OR4030 OPTIMIZATION, Chapter 4
Multi-Dimensional Unconstrained Optimization — Descent Methods
Section 4.1 Introduction

4.1.1 Problem Description

Problem:

$$\begin{aligned} \min \quad & f(x), \\ \text{s.t.} \quad & x \in \Re^n, \end{aligned}$$

where

- $f : \Re^n \mapsto \Re$;
- No constraints are placed on the variables x.

4.1.2 Necessary and Sufficient Conditions

First-Order Necessary Condition

Let $f \in C^1(\Re^n)$. We first show that

$$x^* \text{ is a local minimizer } \Rightarrow \nabla f(x^*)^T p \geq 0, \text{ for any } p \in \Re^n.$$

In fact if not, then there is some $\bar{p}$ such that $\nabla f(x^*)^T \bar{p} < 0$. By the mean-value theorem,

$$f(x^* + \epsilon \bar{p}) = f(x^*) + \epsilon \nabla f(\xi)^T \bar{p},$$

where ξ is a point between x^* and $x^* + \epsilon \bar{p}$. When ϵ is sufficiently small, ξ would be very close to x^*. Hence $\nabla f(\xi)^T \bar{p} < 0$.

So, for such ϵ, the point $x = x^* + \epsilon\bar{p}$ would satisfy

$$f(x) < f(x^*),$$

which contradicts the fact that x^* is a local minimizer. We thus proved that

$$\nabla f(x^*)^T p \geq 0$$

for every vector $p \in \Re^n$.

Now take $p = -\nabla f(x^*)$. From the above result it follows that

$$-\|\nabla f(x^*)\|^2 \geq 0 \Rightarrow \|\nabla f(x^*)\| = 0 \Rightarrow \nabla f(x^*) = 0.$$

Therefore, we obtain a conclusion that:

Conclusion 1 Let $f \in C^1(\Re^n)$. If x^* is a local minimizer of f, then $\nabla f(x^*) = 0$.

Note that this necessary condition leads to n equations in n unknowns. A point x^* satisfying this condition is called a *stationary point*. A stationary point is not necessarily a local minimizer. It may be a local minimizer, or a local maximizer, or a saddle point.

Second-Order Necessary Conditions

Conclusion 2 Let $f \in C^2(\Re^n)$. If x^* is a local minimizer of f, then

$$\nabla f(x^*) = 0 \quad \text{and} \quad \nabla^2 f(x^*) \text{ is positive semi-definite.}$$

We again prove the conclusion by contradiction. Suppose $\nabla^2 f(x^*)$ is not positive semi-definite, then there would be some vector v such that

$$v^T \nabla^2 f(x^*) v < 0.$$

Consider point $x = x^* + \epsilon v$. By Taylor's expansion and due to the already proved result $\nabla f(x^*) = 0$, we have

$$f(x) = f(x^* + \epsilon v) = f(x^*) + \frac{1}{2}\epsilon^2 v^T \nabla^2 f(\xi) v,$$

where ξ is a point between x^* and $x^* + \epsilon v$. When ϵ is sufficiently small, ξ would be very close to x^*. Hence $v^T \nabla^2 f(\xi) v < 0$. So, for such $x = x^* + \epsilon v$,

$$f(x) < f(x^*),$$

which contradicts the fact that x^* is a local minimizer. Therefore, the conclusion 2 is proved.

Second-Order Sufficient Conditions

Conclusion 3 Let $f \in C^2(\Re^n)$. If

$$\nabla f(x^*) = 0 \quad \text{and} \quad \nabla^2 f(x^*) \text{ is positive definite,}$$

then x^* is a strict local minimizer of f.

In fact

$$\begin{array}{rl} & \nabla^2 f(x^*) \text{ is positive definite} \\ \Rightarrow & \text{there is } \delta > 0 \text{ such that} \\ & \nabla^2 f(\xi) \text{ is positive definite if } \|\xi - x^*\| < \delta. \end{array}$$

Now for each x satisfying $0 < \|x - x^*\| < \delta$, let $p = x - x^*$, i.e., $x = x^* + p$, then

$$\begin{array}{rcl} f(x) & = & f(x^*) + \dfrac{1}{2} p^T \nabla^2 f(\xi) p \\ & > & f(x^*) \text{ (because } \nabla^2 f(\xi) \text{ is positive definite).} \end{array}$$

This means that x^* is a strict local minimizer.

For maximizing a function, we also have similar optimality conditions.

Conclusion 4

1. $f \in C^1(\Re^n)$ and x^* is a local maximizer of $f \Rightarrow \nabla f(x^*) = 0$.
2. $f \in C^2(\Re^n)$ and x^* is a local maximizer of $f \Rightarrow \nabla f(x^*) = 0$, and $\nabla^2 f(x^*)$ is negative semi-definite,
3. $\nabla f(x^*) = 0$, and $\nabla^2 f(x^*)$ is negative definite $\Rightarrow x^*$ is a strict local maximizer.

Example 1

Show that the point $c = (4, 0)^T$ is a strict local minimizer of the function

$$f(x) = (4 - x_1)^2 + x_2^2.$$

The gradient and Hessian of f at any point x are:

$$\nabla f(x) = \begin{bmatrix} -2(4 - x_1) \\ 2x_2 \end{bmatrix}, \qquad \nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

Since $\nabla f(c) = 0$ and $\nabla^2 f(c)$ is positive definite, the point c satisfies the sufficient conditions to be a strict local minimizer of f.

Example 2

Find stationary points for the function

$$f(x_1, x_2) = \frac{x_1^3}{3} + \frac{x_1^2}{2} + 2x_1x_2 + \frac{x_2^2}{2} - x_2 + 9,$$

and check if they are minimum or maximum points.

As

$$\nabla f(x) = \begin{bmatrix} x_1^2 + x_1 + 2x_2 \\ 2x_1 + x_2 - 1 \end{bmatrix},$$

by solving the equations $\nabla f(x) = 0$, we obtain two stationary points:

$$x_a = (1, -1)^T, \quad x_b = (2, -3)^T.$$

We now check the two points. The Hessian matrix is:

$$\nabla^2 f(x) = \begin{bmatrix} 2x_1 + 1 & 2 \\ 2 & 1 \end{bmatrix}.$$

So,

$$\nabla^2 f(x_a) = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix}.$$

This matrix is indefinite, and hence x_a is neither a minimizer, nor a maximizer.

$$\nabla^2 f(x_b) = \begin{bmatrix} 5 & 2 \\ 2 & 1 \end{bmatrix},$$

which is positive definite. Therefore, x_b is a strict local minimizer.

