Note that indeed $H_2 Q = QH_2 = I_2$, and hence $H_2 = Q^{-1}$. ■

For nonquadratic problems, quasi-Newton algorithms will not usually converge in n steps. As in the case of the conjugate gradient methods, here too some modifications may be necessary to deal with nonquadratic problems. For example, we may reinitialize the direction vector to the negative gradient after every few iterations (e.g., n or $n+1$), and continue until the algorithm satisfies the stopping criterion.

EXERCISES

11.4 11.7 11.6b

11.1 Given $f : \mathbb{R}^n \to \mathbb{R}$, $f \in \mathcal{C}^1$, consider the algorithm

$$x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)},$$

where $d^{(1)}, d^{(2)}, \ldots$ are vectors in $\mathbb{R}^n$, and $\alpha_k \geq 0$ is chosen to minimize $f(x^{(k)} + \alpha d^{(k)})$; that is,

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x^{(k)} + \alpha d^{(k)}).$$

Note that the above general algorithm encompasses almost all algorithms that we discussed in this part, including the steepest descent, Newton, conjugate gradient, and quasi-Newton algorithms.

Let $g^{(k)} = \nabla f(x^{(k)})$, and assume that $d^{(k)T} g^{(k)} < 0$.

a. Show that $d^{(k)}$ is a descent direction for f, in the sense that there exists $\bar{\alpha} > 0$ such that for all $\alpha \in (0, \bar{\alpha}]$,

$$f(x^{(k)} + \alpha d^{(k)}) < f(x^{(k)}).$$

b. Show that $\alpha_k > 0$.

c. Show that $d^{(k)T} g^{(k+1)} = 0$.

d. Show that the following algorithms all satisfy the condition $d^{(k)T} g^{(k)} < 0$, if $g^{(k)} \neq 0$:

1. Steepest descent algorithm;
2. Newton's method, assuming the Hessian is positive definite;
3. Conjugate gradient algorithm;
4. Quasi-Newton algorithm, assuming $H_k > 0$.

e. For the case where $f(x) = \frac{1}{2} x^T Q x - x^T b$, with $Q = Q^T > 0$, derive an expression for α_k in terms of Q, $d^{(k)}$, and $g^{(k)}$.

11.2 Consider the algorithm

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

where $f : \mathbb{R}^2 \to \mathbb{R}$, $f \in \mathcal{C}^1$, $M_k \in \mathbb{R}^{2\times 2}$ is given by

$$M_k = \begin{bmatrix} 1 & 0 \\ 0 & a \end{bmatrix}$$

with $a \in \mathbb{R}$, and

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x^{(k)} - \alpha M_k \nabla f(x^{(k)})).$$

Suppose at some iteration k we have $\nabla f(x^{(k)}) = [1, 1]^T$. Find the largest range of values of a that guarantees that $\alpha_k > 0$ for any f.

11.3 Consider the rank one algorithm. Assume that $H_k > 0$. Show that if $\Delta g^{(k)T}(\Delta x^{(k)} - H_k \Delta g^{(k)}) > 0$, then $H_{k+1} > 0$.

11.4 Based on the rank one update equation, derive an update formula using complementarity and the matrix inverse formula.

11.5 Consider the DFP algorithm applied to the quadratic function

$$f(x) = \frac{1}{2} x^T Q x - x^T b,$$

where $Q = Q^T > 0$.

a. Write down a formula for α_k in terms of Q, $g^{(k)}$, and $d^{(k)}$.

b. Show that if $g^{(k)} \neq 0$, then $\alpha_k > 0$.

11.6 Assuming exact line search, show that if $H_0 = I_n$ ($n \times n$ identity matrix), then the first two steps of the BFGS algorithm yield the same points $x^{(1)}$ and $x^{(2)}$ as conjugate gradient algorithms with the Hestenes-Stiefel, the Polak-Ribière, as well as the Fletcher-Reeves formulas.

11.7 Given a function $f : \mathbb{R}^n \to \mathbb{R}$, consider an algorithm $x^{(k+1)} = x^{(k)} - \alpha_k H_k g^{(k)}$ for finding the minimizer of f, where $g^{(k)} = \nabla f(x^{(k)})$ and $H_k \in \mathbb{R}^{n\times n}$ is symmetric.

