Globalizing Newton's method: Descent Directions (II)

Mark S. Gockenbach

1 Introduction

The reader will recall that, if H_k is *any* symmetric positive definite matrix, then

$$p = -H_k^{-1}\nabla f(x^{(k)})$$

is a descent direction for f at $x^{(k)}$. Previously I discussed one method for choosing H_k: Use $H_k = \nabla^2 f(x^{(k)})$ if $\nabla^2 f(x^{(k)})$ is positive definite; otherwise, use $H_k = \nabla^2 f(x^{(k)}) + E_k$ where E_k is chosen to make H_k positive definite.

In many problems, it is expensive to compute $\nabla^2 f(x^{(k)})$, so it is desirable to use an approximation to the Hessian that can be computed in a reasonable time. The most popular method for producing such an approximation is the so-called *secant* method. To explain the secant idea, I will suppose that I have a symmetric positive definite approximation H_k of $\nabla^2 f(x^{(k)})$ and that I take a step from $x^{(k)}$ to produce $x^{(k+1)}$:

$$x^{(k+1)} = x^{(k)} - \alpha_k H_k^{-1}\nabla f(x^{(k)}).$$

To take the next step, I will have to compute $\nabla f(x^{(k+1)})$, and I want to use $x^{(k)}$, $x^{(k+1)}$, $\nabla f(x^{(k)})$, $\nabla f(x^{(k+1)})$ (and H_k) to produce H_{k+1}. Since the Hessian is the Jacobian of the gradient, I can write

$$\nabla f(x^{(k+1)}) = \nabla f(x^{(k)}) + \nabla^2 f(x^{(k)})(x^{(k+1)} - x^{(k)}) + o\left(\|x^{(k+1)} - x^{(k)}\|\right)$$

or

$$\nabla f(x^{(k)}) = \nabla f(x^{(k+1)}) + \nabla^2 f(x^{(k+1)})(x^{(k)} - x^{(k+1)}) + o\left(\|x^{(k)} - x^{(k+1)}\|\right).$$

Writing $s^{(k)} = x^{(k+1)} - x^{(k)}$ and $y^{(k)} = \nabla f(x^{(k+1)}) - \nabla f(x^{(k)})$, the two relationships are

$$y^{(k)} = \nabla^2 f(x^{(k)})s^{(k)} + o\left(\|s^{(k)}\|\right)$$

and

$$-y^{(k)} = \nabla^2 f(x^{(k+1)})(-s^{(k)}) + o\left(\|s^{(k)}\|\right).$$

The goal is for H_k and H_{k+1} to give some information about the curvature of the objective function f. The change in the gradients from $x^{(k)}$ to $x^{(k+1)}$ provides some information about this curvature. It would be desirable for H_k and H_{k+1} to satisfy

$$H_k s^{(k)} = y^{(k)}, \; H_{k+1}s^{(k)} = y^{(k)}$$

(I drop the error terms since I do not have any information about them, and just hope that they are small.) Now, H_k has already been determined before I can compute $x^{(k+1)}$, and therefore there is no way to ensure that $H_k s^{(k)} = y^{(k)}$ holds. However, I can *choose* H_{k+1} to satisfy $H_{k+1}s^{(k)} = y^{(k)}$.

This, then, is the secant method: Given a symmetric positive definite matrix H_k, $s^{(k)} = x^{(k+1)} - x^{(k)}$, $y^{(k)} = \nabla f(x^{(k+1)}) - \nabla f(x^{(k)})$, choose H_{k+1} to be a symmetric positive definite matrix satisfying the *secant equation*

$$H_{k+1}s^{(k)} = y^{(k)}. \tag{1}$$

The matrix H_{k+1} will be used as the approximation to $\nabla^2 f(x^{(k+1)})$. Since H_{k+1} is an $n \times n$ symmetric matrix, it is determined by $(n^2+n)/2$ entries. The secant equation (1) only consists of n equations and so does not begin to define all $(n^2+n)/2$ entries of H_{k+1}. The requirement that H_{k+1} be positive definite imposes some more constraints, but only n more. The result is that there is not a unique secant approximation to the Hessian.