Section 4.2 Steepest Descent Method

- Assume that $f \in C^1(\Re^n)$ and an initial point x^0 is given.
- The gradient direction is the direction of the *steepest ascent*. And the direction of negative gradient is the direction of the *steepest descent*. See Section 4.6 (Appendix 1) for the reason.
- The steepest descent method searches along the direction of the negative gradient $-\nabla f(x^k)$ from the current point x^k to find a minimum point in this direction. This minimum point is taken as x^{k+1}.
- The steepest descent method is the simplest one among all methods for unconstrained optimization. It forms a basis for satisfactory analysis of descent methods.

4.2.1 Algorithm (Steepest Descent Method)

Step 0. Input and Initialization:
(a) Input $x^0 =$ the initial point.
(b) Let $k := 0$.

Step 1. Repeat the following computation (a)-(e) until certain stopping criteria are met (e.g., $\|\nabla f(x^k)\| < \varepsilon$ where ε is the tolerance which is usually a very small positive constant).
(a) Let $d_S^k := -\nabla f(x^k) =$ the steepest descent direction at x^k.
(b) Let $\phi_k(\alpha) := f(x^k + \alpha d_S^k)$.
(c) Use an one dimensional search method to determine the minimizer $\alpha_k^* > 0$ of the function $\phi_k(\alpha)$.
(d) Let $x^{k+1} := x^k + \alpha_k^* d_S^k$.
(e) Let $k := k + 1$.

Step 2. Output x^* which is the x^k satisfying the stopping criteria and $f(x^*)$.

4.2.2 The Quadratic Case

- We want to know how the method works. When the method is used to minimize a general function $f(x)$, it is quite difficult to analyze convergence behavior. Here we consider a special case: f is a quadratic function.
- Assume that

$$f(x) = \frac{1}{2}x^T Qx - b^T x = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}x_i x_j - \sum_{j=1}^{n} b_j x_j,$$

where Q is a positive definite symmetric $n \times n$ matrix. Thus

$$\nabla f(x) = Qx - b \quad \text{and} \quad \nabla^2 f(x) = Q.$$

Therefore, the unique minimum solution is

$$x^* = Q^{-1}b.$$

We now consider how the steepest descent method finds the minimum point.

- What is the step size if an exact line search is taken? Let $g_k \equiv \nabla f(x^k)$, and $\phi_k(\alpha) = f(x^k - \alpha g_k)$. By the chain rule,

$$\begin{aligned}
\phi_k'(\alpha) &= -\nabla f(x^k - \alpha g_k)^T g_k \\
&= -[Q(x^k - \alpha g_k) - b]^T g_k \\
&= b^T g_k - (x^k - \alpha g_k)^T Q g_k \\
&= b^T g_k - (x^k)^T Q g_k + \alpha g_k^T Q g_k.
\end{aligned}$$

  In an exact line search we should obtain the minimum point of $\phi_k(\alpha)$. So, we ask $\phi_k'(\alpha) = 0$, i.e.,

$$\alpha g_k^T Q g_k = (Qx^k - b)^T g_k = g_k^T g_k.$$

  So, the step length for the exact line search is

$$\alpha_k^* = \frac{g_k^T g_k}{g_k^T Q g_k}.$$

- Explicit form of the steepest descent method in quadratic case:

$$x^{k+1} = x^k - \alpha_k^* g_k = x^k - \frac{g_k^T g_k}{g_k^T Q g_k} g_k.$$

- Let us consider the amount $f(x^k) - f(x^*)$. As $Qx^* = b$ and Q is symmetric, we have

$$\begin{aligned}
& f(x^k) - f(x^*) \\
= \; & (\frac{1}{2}(x^k)^T Q x^k - b^T x^k) - (\frac{1}{2} x^{*T} Q x^* - b^T x^*) \\
= \; & \frac{1}{2}(x^k)^T Q x^k - (Qx^*)^T x^k - (\frac{1}{2} x^{*T} Q x^* - (Qx^*)^T x^*) \\
= \; & \frac{1}{2}(x^k)^T Q x^k - x^{*T} Q x^k + \frac{1}{2} x^{*T} Q x^* \\
= \; & \frac{1}{2}(x^k - x^*)^T Q (x^k - x^*).
\end{aligned}$$

- **Theorem.** Define

$$E(x) = \frac{1}{2}(x - x^*)^T Q(x - x^*).$$

For any $x^0 \in \Re^n$, the steepest descent method has the following property: for every step k,

$$E(x^{k+1}) \leq \left(\frac{A - a}{A + a}\right)^2 E(x^k),$$

where A and a are the largest and smallest eigenvalues of Q.

The proof of the theorem is quite difficult and thus omitted. Interested students may see pages 405-406 of the textbook and the file: Kantorovich Inequality put in I-Space.

- According to the theorem, $E(x^k) \to 0$, i.e., $f(x^k) \to f(x^*)$ when $k \to \infty$. As

$$\frac{a}{2}\|x^k - x^*\|^2 \leq E(x^k),$$

we see that $\|x^k - x^*\| \to 0$, i.e., $x^k \to x^*$, which means that the steepest descent method converges to the solution regardless of the location of the initial point. We say that the method converges *globally* if exact line search is used.

(**global convergence** means that the method can generate a sequence of iterative points that converges to the solution or a stationary point of the optimization problem independent of the location of the initial point.)

- The above theorem also tells us that if the sequence $r_k = f(x^k) - f(x^*)$ is concerned, roughly speaking, the steepest descent method converges linearly with a ratio not greater than
$$\left(\frac{A-a}{A+a}\right)^2.$$
- If all eigenvalues are equal, then the contours of f are a family of circles with the same center. How many steps does the steepest descent method need to reach the minimum point in this case?

Example (Fast Convergent)

Given a function:

$$\begin{aligned} f(x) &= (4 - x_1)^2 + x_2^2 \\ &= \frac{1}{2} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \begin{bmatrix} 8 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 16 \end{aligned}$$

and an initial point: $x^0 = (0, 0)^T$, find the minimum solution to f by the steepest descent method.

