear function f, it holds that

$$f(x) \approx f_k + g_k^T(x - x_k) + \frac{1}{2}(x - x_k)^T G_k(x - x_k).$$

Take derivatives on both sides, obtain

$$\nabla f(x) \approx g_k + G_k(x - x_k).$$

Set $x = x_{k-1}$, we get

$$G_k s_{k-1} \approx y_{k-1} \quad (\text{or } G_k^{-1} y_{k-1} \approx s_{k-1}).$$

If f is a quadratic function, then it holds that

$$G_k s_{k-1} \equiv y_{k-1} \quad (\text{or } G_k^{-1} y_{k-1} \equiv s_{k-1}).$$

Thus, the condition $B_k s_{k-1} = y_{k-1}$ (or $H_k y_{k-1} = s_{k-1}$) imposed by quasi-Newton equation is reasonable.

Outline

Framework of quasi-Newton methods

Algorithm (quasi-Newton methods)

1. *Choose* $x_0 \in R^n$, $B_0 \in R^{n\times n}$ *(or* $H_0 \in R^{n\times n}$*) and* $\epsilon > 0$, *set* $k = 0$;
2. *If* $\|g_k\| \le \epsilon$, *stop, else go ahead;*
3. *Solve* $B_k d = -g_k$ *for* d_k *(or compute* $d_k = -H_k g_k$*);*
4. *Do line search along* d_k, *i.e., determine* $\alpha_k > 0$ *such that*

$$x_{k+1} = x_k + \alpha_k d_k$$

   satisfies certain conditions.

5. *Update* B_k *(resp.* H_k*) to generate* B_{k+1} *(resp.* H_{k+1}*) such that the quasi-Newton equation* $B_{k+1}s_k = y_k$ *(resp.* $H_{k+1}y_k = s_k$*) is satisfied;*
6. *Set* $k = k+1$, *repeat.*

Advantages of quasi-Newton methods

1. Only need to compute first-order derivatives;
2. If B_k (resp. H_k) is always positive definite, then the search direction

$$d_k = -B_k^{-1} g_k \quad (\text{resp. } d_k = -H_k g_k)$$

   is always a descent direction;
3. If we approximate G_k^{-1} directly by H_k, the search direction

$$d_k = -H_k g_k$$

   can be computed by matrix-vector multiplication and no need to solve linear systems.

Desirable properties of B_k

- B_k satisfies the quasi-Newton equation: $B_k s_{k-1} = y_{k-1}$;
- B_k is symmetric since Hessian matrix is always so;
- B_k is positive definite so that $q_k(x)$ has a unique minimizer, and the search direction

$$d_k = -B_k^{-1} g_k \quad \text{(quasi-Newton direction)}$$

  is a descent direction since $g_k^T d_k < 0$ (unless $g_k = 0$);
- Similar properties are desirable for H_k.

Note that the above conditions are not sufficient to uniquely define B_k (and H_k) since the degree of freedom is much greater than the number of constraints.

Outline

Motivation of quasi-Newton methods

Quasi-Newton equation

Framework of quasi-Newton methods

DFP and BFGS updates

Convergence results

L-BFGS

BB gradient method

DFP update of H_k

Suppose we approximate G_k^{-1} directly by H_k. At the beginning, we provide a positive definite H_0. After k iterations, we already have the following information

$$x_0, x_1, \ldots, x_{k-1}, x_k, g_0, g_1, \ldots, g_{k-1}, g_k, H_0, H_1, \ldots, H_{k-1}.$$

Now, based on known information we construct H_k such that

$$H_k y_{k-1} = s_{k-1},$$

where $y_{k-1} = g_k - g_{k-1}$ and $s_{k-1} = x_k - x_{k-1}$.

Intuitively, it is better to have H_k not too far away from H_{k-1}. Thus, we construct H_k based on H_{k-1} and consider the following rank-two update

$$H_k = H_{k-1} + auu^T + bvv^T,$$

where $a, b \in R$ and $u, v \in R^n$.

It follows from $H_k y_{k-1} = s_{k-1}$ that

$$(au^T y_{k-1})u + (bv^T y_{k-1})v = s_{k-1} - H_{k-1} y_{k-1}.$$

An obvious choice of a, b, u and v is

$$\begin{aligned} u = s_{k-1}, &\quad au^T y_{k-1} = 1, \\ v = H_{k-1} y_{k-1}, &\quad bv^T y_{k-1} = -1, \end{aligned}$$

resulting the following updating formula

$$H_k = H_{k-1} + \frac{s_{k-1} s_{k-1}^T}{s_{k-1}^T y_{k-1}} - \frac{H_{k-1} y_{k-1} y_{k-1}^T H_{k-1}}{y_{k-1}^T H_{k-1} y_{k-1}}. \qquad \text{(DFP-H)}$$

- This formula was first proposed by Davidon (1959) and then popularized by Fletcher and Powell (1963). It is now widely known as the DFP formula.
- DFP method is the first quasi-Newton method.