In selecting a matrix to satisfy (1), it would not be a good idea to throw away the curvature information already contained in H_k. A sensible choice would be to choose H_{k+1} to be the symmetric positive definite matrix that is as close to H_k as possible, subject to the contraint that H_{k+1} satisfy (1). This in itself does not uniquely determine H_{k+1}, since there are various ways to define "as close as possible." To develop this idea, it is easiest[1] to begin with the simpler case of updating a Jacobian approximation, which is useful anyway for solving nonlinear systems. As I will now explain, secant approximations to the Jacobian involve many of the same issues, but they are simpler because a Jacobian approximation need not be symmetric or positive definite.

2 A secant update for the Jacobian

In solving $F(x)=0$, where $F:\mathrm{R}^n \to \mathrm{R}^n$, one uses $x^{(k)}$, $F(x^{(k)})$, and $J(x^{(k)})$, or an approximation to it, to produce $x^{(k+1)}$. I assume that I have a nonsingular matrix A_k approximating $J(x^{(k)})$ and I wish to produce an approximation A_{k+1} to $J(x^{(k+1)})$. Since

$$F(x^{(k)}) \doteq F(x^{(k+1)}) + J(x^{(k+1)})(x^{(k)} - x^{(k+1)})$$

should hold, the secant equation will be

$$A_{k+1}s^{(k)} = y^{(k)}, \tag{2}$$

where

$$s^{(k)} = x^{(k+1)} - x^{(k)},\ y^{(k)} = F(x^{(k+1)}) - F(x^{(k)}).$$

Since I have new information only in the direction $s^{(k)}$, I require that A_k and A_{k+1} agree on all vectors orthogonal to $s^{(k)}$:

$$p \cdot s^{(k)} = 0 \ \Rightarrow\ A_{k+1}p = A_k p. \tag{3}$$

I now wish to use (2) and (3) to choose A_{k+1}.

Condition (2) implies that the matrix $A_{k+1} - A_k$ has a null space of dimension $n-1$ (there are $n-1$ independent vectors orthogonal to $s^{(k)}$). Therefore, the Fundamental Theorem of Linear Algebra[2] implies that the rank of $A_{k+1} - A_k$ must be 1. Therefore, every column of $A_{k+1} - A_k$ must be a multiple of a common vector u:

$$A_{k+1} - A_k = [v_1 u | v_2 u | \cdots | v_n u].$$

The matrix on the right can be written as uv^T, where u and v are vectors, regarded as $n \times 1$ matrices. The reader should verify the following fundamental formula:

$$uv^T x = (v \cdot x)u \text{ for all } x \in \mathrm{R}^n.$$

Thus A_{k+1} is to be chosen as a *rank-one update* of A_k:

$$A_{k+1} = A_k + uv^T. \tag{4}$$

[1] However, no derivation that I have encountered is very straightforward. The approach I take is found in Dennis and Schnabel [1], an excellent reference for secant methods.

[2] The Fundamental Theorem of Linear Algebra states, in part, that the rank of any matrix plus the dimension of its null space must equal the number of columns.

I can now determine u and v. Conditions (3) and (4) imply that

$$(uv^T)p = 0 \text{ for all } p \text{ such that } p \cdot s^{(k)} = 0,$$

or, equivalently,

$$(v \cdot p)u = 0 \text{ for all } p \text{ such that } p \cdot s^{(k)} = 0.$$

This implies that

$$v \cdot p = 0 \text{ for all } p \text{ such that } p \cdot s^{(k)} = 0.$$

But the only vector v satisfying this condition is $s^{(k)}$ itself (or a multiple of it). Therefore, I take $v = s^{(k)}$ and use (2) to determine u:

$$\begin{aligned} A_{k+1}s^{(k)} = y^{(k)} \quad &\Rightarrow \quad A_k s^{(k)} + (s^{(k)} \cdot s^{(k)})u = y^{(k)} \\ &\Rightarrow \quad (s^{(k)} \cdot s^{(k)})u = y^{(k)} - A_k s^{(k)} \\ &\Rightarrow \quad u = \frac{y^{(k)} - A_k s^{(k)}}{s^{(k)} \cdot s^{(k)}}. \end{aligned}$$

Therefore,

$$A_{k+1} = A_k + \frac{\left(y^{(k)} - A_k s^{(k)}\right)(s^{(k)})^T}{s^{(k)} \cdot s^{(k)}}. \tag{5}$$

Equation (5) is referred to as *Broyden's update*.