The gradient of f at any point x is:

$$\nabla f(x) = \begin{bmatrix} 2x_1 - 8 \\ 2x_2 \end{bmatrix}.$$

Thus,

$$\nabla f(x^0) = g_0 = \begin{bmatrix} -8 \\ 0 \end{bmatrix}.$$

By the steepest descent method, we have

$$\begin{aligned}\phi_0(\alpha) &= f(x^0 - \alpha\nabla f(x^0)) \\ &= f(\begin{bmatrix}8\alpha \\ 0\end{bmatrix}) = (4-8\alpha)^2.\end{aligned}$$

From

$$\phi_0'(\alpha^*) = -16(4-8\alpha^*) = 0,$$

we obtain $\alpha_0^* = \frac{1}{2}$, or equivalently,

$$\alpha_0^* = \frac{g_0^T g_0}{g_0^T Q g_0} = \frac{\begin{bmatrix}-8 & 0\end{bmatrix}\begin{bmatrix}-8 \\ 0\end{bmatrix}}{\begin{bmatrix}-8 & 0\end{bmatrix}\begin{bmatrix}2 & 0 \\ 0 & 2\end{bmatrix}\begin{bmatrix}-8 \\ 0\end{bmatrix}} = \frac{64}{128} = \frac{1}{2}.$$

So,

$$x^1 = x^0 - \alpha_0^*\nabla f(x^0) = \begin{bmatrix}0 \\ 0\end{bmatrix} - \frac{1}{2}\begin{bmatrix}-8 \\ 0\end{bmatrix} = \begin{bmatrix}4 \\ 0\end{bmatrix}.$$

Since

$$\nabla f(x^1) = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$x^1 = x^*$ is the minimum solution. In this example only one iteration is required to obtain the optimal solution. Note that here the two eigenvalues of Q are equal: $A = a = 2$.

Example (Slow Convergent)

Given a function:

$$\begin{aligned} f(x) &= x_1^2 + 10x_2^2 \\ &= \frac{1}{2}\begin{bmatrix} x_1 & x_2 \end{bmatrix}\begin{bmatrix} 2 & 0 \\ 0 & 20 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned}$$

and an initial point: $x^0 = (-3, 1)^T$, find the minimum solution to f by the steepest descent method. The gradient of f at any point x is:

$$\nabla f(x) = \begin{bmatrix} 2x_1 \\ 20x_2 \end{bmatrix}.$$

The optimal step length of the steepest descent method at iteration k is:

$$\begin{aligned}
\alpha_k^* &= \frac{g_k^T g_k}{g_k^T Q g_k} \\
&= \frac{\begin{bmatrix} 2x_1^k & 20x_2^k \end{bmatrix} \begin{bmatrix} 2x_1^k \\ 20x_2^k \end{bmatrix}}{\begin{bmatrix} 2x_1^k & 20x_2^k \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 20 \end{bmatrix} \begin{bmatrix} 2x_1^k \\ 20x_2^k \end{bmatrix}} \\
&= \frac{(x_1^k)^2 + 100(x_2^k)^2}{2(x_1^k)^2 + 2000(x_2^k)^2}.
\end{aligned}$$

Therefore, we have the following recursive relationship for the steepest descent method:

$$\begin{aligned} x^0 &= \begin{bmatrix} -3 \\ 1 \end{bmatrix}, \\ x^{k+1} &= x^k - \frac{(x_1^k)^2 + 100(x_2^k)^2}{2(x_1^k)^2 + 2000(x_2^k)^2} \begin{bmatrix} 2x_1^k \\ 20x_2^k \end{bmatrix}, \quad \text{for } k = 0, 1, 2, \ldots. \end{aligned}$$

The following table lists the first five iterations, as well as the 11th, the 15th, the 19th, the 25th and the 29th iterations. So the method progresses very slowly in this case. Note that in this example, the two eigenvalues of Q are $A = 20$ and $a = 2$. So,

$$(\frac{A-a}{A+a})^2 = (\frac{18}{22})^2 \approx 0.67.$$

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$
0	-3	1	19	20.9
1	-2.68	-8.03×10^{-2}	7.22	5.59
2	-1.14	3.80×10^{-1}	2.75	7.94
3	-1.02	-3.05×10^{-2}	1.04	2.12
4	-4.34×10^{-1}	1.45×10^{-1}	3.97×10^{-1}	3.02
5	-3.87×10^{-1}	-1.16×10^{-2}	1.51×10^{-1}	8.08×10^{-1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
11	-2.13×10^{-2}	-6.38×10^{-4}	4.57×10^{-4}	4.44×10^{-2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
15	-3.08×10^{-3}	-9.23×10^{-5}	9.55×10^{-6}	6.42×10^{-3}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$
19	-4.45×10^{-4}	-1.33×10^{-5}	2.00×10^{-7}	9.29×10^{-4}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
25	-2.45×10^{-5}	-7.34×10^{-7}	6.04×10^{-10}	5.11×10^{-5}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
29	-3.54×10^{-6}	-1.06×10^{-7}	1.26×10^{-11}	7.39×10^{-6}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

- Roughly speaking, the convergence rate of the steepest descent method is slowed down if the contours of f become flat.
  (a contour of function f is the graph of equation $f(x) = c$ for a constant c, see Appendix 2 for more details.)

4.2.4 The Non-Quadratic Case

- Assume that $f \in C^2(\Re^n)$ has a local minimizer x^* and $\nabla^2 f(x^*)$ has a smallest eigenvalue $a > 0$ and a largest eigenvalue $A > 0$. After a complicate proof, we can conclude that if $\{x^k\}$ is a sequence generated by the steepest descent method (Algorithm 4.2.1) with exact line search, then under some conditions, the sequence of objective values $\{f(x^k)\}$ converges to $f(x^*)$ linearly with a convergence ratio not greater than

$$\left(\frac{A-a}{A+a}\right)^2.$$

Also, $x^k \longrightarrow x^*$ independent of the location of the initial point. — **global convergence**

- When exact line search is used, the steepest descent method obtains

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

where α_k is the solution of the one-dimensional minimization problem

$$\min_{\alpha>0} F(\alpha) \equiv f(x^k - \alpha \nabla f(x^k)).$$

So,

$$\begin{aligned} F'(\alpha_k) = 0 \quad &\Rightarrow \quad -\nabla f(x^k - \alpha_k \nabla f(x^k))^T \nabla f(x^k) = 0 \\ &\Rightarrow \quad \nabla f(x^{k+1})^T \nabla f(x^k) = 0. \end{aligned}$$

This means that every pair of $\nabla f(x^{k+1})$ and $\nabla f(x^k)$ are vertical. Hence $\{x^k\}$ usually takes a zigzag path to approach the solution x^*. By this fact, we may better understand the reason why steepest descent method progresses slowly when the contours of f are flat, see Section 4.7 (Appendix 2).

