orthogonal directions. Therefore, by Lemma 8.3, for each k, the value of γ_k is one of two numbers, both of which are strictly less than 1. This proves the $n = 2$ case.

For the general n case, let $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ be mutually orthogonal eigenvectors corresponding to $\lambda_{\max}(\boldsymbol{Q})$ and $\lambda_{\min}(\boldsymbol{Q})$. Choose $\boldsymbol{x}^{(0)}$ such that $\boldsymbol{x}^{(0)} - \boldsymbol{x}^* \neq \boldsymbol{0}$ lies in the span of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ but is not equal to either. Note that $\boldsymbol{g}^{(0)} = \boldsymbol{Q}(\boldsymbol{x}^{(0)} - \boldsymbol{x}^*)$ also lies in the span of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, but is not equal to either. By manipulating $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}$ as before, we can write $\boldsymbol{g}^{(k+1)} = (\boldsymbol{I} - \alpha_k \boldsymbol{Q})\boldsymbol{g}^{(k)}$. Any eigenvector of $\boldsymbol{Q}$ is also an eigenvector of $\boldsymbol{I} - \alpha_k \boldsymbol{Q}$. Therefore, $\boldsymbol{g}^{(k)}$ lies in the span of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ for all k; that is, the sequence $\{\boldsymbol{g}^{(k)}\}$ is confined within the 2-dimensional subspace spanned by $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. We can now proceed as in the $n = 2$ case. ∎

In the next chapter, we discuss Newton's method, which has order of convergence at least 2 if the initial guess is near the solution.

EXERCISES

8.5 8.6 8.13 8.14

8.1 Let $\{\boldsymbol{x}^{(k)}\}$ be a sequence that converges to $\boldsymbol{x}^*$. Show that if there exists $c > 0$ such that

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\| \geq c\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p$$

for sufficiently large k, then the order of convergence (if it exists) is at most p.

8.2 Let $\{\boldsymbol{x}^{(k)}\}$ be a sequence that converges to $\boldsymbol{x}^*$. Show that there does not exist $p < 1$ such that

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^p} > 0.$$

8.3 Suppose that we use the Golden Section algorithm to find the minimizer of a function. Let u_k be the uncertainty range at the kth iteration. Find the order of convergence of $\{u_k\}$.

8.4 Suppose that we wish to minimize a function $f : \mathbb{R} \to \mathbb{R}$ that has a derivative f'. A simple line search method, called *derivative descent search* (DDS), is described as follows: given that we are at a point $x^{(k)}$, we move in the direction of the negative derivative with step size α; that is, $x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)})$, where $\alpha > 0$ is a constant.

In the following parts, assume that f is quadratic: $f(x) = \frac{1}{2}ax^2 - bx + c$ (where a, b, and c are constants, and $a > 0$).

a. Write down the value of x^* (in terms of a, b, and c) that minimizes f.

b. Write down the recursive equation for the DDS algorithm explicitly for this quadratic f.

c. Assuming the DDS algorithm converges, show that it converges to the optimal value x^* (found in part a).

d. Find the order of convergence of the algorithm, assuming it does converge.

e. Find the range of values of α for which the algorithm converges (for this particular f) for all starting points $x^{(0)}$.

8.5 Consider the function

$$f(\boldsymbol{x}) = 3(x_1^2 + x_2^2) + 4x_1x_2 + 5x_1 + 6x_2 + 7,$$

where $\boldsymbol{x} = [x_1, x_2]^T \in \mathbb{R}^2$. Suppose we use a fixed step size gradient algorithm to find the minimizer of f:

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

Find the largest range of values of α for which the algorithm is globally convergent.

8.6 Consider the function $f : \mathbb{R}^2 \to \mathbb{R}$ given by

$$f(\boldsymbol{x}) = \frac{3}{2}(x_1^2 + x_2^2) + (1 + a)x_1x_2 - (x_1 + x_2) + b,$$

where a and b are some unknown real-valued parameters.

a. Write the function f in the usual multivariable quadratic form.

b. Find the largest set of values of a and b such that the unique global minimizer of f exists, and write down the minimizer (in terms of the parameters a and b).

c. Consider the following algorithm:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \frac{2}{5}\nabla f(\boldsymbol{x}^{(k)}).$$

Find the largest set of values of a and b for which the above algorithm converges to the global minimizer of f for any initial point $\boldsymbol{x}^{(0)}$.