Normally when using Broyden's update, the initial Jacobian estimate is taken to be $J(x^{(0)})$ or a finite-difference estimate of it.[3] It can be shown that, under certain conditions, the local convergence of Broyden's method for solving $F(x) = 0$ is superlinear. This is not as fast as Newton's method, which converges quadratically; however, since Broyden's method can use much less time per iteration by avoiding the computation of $J(x^{(k)})$, it is more efficient than Newton's method on some problems.

I will give more details about Broyden's method for solving $F(x) = 0$ later.

3 A secant update for the Hessian

I now return to the question of finding a secant update for the Hessian that preserves symmetry and positive definiteness. That is, given a symmetric and positive definite matrix H_k and

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

I want to find a symmetric positive definite matrix H_{k+1} that satisfies

$$H_{k+1}s^{(k)} = y^{(k)}. \tag{6}$$

Such an H_{k+1} does not necessarily exist. If (6) holds, where H_{k+1} is positive definite, then

$$s^{(k)} \cdot y^{(k)} = s^{(k)} \cdot H_{k+1}s^{(k)} > 0.$$

Therefore, if $s^{(k)} \cdot y^{(k)} > 0$ fails, it will be impossible to find H_{k+1}. The condition $s^{(k)} \cdot y^{(k)} > 0$ is equivalent to

$$\left(\nabla f(x^{(k+1)}) - \nabla f(x^{(k)})\right) \cdot \left(x^{(k+1)} - x^{(k)}\right) > 0. \tag{7}$$

A strictly convex function f would satisfy (7) for *any* points $x^{(k)}$ and $x^{(k+1)}$. The condition (7) therefore means that $\nabla f(x^{(k)})$ and $\nabla f(x^{(k+1)})$ are consistent with f's having positive curvature on the line segment between $x^{(k)}$ and $x^{(k+1)}$. If this condition were to fail, it would be impossible to find a positive definite matrix H_{k+1} satisfying the secant equation.

[3] I will explain how to compute finite-difference estimates of the Jacobian later.

Next I wish to point out the following: No rank-one update can produce H_{k+1} that is symmetric, positive definite, and satisfies the secant equation, at least not in every case. Indeed, to preserve symmetry and positive definiteness, a rank-one update would have to take the form

$$H_{k+1} = H_k + uu^T.$$

However, it is easy to show that, in some cases (even when $s^{(k)} \cdot y^{(k)} > 0$ holds), no vector u causes H_{k+1} to satisfy the secant equation.[4]

Since a rank one update cannot be found, I will try a different approach. To simplify the notation in the following derivation, I will write $H = H_k$, $H_+ = H_{k+1}$, $s = s^{(k)}$, and $y = y^{(k)}$. The positive definite matrix H has a Cholesky factorization $H = LL^T$. I will look for H_+ in the form $H_+ = JJ^T$, where J is nonsingular and close to L in some sense. The reader should notice that, if J is nonsingular, then JJ^T is necessarily symmetric and positive definite.