To summarize, for the steepest descent method,

Advantages

- very easy to use;
- global convergence (if exact line search is conducted).

Disadvantage

- slow convergence.

Section 4.3 Newton's Method

- The idea behind Newton's method is that the function f being minimized is approximated locally by a quadratic function, and this approximate function is minimized exactly.
- Near x^k, $f(x)$ is approximated by the second-order Taylor expansion:

$$\begin{aligned} f(x) \approx q(x) &= f(x^k) + \nabla f(x^k)^T (x - x^k) \\ &\quad + \frac{1}{2}(x - x^k)^T \nabla^2 f(x^k)(x - x^k). \end{aligned}$$

The minimum point of $q(x)$ is taken as x^{k+1}, that is,

$$\nabla q(x^{k+1}) = \nabla f(x^k) + \nabla^2 f(x^k)(x - x^k)|_{x=x^{k+1}} = 0.$$

So,

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

- Question: For positive-definite quadratic functions, how many steps Newton's method need to reach the minimum point?

4.3.1 Algorithm (The original Newton's method)

Step 0. Input and Initialization:
(a) Input x^0 = the initial point.
(b) Let $k := 0$.

Step 1. Repeat the following computation (a)-(e) until certain stopping criterion is met. (e.g., $\|\nabla f(x^k)\| < \varepsilon$ = tolerance)
(a) If $\nabla^2 f(x^k)$ is NOT positive definite, then STOP!
(b) Let $d_N^k := -\nabla^2 f(x^k)^{-1} \nabla f(x^k)$.
(this d_N^k is called **Newton direction**)
(c) If $f(x^k + d_N^k) \geq f(x^k)$, then STOP!
(d) Let $x^{k+1} := x^k + d_N^k$.
(e) Let $k := k + 1$.

Step 2. Output x^* which is the x^k satisfying the stopping criteria and $f(x^*)$.

4.3.2 Example Given an initial point $x^0 = (2, 2)^T$, find the minimum solution to the following function:

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

At any point x, we have

$$\begin{aligned} \nabla f(x) &= \begin{bmatrix} -4x_1(x_2 - x_1^2) + 2(x_1 - 1) \\ 2(x_2 - x_1^2) \end{bmatrix}, \\ \nabla^2 f(x) &= \begin{bmatrix} 12x_1^2 - 4x_2 + 2 & -4x_1 \\ -4x_1 & 2 \end{bmatrix}. \end{aligned}$$

When $k = 0$,

$$\begin{aligned} f(x^0) &= 5, \\ \nabla f(x^0) &= \begin{bmatrix} 18 \\ -4 \end{bmatrix}, \\ \nabla^2 f(x^0) &= \begin{bmatrix} 42 & -8 \\ -8 & 2 \end{bmatrix}, \\ d_N^0 &= -\nabla^2 f(x^0)^{-1} \nabla f(x^0) = \begin{bmatrix} -0.2 \\ 1.2 \end{bmatrix}, \\ f(x^0 + d_N^0) &= 0.6416. \end{aligned}$$

Since $f(x^0 + d_N^0) < f(x^0)$, we set

$$x^1 := x^0 + d_N^0 = \begin{bmatrix} 1.8 \\ 3.2 \end{bmatrix}.$$

Repeat the above procedure, we have the following results:

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$
0	2.00000000	2.00000000	5.0	18.4
1	1.80000000	3.20000000	6.4×10^{-1}	1.9
2	1.05925926	0.57333333	3.0×10^{-1}	2.7
3	1.03100550	1.06217406	9.6×10^{-4}	6.5×10^{-2}
4	1.00004942	0.99914057	9.2×10^{-7}	4.4×10^{-3}
5	1.00000009	1.00000019	8.9×10^{-15}	2.0×10^{-7}
6	1.00000000	1.00000000	8.1×10^{-29}	4.1×10^{-14}

If the stopping criterion is $\|\nabla f(x^k)\| \leq 10^{-10}$, then we may stop at x^6.

It is proved that if $f \in C^3(\Re^n)$ and the Hessian $\nabla^2 f(x^*)$ is positive definite at the local minimum point x^*, then the order of convergence of Newton's method is at least two provided that the algorithm starts sufficiently close to x^*.

For Newton's method,

Advantage

- fast (second order) local convergence.

Disadvantages

- need to use n^2 second order partial derivatives;
- all $\nabla^2 f(x^k)$ should be positive definite;
- global convergence is not guaranteed. (because when the initial point is not close to the minimum point x^*, the method may fail to approach the solution x^*)

Section 4.4 Globally Convergent Modifications of Newton's Method – Damped Newton's Method

4.4.1 Newton's Method with Line Search

Newton's method requires modifications before it can be used at points that are remote from the solution.

Example

Given an initial point $x^0 = (3, 3)^T$, find the minimum solution to the following function:

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

(Note that this is the same function on Page 33, but the initial point is changed.)

Solving the problem by the Newton's method in its original form, we have the following results:

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$
0	3.00000000	3.00000000	40.0	76.9
1	2.84615385	8.07692308	3.4	4.0
2	1.08344198	-1.93330659	9.7	15.0

Notice that the Newton's method breaks down when $k = 2$ as $f(x^2) > f(x^1)$. If we continue, it may or may not converge.

Descent Direction If a direction vector d has the property that at least for sufficiently small $\alpha > 0$,

$$f(x + \alpha d) < f(x),$$

then d is said a *descent direction* for function f at point x.

- If the directional derivative of f along the direction d is negative at x, then d is a descent direction. In fact directional derivative is defined as the following limit:

$$\lim_{t \to 0} \frac{f(x + td) - f(x)}{t\|d\|}.$$

If this limit is negative, then for small $t > 0$,

$$\frac{f(x + td) - f(x)}{t\|d\|} < 0,$$

i.e., $f(x + td) < f(x)$, and d is a descent direction.

- Since

$$\lim_{t\to 0}\frac{f(x+td)-f(x)}{t\|d\|}=\lim_{t\to 0}\nabla f(x+\xi td)^T\frac{d}{\|d\|}=\nabla f(x)^T\frac{d}{\|d\|},$$

(ξ is between 0 and 1) if $\nabla f(x)^T d<0$, then d is a descent direction.