8.7 Consider the function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = \frac{1}{2}(x - c)^2$, $c \in \mathbb{R}$. We are interested in computing the minimizer of f using the iterative algorithm

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

where f' is the derivative of f and α_k is a step size satisfying $0 < \alpha_k < 1$.

a. Derive a formula relating $f(x_{k+1})$ with $f(x_k)$, involving α_k.

b. Show that the algorithm is globally convergent if and only if

$$\sum_{k=0}^{\infty} \alpha_k = \infty.$$

Hint: Use part a and the fact that for any sequence $\{\alpha_k\} \subset (0,1)$, we have

$$\prod_{k=0}^{\infty}(1-\alpha_k) = 0 \Leftrightarrow \sum_{k=0}^{\infty} \alpha_k = \infty.$$

8.8 Consider the function $f : \mathbb{R} \to \mathbb{R}$ given by $f(x) = x^3 - x$. Suppose we use a fixed step size algorithm $x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)})$ to find a local minimizer of f. Find the largest range of values of α such that the algorithm is locally convergent (i.e., for all $x^{(0)}$ sufficiently close to a local minimizer x^*, we have $x^{(k)} \to x^*$).

8.9 Consider the function f given by $f(x) = (x-1)^2$, $x \in \mathbb{R}$. We are interested in computing the minimizer of f using the iterative algorithm $x_{k+1} = x_k - \alpha 2^{-k} f'(x_k)$, where f' is the derivative of f, and $0 < \alpha < 1$. Does the algorithm have the descent property? Is the algorithm globally convergent?

8.10 Let $f : \mathbb{R} \to \mathbb{R}$, $f \in C^3$, with first derivative f' and second derivative f'', and unique minimizer x^*. Consider a fixed step size gradient algorithm

$$x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)}).$$

Suppose $f''(x^*) \neq 0$ and $\alpha = 1/f''(x^*)$. Assuming the algorithm converges to x^*, show that the order of convergence is at least 2.

8.11 Consider the optimization problem:

$$\text{minimize } \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2,$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $m \geq n$, and $\boldsymbol{b} \in \mathbb{R}^m$.

a. Show that the objective function for the above problem is a quadratic function, and write down the gradient and Hessian of this quadratic.

b. Write down the fixed step size gradient algorithm for solving the above optimization problem.

c. Suppose

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}.$$

Find the largest range of values for α such that the algorithm in part b converges to the solution of the problem.

8.12 Consider a function $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ given by $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$, where $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{b} \in \mathbb{R}^n$. Suppose $\boldsymbol{A}$ is invertible, and $\boldsymbol{x}^*$ is the zero of $\boldsymbol{f}$ (i.e., $\boldsymbol{f}(\boldsymbol{x}^*) = \boldsymbol{0}$). We wish to compute $\boldsymbol{x}^*$ using the iterative algorithm

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha \boldsymbol{f}(\boldsymbol{x}^{(k)}),$$

where $\alpha \in \mathbb{R}$, $\alpha > 0$. We say that the algorithm is *globally monotone* if for any $\boldsymbol{x}^{(0)}$, $\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\| \leq \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|$ for all k.

a. Assume that all the eigenvalues of $\boldsymbol{A}$ are real. Show that a necessary condition for the algorithm above to be *globally monotone* is that all the eigenvalues of $\boldsymbol{A}$ are nonnegative.
Hint: Use contraposition.

b. Suppose

$$\boldsymbol{A} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}.$$

Find the largest range of values of α for which the algorithm is *globally convergent* (i.e., $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^*$ for all $\boldsymbol{x}^{(0)}$).

8.13 Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b}$, where $\boldsymbol{b} \in \mathbb{R}^n$ and $\boldsymbol{Q}$ is a real symmetric positive definite $n \times n$ matrix. Suppose that we apply the steepest descent method to this function, with $\boldsymbol{x}^{(0)} \neq \boldsymbol{Q}^{-1}\boldsymbol{b}$. Show that the method converges in one step, that is, $\boldsymbol{x}^{(1)} = \boldsymbol{Q}^{-1}\boldsymbol{b}$, if and only if $\boldsymbol{x}^{(0)}$ is chosen such that $\boldsymbol{g}^{(0)} = \boldsymbol{Q}\boldsymbol{x}^{(0)} - \boldsymbol{b}$ is an eigenvector of $\boldsymbol{Q}$.