Positive definiteness

Theorem

Suppose that H_{k-1} is positive definite. Then, H_k given by the DFP-H formula is positive definite if and only if $s_{k-1}^T y_{k-1} > 0$.

Necessity.

Suppose H_k is positive definite (thus $y_{k-1} \neq 0$, otherwise H_k undefined). Then, it follows from $H_k y_{k-1} = s_{k-1}$ that

$$s_{k-1}^T y_{k-1} = y_{k-1}^T H_k y_{k-1} > 0.$$

□

Sufficiency.

Since H_{k-1} is positive definite, there exists nonsingular R such that $H_{k-1} = R^T R$. For any $0 \neq z \in R^n$, it holds that

$$z^T H_k z = \|u\|^2 + \frac{(s_{k-1}^T z)^2}{s_{k-1}^T y_{k-1}} - \frac{(u^T v)^2}{\|v\|^2},$$

where $u = Rz$ and $v = Ry_{k-1}$. Clearly,

$$\frac{(s_{k-1}^T z)^2}{s_{k-1}^T y_{k-1}} \geq 0$$

since $s_{k-1}^T y_{k-1} > 0$ is the condition of sufficiency. From Cauchy-Schwartz inequality, it holds that

$$\|u\|^2 - \frac{(u^T v)^2}{\|v\|^2} \geq 0,$$

and equality holds iff u and v are parallel. Since R is nonsingular, this holds iff there exists $\beta \neq 0$ such that $z = \beta y_{k-1}$, in which case it is easy to verify that

$$\frac{(s_{k-1}^T z)^2}{s_{k-1}^T y_{k-1}} = \beta^2 s_{k-1}^T y_{k-1} > 0.$$

In all, $z^T H_k z > 0$ and thus H_k is positive definite. $\square$

About condition $s_k^T y_k > 0$

The condition $s_k^T y_k > 0$ is quite easy to be satisfied.

- For positive definite quadratic function, it holds that $s_k^T y_k = s_k^T G s_k > 0$ (unless $s_k = 0$, in which case g_k is already zero).
- For general nonlinear function, it holds that

$$s_k^T y_k = g_{k+1}^T s_k - g_k^T s_k.$$

  - For exact line search, $g_{k+1}^T s_k = 0$ and thus

$$s_k^T y_k = -g_k^T s_k > 0$$

    since s_k is a descent direction.
  - For inexact line search, a condition $|g_{k+1}^T d_k| \leq \sigma |g_k^T d_k|$ can be enforced (such as in strong Wolfe-Powell line search), where $0 < \sigma < 1$. Thus,

$$s_k^T y_k = g_{k+1}^T s_k - g_k^T s_k \geq -(1-\sigma) g_k^T s_k > 0.$$

Properties of DFP quasi-Newton method

If f is a quadratic function and exact line search rule is used:

- Quadratic termination: $H_n = G^{-1}$, and, no matter where x_0 is, the DFP method will find exact solution in n steps.
- $H_i y_k = s_k$ for all $k < i$.
- If $H_0 = I$, the DFP quasi-Newton method reduces to the conjugate gradient method.

For general nonlinear function:

- Positive definiteness of H_k can be maintained, and the search direction is always descent.
- At each iteration, the computation of d_k costs $O(n^2)$ if using H_k and $O(n^3)$ if using B_k.
- With exact line search, DFP quasi-Newton method converges globally for convex functions.
- Local supper-linear convergence rate.

Sherman-Morrison formula

Theorem (Sherman-Morrison)

Let $A \in R^{n \times n}$ be nonsingular and $u, v \in R^n$. If $1 + v^T A^{-1} u \neq 0$, then $A + uv^T$ is nonsingular, and

$$\left(A + uv^T\right)^{-1} = A^{-1} - \frac{A^{-1} uv^T A^{-1}}{1 + v^T A^{-1} u}.$$

Theorem (Sherman-Morrison-Woodburg)

Let $A \in R^{n \times n}$ be nonsingular and $U, V \in R^{n \times m}$. If $I + V^T A^{-1} U$ is nonsingular, then $A + UV^T$ is nonsingular, and

$$\left(A + UV^T\right)^{-1} = A^{-1} - A^{-1} U \left(I + V^T A^{-1} U\right)^{-1} V^T A^{-1}.$$

DFP formula of B

Recall that the DFP formula for H_{k+1} is

$$H_{k+1}^{(DFP)} = H_k + \frac{s_k s_k^T}{s_k^T y_k} - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k}. \tag{DFP-H}$$

Let $B_{k+1} = H_{k+1}^{-1}$, which approximates G_{k+1}. By utilizing the Sherman-Morrison theorem twice, we get

$$B_{k+1}^{(DFP)} = B_k + \left(1 + \frac{s_k^T B_k s_k}{s_k^T y_k}\right) \frac{y_k y_k^T}{s_k^T y_k} - \frac{B_k s_k y_k^T + y_k s_k^T B_k}{s_k^T y_k}. \tag{DFP-B}$$

This is the DFP formula for updating B.