- In Newton's method, if $\nabla^2 f(x^k)$ is a positive definite matrix, then the Newton direction must be a descent direction. In fact as the Newton direction is $d_N^k=-[\nabla^2 f(x^k)]^{-1}\nabla f(x^k)$,

$$\nabla f(x^k)^T d_N^k=-\nabla f(x^k)^T[\nabla^2 f(x^k)]^{-1}\nabla f(x^k)<0,$$

which means that d_N^k is a descent direction of f at x^k.

- Therefore, if $\nabla^2 f(x^k)$ is positive definite, then for small $\alpha>0$, it must have

$$f(x^k+\alpha d_N^k)<f(x^k).$$

But for $\alpha=1$ the above inequality may not be true.

Remedy

A line search should be introduced in order to avoid the possibility that f might increase at the point $x^k + d_N^k$, that is, let

$$x^{k+1} = x^k + \alpha_k^* d_N^k,$$

where α_k^* is determined by an exact or inexact line search in the direction of d_N^k.

Algorithm (A revised Newton's method)

Step 0. Input and Initialization:

(a) Input $x^0 =$ the initial point.

(b) Let $k := 0$.

Step 1. Repeat the following computation (a)-(e) until certain stopping criteria are met (e.g., $\|\nabla f(x^k)\| < \varepsilon$).

(a) If $\nabla^2 f(x^k)$ is NOT positive definite, then STOP!

(b) Let $d_N^k := -[\nabla^2 f(x^k)]^{-1} \nabla f(x^k)$ (Newton direction).

(c) If $f(x^k + d_N^k) < f(x^k)$, then let $\alpha_k^* := 1$;
else use either exact or inexact line search to find $\alpha_k^* < 1$ such that $f(x^k + \alpha_k^* d_N^k) < f(x^k)$.
(this point will be further explained later)

(d) Let $x^{k+1} := x^k + \alpha_k^* d_N^k$.

(e) Let $k := k + 1$.

Step 2. Output x^* which is the x^k satisfying the stopping criteria and $f(x^*)$.

In the following computation for solving the problem of Page 37, in order to find α_k^* in Step 1(c) easily, we simply try $\alpha_k^* = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8} \ldots$ in the decreasing order until obtain the first one to meet the request $f(x^k + \alpha_k^* d_N^k) < f(x^k)$. The computation result is summarized in the table below.

Solving the Example Problem in Section 4.4.1 by an Inexact Line Search

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$	α_k^*
0	3.00000000	3.00000000	40.0	76.9	1
1	2.84615385	8.07692308	3.4	4.0	0.5
2	1.96479791	3.07180824	1.6	8.3	1
3	1.59044552	2.38937723	3.7×10^{-1}	2.1	1
4	1.12926064	1.06253809	6.2×10^{-2}	1.3	1
5	1.03857579	1.07041593	1.6×10^{-3}	1.1×10^{-1}	1
6	1.00062421	0.99980848	2.5×10^{-6}	7.6×10^{-3}	1
7	1.00000179	1.00000320	3.4×10^{-12}	5.2×10^{-6}	1
8	1.00000000	1.00000000	1.2×10^{-23}	1.7×10^{-11}	–

4.4.2 Newton's Method with Line Search and Modified Hessian Matrix

- $\nabla^2 f(x^k)$ may not be positive definite or even be singular.
- Newton's method must be modified to accommodate the possible non-positive definiteness at regions remote from the solution.

Example

Given an initial point $x^0 = (-2, 5)^T$, find the minimum solution to the following function:

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

Here function f is the same, but the initial point is changed again.

When $k = 0$,

$$\nabla^2 f(x^0) = \begin{bmatrix} 30 & 8 \\ 8 & 2 \end{bmatrix}.$$

Since its eigenvalues are 32.125 and -0.125, it is not positive definite. Thus, the Newton's method breaks down when $k = 0$. In fact $\nabla f(x^0) = (2, 2)^T$ and the Newton direction

$$d = -(\nabla^2 f(x^0))^{-1} \nabla f(x^0) = (-3, 11)^T.$$

So,

$$d^T \nabla f(x^0) = 16 > 0,$$

i.e., the Newton direction at x^0 is not a descent direction.

Remedy

To replace $\nabla^2 f(x^k)$ by $\nabla^2 f(x^k) + \beta_k I$ for some non-negative value of β_k and let

$$x^{k+1} = x^k - \alpha_k^* [\nabla^2 f(x^k) + \beta_k I]^{-1} \nabla f(x^k).$$

This can be viewed as a kind of compromise between the steepest descent method (if β_k is very large) and Newton's method (if $\beta_k = 0$).

As we know, if λ is an eigenvalue of $\nabla^2 f(x^k)$, then $\lambda + \beta_k$ is an eigenvalue of $\nabla^2 f(x^k) + \beta_k I$. So, suppose the smallest eigenvalue of $\nabla^2 f(x^k)$ is λ_{min} which is negative, then for any $\beta_k > -\lambda_{min}$, all eigenvalues of $\nabla^2 f(x^k) + \beta_k I$ are positive. In other words, $\nabla^2 f(x^k) + \beta_k I$ is a positive definite matrix.

We now further revise original Newton's method. In the textbook, the following revised method is called *damped Newton's method*.

Algorithm (A Damped Newton's Method)

Step 0. Input and Initialization:

(a) Input $x^0 =$ the initial point.

(b) Let $k := 0$.

Step 1. Repeat the following computation (a)-(e) until certain stopping criteria are met (e.g., $\|\nabla f(x^k)\| < \varepsilon$).

(a) If $\nabla^2 f(x^k)$ is positive definite, then let $\beta_k := 0$; else find $\beta_k > 0$ such that $\nabla^2 f(x^k) + \beta_k I$ is positive definite.

(b) Let $d^k := -[\nabla^2 f(x^k) + \beta_k I]^{-1} \nabla f(x^k)$.

(c) (line search) Use the following Amijo's rule to find a $\alpha_k^* > 0$.

(d) Let $x^{k+1} := x^k + \alpha_k^* d^k$.

(e) Let $k := k + 1$.

Step 2. Output x^* which is the x^k satisfying the stopping criteria and $f(x^*)$.

We now explain the **Amijo's rule**. The rule is an inexact line search method which is in fact used quite generally, not only for the damped Newton's method. As long as the search direction d^k is a descent direction, the method can be utilized if an exact line search is time-consuming.