8.14 Suppose we apply a fixed step size gradient algorithm to minimize

$$f(\boldsymbol{x}) = \boldsymbol{x}^T \begin{bmatrix} 3/2 & 2 \\ 0 & 3/2 \end{bmatrix} \boldsymbol{x} + \boldsymbol{x}^T \begin{bmatrix} 3 \\ -1 \end{bmatrix} - 22.$$

a. Find the range of values of the step size for which the algorithm converges to the minimizer.

b. Suppose we use a step size of 1000 (which is too large). Find an initial condition that will cause the algorithm to diverge (not converge).

8.15 Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b}$, where $\boldsymbol{b} \in \mathbb{R}^n$, and $\boldsymbol{Q}$ is a real symmetric positive definite $n \times n$ matrix. Consider the algorithm

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \beta\alpha_k \boldsymbol{g}^{(k)},$$

where $\boldsymbol{g}^{(k)} = \boldsymbol{Q}\boldsymbol{x}^{(k)} - \boldsymbol{b}$, $\alpha_k = \boldsymbol{g}^{(k)T}\boldsymbol{g}^{(k)}/\boldsymbol{g}^{(k)T}\boldsymbol{Q}\boldsymbol{g}^{(k)}$, and $\beta \in \mathbb{R}$ is a given constant. (Note that the above reduces to the steepest descent algorithm if $\beta = 1$.)

Show that $\{x^{(k)}\}$ converges to $x^* = Q^{-1}b$ for any initial condition $x^{(0)}$ if and only if $0 < \beta < 2$.

8.16 Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by $f(x) = \frac{1}{2}x^T Q x - x^T b$, where $b \in \mathbb{R}^n$, and Q is a real symmetric positive definite $n \times n$ matrix. Consider a gradient algorithm

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

where $g^{(k)} = Qx^{(k)} - b$ is the gradient of f at $x^{(k)}$, and α_k is some step size.

Show that the above algorithm has the descent property (i.e., $f(x^{(k+1)}) < f(x^{(k)})$ whenever $g^{(k)} \neq 0$) if and only if $\gamma_k > 0$ for all k.

8.17 Given $f : \mathbb{R}^n \to \mathbb{R}$, consider the general iterative algorithm

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

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

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

Show that for each k, the vector $x^{(k+1)} - x^{(k)}$ is orthogonal to $\nabla f(x^{(k+1)})$ (assuming the gradient exists).

8.18 Write a simple MATLAB routine for implementing the steepest descent algorithm using the secant method for the line search (e.g., the MATLAB function of Exercise 7.9). For the stopping criterion, use the condition $\|g^{(k)}\| \leq \varepsilon$, where $\varepsilon = 10^{-6}$. Test your routine by comparing the output with the numbers in Example 8.1. Also test your routine using an initial condition of $[-4, 5, 1]^T$, and determine the number of iterations required to satisfy the above stopping criterion. Evaluate the objective function at the final point to see how close it is to 0.

8.19 Apply the MATLAB routine from Exercise 8.18 to Rosenbrock's function:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Use an initial condition of $x^{(0)} = [-2, 2]^T$. Terminate the algorithm when the norm of the gradient of f is less than 10^{-4}.

9 Newton's Method

9.1 INTRODUCTION

Recall that the method of steepest descent uses only first derivatives (gradients) in selecting a suitable search direction. This strategy is not always the most effective. If higher derivatives are used, the resulting iterative algorithm may perform better than the steepest descent method. Newton's method (sometimes called the Newton-Raphson method) uses first and second derivatives and indeed does perform better than the steepest descent method if the initial point is close to the minimizer. The idea behind this method is as follows. Given a starting point, we construct a quadratic approximation to the objective function that matches the first and second derivative values at that point. We then minimize the approximate (quadratic) function instead of the original objective function. We use the minimizer of the approximate function as the starting point in the next step and repeat the procedure iteratively. If the objective function is quadratic, then the approximation is exact, and the method yields the true minimizer in one step. If, on the other hand, the objective function is not quadratic, then the approximation will provide only an estimate of the position of the true minimizer. Figure 9.1 illustrates the above idea.

We can obtain a quadratic approximation to the given twice continuously differentiable objection function $f : \mathbb{R}^n \to \mathbb{R}$ using the Taylor series expansion of f about the current point $x^{(k)}$, neglecting terms of order three and higher. We obtain

$$f(x) \approx f(x^{(k)}) + (x - x^{(k)})^T g^{(k)} + \frac{1}{2}(x - x^{(k)})^T F(x^{(k)})(x - x^{(k)}) \triangleq q(x),$$