Suppose $H_k = \phi H_k^{DFP} + (1-\phi) H_k^{BFGS}$, where $\phi \in \mathbb{R}$, and H_k^{DFP} and H_k^{BFGS} are matrices generated by the DFP and BFGS algorithms, respectively.

a. Show that the above algorithm is a quasi-Newton algorithm. Is the above algorithm a conjugate direction algorithm?

b. Suppose $0 \le \phi \le 1$. Show that if $\boldsymbol{H}_0^{DFP} > 0$ and $\boldsymbol{H}_0^{BFGS} > 0$, then $\boldsymbol{H}_k > 0$ for all k. What can you conclude from this about whether or not the algorithm has the descent property?

11.8 Consider the following simple modification to the quasi-Newton family of algorithms. In the quadratic case, instead of the usual quasi-Newton condition $\boldsymbol{H}_{k+1}\Delta \boldsymbol{g}^{(i)} = \Delta \boldsymbol{x}^{(i)}$, $0 \le i \le k$, suppose that we have $\boldsymbol{H}_{k+1}\Delta \boldsymbol{g}^{(i)} = \rho_i \Delta \boldsymbol{x}^{(i)}$, $0 \le i \le k$, where $\rho_i > 0$. We refer to the set of algorithms that satisfy the above condition as the *symmetric Huang family*.

Show that the symmetric Huang family algorithms are conjugate direction algorithms.

11.9 Write a MATLAB routine to implement the quasi-Newton algorithm for general functions. Use the secant method for the line search (e.g., the MATLAB function of Exercise 7.9). Test the different update formulas for $\boldsymbol{H}_k$ on Rosenbrock's function (see Exercise 9.3), with an initial condition $\boldsymbol{x}^{(0)} = [-2, 2]^T$. For this exercise, reinitialize the update direction to the negative gradient every 6 iterations.

11.10 Consider the function

$$f(\boldsymbol{x}) = \frac{x_1^4}{4} + \frac{x_2^2}{2} - x_1 x_2 + x_1 - x_2.$$

a. Use MATLAB to plot the level sets of f at levels $-0.72, -0.6, -0.2, 0.5, 2$. Locate the minimizers of f from the plots of the level sets.

b. Apply the DFP algorithm to minimize the above function with the following starting initial conditions: (i) $[0, 0]^T$; (ii) $[1.5, 1]^T$. Use $\boldsymbol{H}_0 = \boldsymbol{I}_2$. Does the algorithm converge to the same point for the two initial conditions? If not, explain.

12

Solving $\boldsymbol{Ax} =$

12.1 LEAST-SQUARES ANALYSIS

Consider a system of linear equations

$$\boldsymbol{Ax} = \boldsymbol{b},$$

where $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $m \ge n$, and rank $\boldsymbol{A} = n$. Note that the numbe unknowns, n, is no larger than the number of equations, m. If $\boldsymbol{b}$ does not be to the range of $\boldsymbol{A}$; that is, if $\boldsymbol{b} \notin \mathcal{R}(\boldsymbol{A})$, then this system of equations is said t *inconsistent* or *overdetermined*. In this case, there is no solution to the above se equations. Our goal then is to find the vector (or vectors) $\boldsymbol{x}$ minimizing $\|\boldsymbol{Ax} -$ This problem is a special case of the nonlinear least-squares problem discusse Section 9.4.

Let $\boldsymbol{x}^*$ be a vector that minimizes $\|\boldsymbol{Ax} - \boldsymbol{b}\|^2$; that is, for all $\boldsymbol{x} \in \mathbb{R}^n$,

$$\|\boldsymbol{Ax} - \boldsymbol{b}\|^2 \ge \|\boldsymbol{Ax}^* - \boldsymbol{b}\|^2.$$

We refer to the vector $\boldsymbol{x}^*$ as a *least-squares solution* to $\boldsymbol{Ax} = \boldsymbol{b}$. In the case wl $\boldsymbol{Ax} = \boldsymbol{b}$ has a solution, then the solution is a least-squares solution. Otherwis least-squares solution minimizes the norm of the difference between the left- right-hand sides of the equation $\boldsymbol{Ax} = \boldsymbol{b}$. To characterize least-squares solutions need the following lemma.

Lemma 12.1 *Let* $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $m \ge n$. *Then,* rank $\boldsymbol{A} = n$ *if and onl* rank $\boldsymbol{A}^T\boldsymbol{A} = n$ *(i.e., the square matrix* $\boldsymbol{A}^T\boldsymbol{A}$ *is nonsingular).*