If H_+ is to satisfy the secant equation, then

$$JJ^T s = y$$

must hold. The matrix J will be found by a two-step process:

1. Given any vector v, choose J so that $Jv = y$ and J is as close as possible to L. Broyden's method shows how to do this:
$$J = L + \frac{(y - Lv)\,v^T}{v \cdot v}. \tag{8}$$

2. Choose v so that $J^T s = v$ also holds (then JJ^T will satisfy the secant equation (6)). Using (8),
$$\begin{aligned} J^T s = v \quad &\Rightarrow \quad L^T s + \frac{v(y - Lv)^T s}{v \cdot v} = v \\ &\Rightarrow \quad L^T s + \frac{(y \cdot s)v - vv^T L^T s}{v \cdot v} = v \\ &\Rightarrow \quad L^T s + \frac{(y \cdot s)v - (v \cdot L^T s)v}{v \cdot v} = v \\ &\Rightarrow \quad L^T s = \left(1 - \frac{y \cdot s}{v \cdot v} + \frac{v \cdot L^T s}{v \cdot v}\right) v. \end{aligned}$$

   This last equation is quite nonlinear in v, and to solve it directly would be difficult or impossible. However, it shows that the vectors v and $L^T s$ must point in the same direction. Therefore, there exists α such that
$$v = \alpha L^T s.$$

   Substituting this formula for v into (8) and simplifying yields
$$J = L + \frac{ys^T L}{\alpha s \cdot Hs} - \frac{Hss^T L}{s \cdot Hs}$$

   (to arrive at this formula for J, I used the fact that $H = LL^T$).

   I now use the equation $J^T s = v$, where $v = \alpha L^T s$, to solve for α:
$$\begin{aligned} J^T s = v \quad &\Rightarrow \quad L^T s + \frac{L^T sy^T s}{\alpha s \cdot Hs} - \frac{L^T ss^T LL^T s}{s \cdot Hs} = \alpha L^T s \\ &\Rightarrow \quad L^T s + \frac{y \cdot s}{\alpha s \cdot Hs} L^T s - \frac{s \cdot LL^T s}{s \cdot Hs} L^T s = \alpha L^T s \end{aligned}$$

[4]The proof is left as an exercise.

$$\Rightarrow \quad L^T s + \frac{y \cdot s}{\alpha s \cdot Hs} L^T s - L^T s = \alpha L^T s$$
$$\Rightarrow \quad \frac{y \cdot s}{\alpha s \cdot Hs} L^T s = \alpha L^T s$$
$$\Rightarrow \quad \alpha^2 = \frac{y \cdot s}{s \cdot Hs}.$$

There are two solutions for α; the positive solution is the correct one to take, since then $J = L$ if H already satisfies the secant equation. (The reader should notice that the requirement that $s \cdot y$ be positive appears here.)

Therefore, if

$$J = L + \frac{ys^T L}{\alpha s \cdot Hs} - \frac{Hss^T L}{s \cdot Hs}, \tag{9}$$

where

$$\alpha = \sqrt{\frac{y \cdot s}{s \cdot Hs}},$$

then $H_+ = JJ^T$ is symmetric, positive definite, and satisfies

$$H_+ s = y.$$

Since J, as given by (9), is not necessarily lower triangular (even though L is), it is more convenient to express the update in terms of H and H_+ rather than in terms of L and J. A tedious calculation, which simplifies nicely in the end, shows that

$$JJ^T = H - \frac{Hss^T H}{s \cdot Hs} + \frac{yy^T}{y \cdot s}.$$

The resulting update is known as the *BFGS update*:[5]

$$H_{k+1} = H_k - \frac{H_k s^{(k)} (s^{(k)})^T H_k}{s^{(k)} \cdot H_k s^{(k)}} + \frac{y^{(k)} (y^{(k)})^T}{y^{(k)} \cdot s^{(k)}}. \tag{10}$$

Note: The above derivation does not guarantee that J (and hence H_+) is nonsingular. However, I can prove directly that (9) defines a nonsingular matrix J using the following theorem:

Theorem 3.1 (Sherman-Morrison-Woodbury) *Suppose $A \in \mathrm{R}^{n \times n}$ is nonsingular and $U, V \in \mathrm{R}^{n \times p}$ are such that*

$$I + V^T A^{-1} U \in \mathrm{R}^{p \times p}$$

is nonsingular. Then

$$B = A + UV^T$$

is nonsingular and

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

(Note: The rank of UV^T is easily shown to be p or less.)