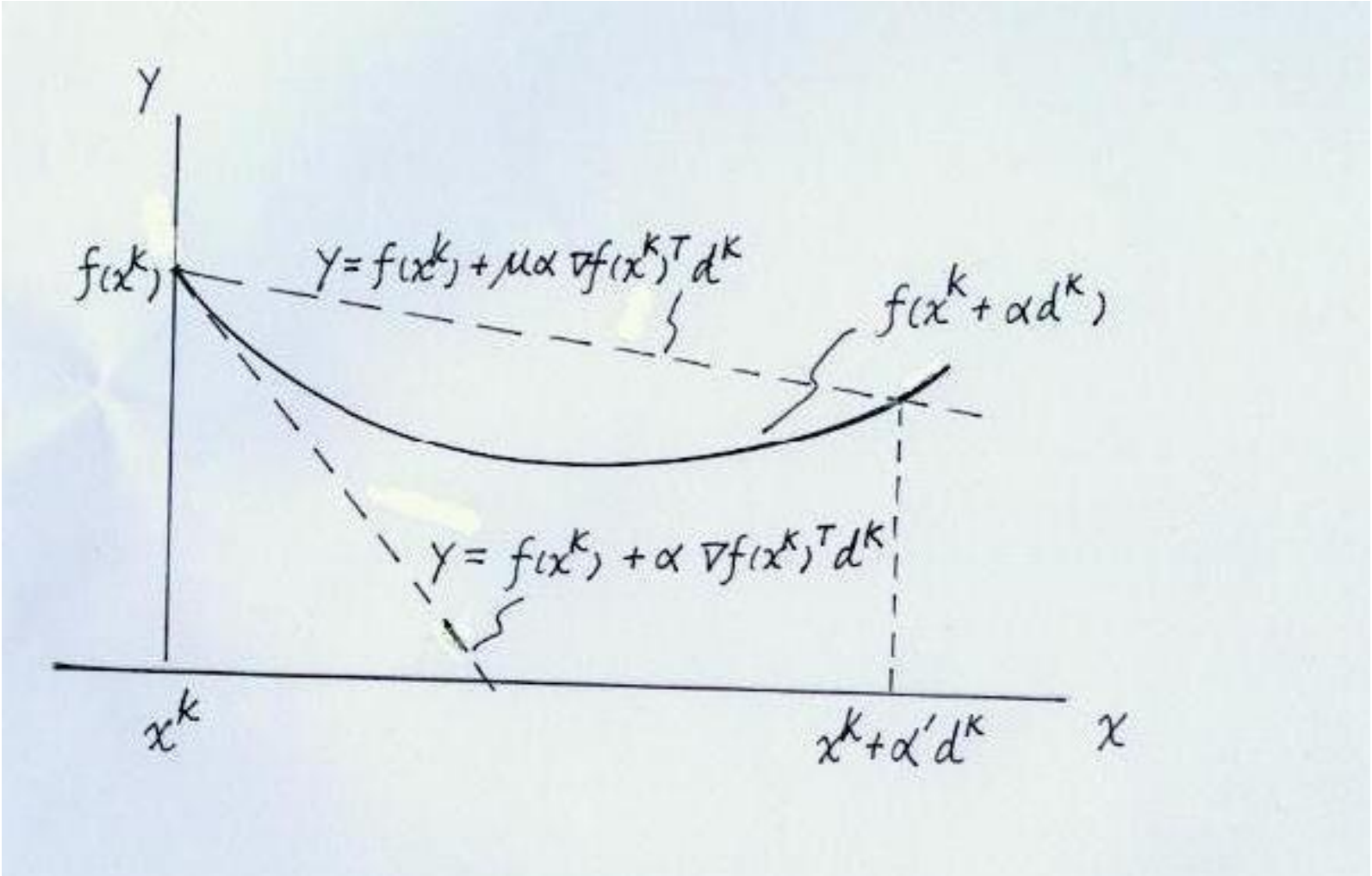
It is found that in order to guarantee global convergence, letting $f(x^k + \alpha_k^* d^k)$ be smaller than $f(x^k)$ is not enough (sometimes it is successful such as the example on Pages 43-44, but not always so), and we need to ask $f(x^k + \alpha_k^* d^k)$ to be smaller than $f(x^k)$ by a certain level. Such request is called a **sufficient decrease condition**.

A frequently used condition is that we ask the step length α_k to meet the inequality:

$$f(x^k + \alpha_k d^k) \leq f(x^k) + \mu \alpha_k \nabla f(x^k)^T d^k, \tag{1}$$

where μ is a given scalar satisfying $0 < \mu < 1$. Note that as d^k is a descent direction, the second term on the right side is negative.

The geometric meaning of the condition can be seen from the figure below. That is, we should choose an α_k in the interval $(0, \alpha']$ where the function curve is below the line $y = f(x^k) + \mu \alpha \nabla f(x^k)^T d^k$.



From the graph, we see that the coefficient μ in the sufficient decrease condition is important. If we do not introduce μ, the condition may not be satisfied by any positive α.

It can be proved that for sufficiently small α, inequality (1) must be satisfied. In fact

$$\begin{aligned} & f(x^k + \alpha d^k) - f(x^k) - \mu\alpha\nabla f(x^k)^T d^k \\ = \; & \alpha\nabla f(x^k)^T d^k + \frac{1}{2}\alpha^2 (d^k)^T \nabla^2 f(\xi) d^k - \mu\alpha\nabla f(x^k)^T d^k \\ = \; & (1-\mu)\alpha\nabla f(x^k)^T d^k + \frac{1}{2}\alpha^2 (d^k)^T \nabla^2 f(\xi) d^k, \qquad (2) \end{aligned}$$

where ξ is a point between x^k and $x^k + \alpha d^k$. So, when α is small enough, $\|\nabla^2 f(\xi)\|$ must be bounded, say $\|\nabla^2 f(\xi)\| \leq K$, where K is a positive number. Now,

$$(d^k)^T \nabla^2 f(\xi) d^k \leq K\|d^k\|^2 \stackrel{def}{=} L.$$

Let

$$\nabla f(x^k)^T d^k \stackrel{def}{=} -M$$

($M > 0$). Then from (2) we see that

$$\begin{aligned} & f(x^k + \alpha d^k) - f(x^k) - \mu\alpha\nabla f(x^k)^T d^k \\ \leq & -(1-\mu)\alpha M + \frac{1}{2}\alpha^2 L \\ = & \alpha[-(1-\mu)M + \frac{1}{2}\alpha L]. \end{aligned}$$

It is seen that when α is small enough the right hand side must be negative. Hence the sufficient decrease condition (1) holds.