BFGS formulas

Note that the quasi-Newton equations are

$$B_{k+1}s_k = y_k \quad \text{and} \quad H_{k+1}y_k = s_k.$$

By interchanging $B_{k+1} \leftrightarrow H_{k+1}$ and $s_k \leftrightarrow y_k$ in either one, we obtain the other. Apply these interchanges to the (DFP-H) formula, we get

$$B_{k+1}^{(BFGS)} = B_k + \frac{y_k y_k^T}{s_k^T y_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k}. \tag{BFGS-B}$$

By utilizing the Sherman-Morrison theorem twice to (BFGS-B), or apply interchanges to the (DFP-B) formula, we get

$$H_{k+1}^{(BFGS)} = H_k + \left(1 + \frac{y_k^T H_k y_k}{s_k^T y_k}\right) \frac{s_k s_k^T}{s_k^T y_k} - \frac{H_k y_k s_k^T + s_k y_k^T H_k}{s_k^T y_k}. \tag{BFGS-H}$$

(independently studied by Broyden, Fletcher, Goldfarb and Shanno, four papers all published in 1970. Till far, the best quasi-Newton method in practice.)

Other quasi-Newton formulas

- Symmetric rank-one formula (SR1):

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^T}{(s_k - H_k y_k)^T y_k}.$$

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}.$$

- Powell's symmetric Broyden formula (PSB):

$$\begin{aligned} B_{k+1} = B_k \quad &+ \quad \frac{(y_k - B_k s_k)c_k^T + c_k(y_k - B_k s_k)^T}{c_k^T s_k} \\ &- \quad \frac{(y_k - B_k s_k)^T s_k}{(c_k^T s_k)^2} c_k c_k^T. \end{aligned}$$

By letting $c_k = y_k - B_k s_k, y_k, s_k, \ldots$ and interchanging $H \leftrightarrow B$ and $s \leftrightarrow y$, we can recover old formulas and generate new ones. Thus, this formula is important both in theory and in practice.

Outline

Convergence analysis [1]

The fact that the Hessian approximations evolve by means of updating formulas makes the analysis of quasi-Newton methods much more complex than that of steepest descent and Newton's method.

Although the BFGS and SR1 methods are known to be remarkably robust in practice, we will not be able to establish truly global convergence results for general nonlinear objective functions. That is, we cannot prove that the iterates of these quasi-Newton methods approach a stationary point of the problem from any starting point and any (suitable) initial Hessian approximation. In fact, it is not yet known if the algorithms enjoy such properties.

In our analysis we will either assume that the objective function is convex or that the iterates satisfy certain properties. On the other hand, there are well known local, super linear convergence results that are true under reasonable assumptions.

---

[1] copied from Nocedal and Wright's *Numerical optimization* book.

Global convergence of the BFGS method

Assumptions

1. *The objective function f is twice continuously differentiable.*
2. *The level set $\mathcal{L} = \{x \in R^n : f(x) \leq f(x_0)\}$ is convex, and there exist positive constants m and M such that*

$$m\|z\|^2 \leq z^T G(x) z \leq M\|z\|^2$$

   for all $z \in R^n$ and $x \in \mathcal{L}$.

The second part of this assumption implies that $G(x)$ is positive definite on $\mathcal{L}$ and that f has a unique minimizer $x^* \in \mathcal{L}$.

Wolfe line search: step size $\alpha_k > 0$ satisfies

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + c_1 \alpha_k g_k^T d_k \\ \nabla f(x_k + \alpha_k d_k)^T d_k &\geq c_2 g_k^T d_k, \end{aligned}$$

where $0 < c_1 < c_2 < 1$.

Theorem (Global convergence of the BFGS method[2])

Let B_0 be any symmetric positive definite initial matrix, and let x_0 be a starting point for which the assumed conditions are satisfied. At each iteration, the step size α_k satisfies the Wolfe line search condition. Then the sequence $\{x_k\}$ generated by the BFGS method converges to the minimizer x^* of f.

Remark

This theorem has been generalized to the entire restricted Broyden class, except for the DFP method, i.e., convergence for

$$B_{k+1} = \theta B_k^{DFP} + (1-\theta) B_k^{BFGS},$$

where $\theta \in [0, 1)$.

[2] Theorem 6.5 in Nocedal and Wright's *Numerical Optimization* book.