[5]The formula is due to Broyden, Fletcher, Goldfarb, and Shanno.

Proof: The proof is a direct computation:

$$\begin{aligned}
& \left(A+UV^T\right)\left(A^{-1}-A^{-1}U\left(I+V^TA^{-1}U\right)^{-1}V^TA^{-1}\right) \\
= \; & I+UV^TA^{-1}-U\left(I+V^TA^{-1}U\right)^{-1}V^TA^{-1}-UV^TA^{-1}U\left(I+V^TA^{-1}U\right)^{-1}V^TA^{-1} \\
= \; & I+U\left(I-\left(I+V^TA^{-1}U\right)^{-1}-V^TA^{-1}U\left(I+V^TA^{-1}U\right)^{-1}\right)V^TA^{-1} \\
= \; & I+U\left(I+V^TA^{-1}U-I-V^TA^{-1}U\right)\left(I+V^TA^{-1}U\right)^{-1}V^TA^{-1} \\
= \; & I.
\end{aligned}$$

QED

I now apply the Sherman-Morrison-Woodbury formula to the matrix J defined by (9) by defining

$$U=[\delta y|\gamma Hs],\; V=\left[L^Ts|L^Ts\right],$$

where

$$\delta=\frac{1}{\alpha s\cdot Hs}>0,\; \gamma=-\frac{1}{s\cdot Hs}<0.$$

Then $J=L+UV^T$, and, assuming L is nonsingular, J is nonsingular provided

$$I+V^TL^{-1}U$$

is nonsingular. But

$$I+V^TL^{-1}U=\left[\begin{array}{cc} 1+\delta s\cdot y & \gamma s\cdot Hs \\ \delta s\cdot y & 1+\gamma s\cdot Hs \end{array}\right]=\left[\begin{array}{cc} 1+\frac{s\cdot y}{\alpha s\cdot Hs} & -1 \\ \frac{s\cdot y}{\alpha s\cdot Hs} & 0 \end{array}\right],$$

and so

$$\det\left(I+V^TL^{-1}U\right)=\frac{s\cdot y}{\alpha s\cdot Hs}>0.$$

Therefore $I+V^TL^{-1}U$ is nonsingular and hence so is J.

4 Convergence of secant methods

I have now defined two quasi-Newton methods, namely, Broyden's method for nonlinear equations and the BFGS method for unconstrained minimization. These methods can be summarized as follows.

Broyden's method for solving $F(x)=0$ To initialize this algorithm, a starting point $x^{(0)}$ and an estimate A_0 of $J(x^{(0)})$ must be given. The iteration is then

$$\begin{aligned}
x^{(k+1)} &= x^{(k)}-A_k^{-1}F(x^{(k)}), \\
s^{(k)} &= x^{(k+1)}-x^{(k)}, \\
y^{(k)} &= F(x^{(k+1)})-F(x^{(k)}), \\
A_{k+1} &= A_k+\frac{\left(y^{(k)}-A_ks^{(k)}\right)(s^{(k)})^T}{s^{(k)}\cdot s^{(k)}}
\end{aligned}$$

for $k=1,2,3,\ldots$. The initial matrix A_0 is usually taken to be either $J(x^{(0)})$ or a finite-difference approximation to it. The reader should notice that nothing in the development of Broyden's method guarantees that A_k is always nonsingular. Therefore, some enhancements to the algorithm are still necessary.

The BFGS method for minimizing f A starting point $x^{(0)}$ and an estimate of $\nabla^2 f(x^{(0)})$ must be given. The iteration is then

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - H_k^{-1}\nabla f(x^{(k)}), \\ s^{(k)} &= x^{(k+1)} - x^{(k)}, \\ y^{(k)} &= \nabla f(x^{(k+1)}) - \nabla f(x^{(k)}), \\ H_{k+1} &= H_k - \frac{H_k s^{(k)}(s^{(k)})^T H_k}{s^{(k)} \cdot H_k s^{(k)}} + \frac{y^{(k)}(y^{(k)})^T}{y^{(k)} \cdot s^{(k)}} \end{aligned}$$

for $k = 1, 2, 3, \ldots$. The initial matrix is usually taken to be a positive multiple of the identity, which means that the initial search direction will be the steepest descent direction. This algorithm is well-defined provided $s^{(k)} \cdot y^{(k)} > 0$ holds at every step. Although this may not be true for the basic algorithm given above, I will show later that it can be ensured by the use of an appropriate line search.