The above reasoning tells us that for small α, this condition can be satisfied. However, α cannot be too small, for otherwise each step would move a very short distance making the progress of computation very slow. The Amijo's rule asks that

The Amijo's Rule for an Inexact Line Search

Let α_k^* be the first element of the sequence

$$1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots\ldots$$

that satisfies the sufficient decrease condition (1):

$$f(x^k + \alpha d^k) \leq f(x^k) + \mu\alpha\nabla f(x^k)^T d^k.$$

According to this rule, we would first try $\alpha = 1$ because in the original Newton's method a step of one is used, and near the minimum point, we would expect that a step of $\alpha_k^* = 1$ would be acceptable and lead to a quadratic convergence rate. If $\alpha = 1$ does not satisfy condition (1), we try $\alpha = \frac{1}{2}$, $\alpha = \frac{1}{4}$, etc, (each time the step length is halved) until an acceptable α is found.

Solving the Example Problem in Section 4.4.2

When $k = 0$, $x^0 = (-2, 5)^T$,

$$f(x^0) = 10, \quad \nabla f(x^0) = \begin{bmatrix} 2 \\ 2 \end{bmatrix},$$

$$\nabla^2 f(x^0) = \begin{bmatrix} 30 & 8 \\ 8 & 2 \end{bmatrix}.$$

Letting $\beta_0 = 1$, we have

$$\nabla^2 f(x^0) + \beta_0 I = \begin{bmatrix} 31 & 8 \\ 8 & 3 \end{bmatrix}.$$

Since its eigenvalues are 33.125 and 0.875, it is positive definite.

Thus, we have

$$d^0 = -[\nabla^2 f(x^0) + \beta_0 I]^{-1} \nabla f(x^0) = \begin{bmatrix} 0.34482759 \\ -1.58620690 \end{bmatrix}.$$

We now use Amijo's rule to determine the step length. Suppose we take $\mu = 0.5$. We try $\alpha = 1$ first.

$$f(x^0 + \alpha d^0) = f(x^0 + d^0) = 7.5045.$$

On the other hand,

$$\begin{aligned} & f(x^0) + \mu\alpha \nabla f(x^0)^T d^0 \\ = \; & 10 + \frac{1}{2}(2, 2) \begin{bmatrix} 0.34482759 \\ -1.58620690 \end{bmatrix} \\ = \; & 8.7586. \end{aligned}$$

Hence the sufficient decrease condition is satisfied and the step length $\alpha_0^* = 1$.

We take

$$x^1 = x^0 + \alpha_0^* d^0 = x^0 + d^0 = \begin{bmatrix} -1.65517241 \\ 3.41379310 \end{bmatrix}.$$

Repeat the above procedure, we have the following results:

k	x_1^k	x_2^k	$f(x^k)$	$\|\nabla f(x^k)\|$	β_k	α_k^*
0	-2.00000000	5.00000000	10.0	2.8	1	1
1	-1.65517241	3.41379310	7.5	1.6	1	1
2	-1.15279866	1.85564127	4.9	2.2	1	1
3	-0.36488382	0.29343403	1.9	2.5	0	0.5
4	0.63957528	-0.51973463	9.9×10^{-1}	2.5	0	1
5	0.76570453	0.57039484	5.5×10^{-2}	4.2×10^{-1}	0	1
6	0.99277525	0.93404159	2.7×10^{-3}	2.2×10^{-1}	0	1
7	0.99932461	0.99860679	4.6×10^{-7}	1.2×10^{-3}	0	1
8	0.99999994	0.99999943	2.1×10^{-13}	1.9×10^{-6}	0	1
9	1.00000000	1.00000000	2.8×10^{-27}	9.3×10^{-14}	–	–

We see that in fact only one iteration used the step length $\alpha = \frac{1}{2}$, and other iterations all took $\alpha = 1$. In the first 3 iterations, $\nabla^2 f(x^k)$ are not positive definite, and we need to add the term $\beta_k I$ to $\nabla^2 f(x^k)$ with $\beta_k = 1$.

Summary of Newton's Method

- The classic (original) Newton's method is motivated by approximating the objective function $f(x)$ by its second order Taylor's expansion. Its formula is

$$x^{k+1} = x^k - [\nabla^2 f(x^k)]^{-1} \nabla f(x^k), \quad k = 1, 2, \ldots$$

- The classic Newton's method does not make line search and hence its global convergence is not guaranteed. When $\nabla^2 f(x^k)$ is not positive definite, $f(x^{k+1})$ may be even larger than $f(x^k)$. Also, the algorithm breaks down if $\nabla^2 f(x^k)$ is not invertible.
- To overcome these disadvantages, some revisions have been made, and we have the Damped Newton's method, which uses exact or some inexact line search, and adds a term βI to $\nabla^2 f(x^k)$ if it is not positive definite.

Section 4.5 Quasi-Newton Method

Main purpose – to develop a class of minimization methods which are faster than the steepest descent method but use only first order derivatives.

4.5.1 Secant Method

- Recall that in the previous chapter, for single variable minimization problems we introduced the secant method, which is based on the approximation:

$$f''(x^k) \approx \frac{f'(x^k) - f'(x^{k-1})}{x^k - x^{k-1}}.$$

- This formula cannot be used in the multi-variable case because now on the right hand side both numerator and denominator are vectors and their division is not defined.
- Rewrite it as

$$f''(x^k)(x^k - x^{k-1}) \approx f'(x^k) - f'(x^{k-1}),$$

which can be extended to multi-variable case:

$$\nabla^2 f(x^k)(x^k - x^{k-1}) \approx \nabla f(x^k) - \nabla f(x^{k-1}).$$

- We now want to use a matrix B_k to approximate $\nabla^2 f(x^k)$, and ask it to satisfy the condition

$$B_k(x^k - x^{k-1}) = \nabla f(x^k) - \nabla f(x^{k-1}).$$

The above request is called *secant condition* or *quasi-Newton equation*. Let

$$s_k = x^{k+1} - x^k, \quad \text{and} \quad y_k = \nabla f(x^{k+1}) - \nabla f(x^k),$$

then the above secant condition becomes

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

or more commonly,

$$B_{k+1} s_k = y_k.$$

- How to obtain a B_{k+1} which satisfies the secant condition? We wish that B_{k+1} is obtained by modifying the previous approximation B_k:

$$B_{k+1} = B_k + [\text{update term}].$$

  Such formula is called an (quasi-Newton) *update formula*. We ask that the update formula uses the first order derivatives (i.e. gradient) only, but **NOT** second order derivatives.
- To determine B_{k+1}, note that it has $n \times n = n^2$ unknowns, but the secant condition contains only n equations. So we may have many different updating formulas.