Local convergence of the BFGS method

Assumption

The Hessian matrix G is Lipschitz continuous at x^*, i.e.,

$$\|G(x) - G(x^*)\| \leq L\|x - x^*\|,$$

for all x near x^*, where $L > 0$ is a constant.

Theorem (Local convergence of the BFGS method[3])

Suppose that $f \in C^2$ and that the iterates generated by the BFGS method converge to a minimizer x^* at which the above assumption holds. Suppose also that $\sum_{k=1}^{\infty} \|x_k - x^*\| < \infty$. Then $\{x_k\}$ converges to x^* at a super linear rate, i.e.,

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0.$$

[3] Theorem 6.6 in Nocedal and Wright's *Numerical Optimization* book.

Outline

Limited memory BFGS method[4]

- Limited-memory quasi-Newton methods are useful for solving large problems whose Hessian matrices cannot be computed at a reasonable cost or are not sparse.
- These methods maintain simple and compact approximations of Hessian matrices: Instead of storing fully dense $n \times n$ approximations, they save only a few vectors of length n that represent the approximations implicitly.
- Despite these modest storage requirements, they often yield an acceptable rate of convergence.
- Various limited-memory methods have been proposed; we focus mainly on L-BFGS, which, as its name suggests, is based on the BFGS updating formula.
- The main idea of this method is to use curvature information from only the most recent iterations to construct the Hessian approximation.
- Curvature information from earlier iterations, which is less likely to be relevant to the actual behavior of the Hessian at the current iteration, is discarded in the interest of saving storage.

[4] Texts copied from Nocedal and Wright's book.

The BFGS formula for H can be rewritten as

$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T,$$

where $\rho_k = \frac{1}{s_k^T y_k}$ and $V_k = I - \rho_k y_k s_k^T$. Thus,

$$\begin{aligned}
H_k &= (V_{k-1}^T \dots V_{k-m}^T) H_k^0 (V_{k-m} \dots V_{k-1}) \\
&+ \rho_{k-m} (V_{k-1}^T \dots V_{k-m+1}^T) s_{k-m} s_{k-m}^T (V_{k-m+1} \dots V_{k-1}) \\
&+ \rho_{k-m+1} (V_{k-1}^T \dots V_{k-m+2}^T) s_{k-m+1} s_{k-m+1}^T (V_{k-m+2} \dots V_{k-1}) \\
&+ \dots \\
&+ \rho_{k-2} V_{k-1}^T s_{k-2} s_{k-2}^T V_{k-1} \\
&+ \rho_{k-1} s_{k-1} s_{k-1}^T.
\end{aligned}$$

Clearly, we can compute $d_k = -H_k g_k$ without explicitly storing H_k. What we need to store is

$$\{(s_i, y_i) : i = k-1, k-2, \dots, k-m\}.$$

The user can determine how large m is.

Algorithm (Computing $H_k g_k$)

1. $q \leftarrow g_k$;
2. For $i = k-1, k-2, \ldots, k-m$, do

$$\alpha_i \leftarrow \rho_i s_i^T q, \quad q \leftarrow q - \alpha_i y_i;$$

3. Compute $r = H_k^0 q$;
4. For $i = k-m, k-m+1, \ldots, k-1$, do

$$\beta \leftarrow \rho_i y_i^T r, \quad r \leftarrow r + s_i(\alpha_i - \beta);$$

5. Stop with $r = H_k g_k$.

Outline

BB gradient method

Consider minimizing a quadratic function

$$f(x) = \frac{1}{2}x^T Ax - b^T x, \quad A \succ 0,$$

by gradient method

$$x_{k+1} = x_k - \alpha_k g_k = x_k - D_k g_k,$$

where $D_k = \alpha_k I$. Note that D_k plays the role of B_k^{-1} (or H_k).

The basic idea of BB method is to choose α_k such that D_k approximately satisfies the quasi-Newton equation.

- Choose α_k such that $\|D_k^{-1} s_{k-1} - y_{k-1}\|$ is minimized over $\alpha_k \in R$, which gives the first BB step length formula:

$$\alpha_k^{BB1} = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}} = \frac{g_{k-1}^T g_{k-1}}{g_{k-1}^T A g_{k-1}}.$$

- Choose α_k such that $\|s_{k-1} - D_k y_{k-1}\|$ is minimized over $\alpha_k \in R$, which gives the second BB step length formula:

$$\alpha_k^{BB2} = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}} = \frac{g_{k-1}^T A g_{k-1}}{g_{k-1}^T A^2 g_{k-1}}.$$

Note that

$$\frac{1}{\lambda_{\max}(A)} \leq \alpha_k^*, \alpha_k^{BB1}, \alpha_k^{BB2} \leq \frac{1}{\lambda_{\min}(A)},$$

where $\alpha_k^* = \frac{g_k^T g_k}{g_k^T A g_k}$ is “the best step length" at kth iteration.