The advantages of the secant methods are:

1. The user of the algorithm does not need to provide code to compute $J(x)$ or $\nabla^2 f(x)$.
2. In the case of BFGS, the positive definiteness of the Hessian approximation is ensured directly.
3. By using some clever linear algebra, it is possible to update the factors of A_k or H_k rather than the matrices themselves. This means that the cost of generating the quasi-Newton step can be reduced to $O(n^2)$ arithmetic operations per iteration, rather than the usual $O(n^3)$.

The primary disadvantage of the secant methods is that the quadratic convergence of Newton's method is lost.

The convergence analysis for both Broyden's method and the BFGS method is quite complicated compared to that of Newton's method. I will merely summarize it by stating that both methods are locally superlinearly convergent under certain reasonable assumptions. Moreover, this superlinear convergence is typically observed in practice. Although the convergence is not as fast as the quadratic convergence of Newton's method, in many cases the secant methods may still be more efficient (namely, when it is expensive to compute $J(x^{(k)})$ or $\nabla^2 f(x^{(k)})$).

Example 4.1 *Consider the nonlinear system defined by*

$$F(x) = \begin{bmatrix} x_1^2 + x_2^2 - 1 \\ x_2 - x_1^2 \end{bmatrix}.$$

(This example was introduced in the lecture entitled Newton's method.*) Using the starting point* $x^{(0)} = (0.5, 0.5)$*, Broyden's method converges to* $x^* \doteq (0.78615, 0.61803)$ *in 9 iterations. The exact Jacobian* $J(x^{(0)})$ *was used for* A_0 *and thereafter* $A_k \doteq J(x^{(k)})$ *was produced using Broyden's update. The progress of the convergence is displayed in Table 1, which shows that Broyden's method converges more slowly than Newton's method (Newton's method took 5 iterations from the same starting point; also, Table 1 shows that Broyden's method did not converge quadratically on this example). However, Table 1 does suggest that the convergence is superlinear.*

References

[1] J.E. Dennis, Jr. and R.B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, 1983.

k	$\|x^* - x^{(k)}\|$	$\|F(x^{(k)})\|$	$\frac{\|x^* - x^{(k)}\|}{\|x^* - x^{(k)}\|}$
0	$3.0954 \cdot 10^{-1}$	$5.5902 \cdot 10^{-1}$	—
1	$8.9121 \cdot 10^{-2}$	$2.1021 \cdot 10^{-1}$	$2.8792 \cdot 10^{-1}$
2	$2.0134 \cdot 10^{-2}$	$4.3951 \cdot 10^{-2}$	$2.2592 \cdot 10^{-1}$
3	$1.0794 \cdot 10^{-3}$	$2.4072 \cdot 10^{-3}$	$5.3610 \cdot 10^{-2}$
4	$3.7640 \cdot 10^{-5}$	$6.1625 \cdot 10^{-5}$	$3.4872 \cdot 10^{-2}$
5	$3.6986 \cdot 10^{-6}$	$5.8448 \cdot 10^{-6}$	$9.8261 \cdot 10^{-2}$
6	$4.6920 \cdot 10^{-8}$	$7.4315 \cdot 10^{-8}$	$1.2686 \cdot 10^{-2}$
7	$3.2076 \cdot 10^{-11}$	$5.0784 \cdot 10^{-11}$	$6.8364 \cdot 10^{-4}$
8	$3.7975 \cdot 10^{-15}$	$5.9633 \cdot 10^{-15}$	$1.1839 \cdot 10^{-4}$

Table 1: Results of applying Broyden's method to a 2×2 nonlinear system.