4.5.2 Symmetric Rank-One Update Formula

Let

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

It is easy to verify that

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

In the update formula, only first order derivatives are used.

This formula is called *symmetric rank-one update formula*, because

1. if B_k is symmetric, so is B_{k+1};

2. the update part

$$[\text{update term}] = \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}$$

is a rank-one matrix. In fact any matrix rab^T, where $r \in \Re^1$, $a, b \in \Re^n$, has the form

$$\begin{aligned} rab^T &= r(a_1, a_2, \ldots, a_n)^T (b_1, \ldots, b_n) \\ &= r \begin{bmatrix} a_1 b_1 & a_1 b_2 & \cdots & a_1 b_n \\ a_2 b_1 & a_2 b_2 & \cdots & a_2 b_n \\ \cdots & \cdots & \cdots & \cdots \\ a_n b_1 & a_n b_2 & \cdots & a_n b_n \end{bmatrix} = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ rb_1 a & rb_2 a & \cdot & rb_n a \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}. \end{aligned}$$

We see that all columns are proportional, and hence the matrix has rank 1.

4.5.3 Algorithm (Quasi-Newton Method)

Step 0. Initialization: Input $x^0 =$ the initial point, and choose an initial Hessian approximation B_0 (if $\nabla^2 f(x^0)$ is not available or not positive definite, take $B_0 = I$.) Let $k := 0$.

Step 1. Solve the equation

$$B_k p = -\nabla f(x^k)$$

for p_k.

Step 2. Use a line search (exact or inexact) to determine

$$x^{k+1} = x^k + \alpha_k p_k.$$

Step 3. If certain stopping criterion is met (e.g., $\|\nabla f(x^{k+1})\| < \varepsilon =$ tolerance) then go to Step 7.

Step 4. Compute

$$s_k = x^{k+1} - x^k, \text{ and } y_k = \nabla f(x^{k+1}) - \nabla f(x^k).$$

Step 5. Update

$$B_{k+1} = B_k + [\text{update term}]$$

by using some update formula that satisfies the secant condition.

Step 6. Let $k := k + 1$, and go back to Step 1.

Step 7. Output $x^* = x^{k+1}$ and $f(x^*)$.

Example

We consider a three dimensional quadratic problem $f(x) = \frac{1}{2}x^T Qx - c^T x$ with

$$Q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 4 \end{bmatrix} \quad \text{and} \quad c = \begin{bmatrix} -8 \\ -9 \\ -8 \end{bmatrix},$$

whose solution is $x^* = (-4, -3, -2)^T$. An exact line search will be used so that the step length is

$$\alpha = -\frac{p^T \nabla f(x)}{p^T Qp},$$

(see exercises) where p is the search direction. The initial guesses are $B_0 = I$, and $x^0 = (0, 0, 0)^T$. As $\nabla f(x) = Qx - c$, we know that $\nabla f(x^0) = -c = (8, 9, 8)^T$.

At the initial point, $\|\nabla f(x^0)\| = \|-c\| = 14.4568$, so this point is not optimal. The first search direction is

$$p_0 = -(B_0)^{-1}\nabla f(x^0) = c = (-8, -9, -8)^T,$$

and the line search formula gives $\alpha_0 = 0.3333$. We obtain $x^1 = x^0 + 0.3333p^0$. So,

$$x^1 = \begin{bmatrix} -2.6667 \\ -3.0000 \\ -2.6667 \end{bmatrix}, \quad \nabla f(x^1) = \begin{bmatrix} 2.6667 \\ 0 \\ -2.6667 \end{bmatrix},$$

$$s_0 = \begin{bmatrix} -2.6667 \\ -3.0000 \\ -2.6667 \end{bmatrix}, \quad y_0 = \begin{bmatrix} -5.3333 \\ -9.0000 \\ -10.6667 \end{bmatrix},$$

and

$$B_1 = I + \frac{(y_0 - Is_0)(y_0 - Is_0)^T}{(y_0 - Is_0)^T s_0} = \begin{bmatrix} 1.1531 & 0.3445 & 0.4593 \\ 0.3445 & 1.7751 & 1.0335 \\ 0.4593 & 1.0335 & 2.3780 \end{bmatrix}.$$

At x^1, $\|\nabla f(x^1)\| = 3.7712$ which is reduced but still quite big. So we keep going, obtaining the search direction

$$p_1 = -B_1^{-1}\nabla f(x^1) = \begin{bmatrix} -2.9137 \\ -0.5557 \\ 1.9257 \end{bmatrix},$$

and the step length $\alpha_1 = 0.3942$. We then obtain:

$$x^2 = \begin{bmatrix} -3.8152 \\ -3.2191 \\ -1.9076 \end{bmatrix}, \quad \nabla f(x^2) = \begin{bmatrix} 0.3697 \\ -0.6572 \\ 0.3697 \end{bmatrix},$$

$$s_1 = \begin{bmatrix} -1.1485 \\ -0.2191 \\ 0.7591 \end{bmatrix}, \quad y_1 = \begin{bmatrix} -2.2970 \\ -0.6572 \\ 3.0363 \end{bmatrix},$$

and

$$B_2 = B_1 + \frac{(y_1 - B_1 s_1)(y_1 - B_1 s_1)^T}{(y_1 - B_1 s_1)^T s_1} = \begin{bmatrix} 1.6568 & 0.6102 & -0.3432 \\ 0.6102 & 1.9153 & 0.6102 \\ -0.3432 & 0.6102 & 3.6568 \end{bmatrix}.$$

At x^2, $\|\nabla f(x^2)\| = 0.8397$. We continue and obtain the search direction

$$p_2 = -B_2^{-1}\nabla f(x^2) = \begin{bmatrix} -0.4851 \\ 0.5749 \\ -0.2426 \end{bmatrix},$$

and the step length $\alpha_2 = 0.3810$. We then obtain:

$$x^3 = \begin{bmatrix} -4 \\ -3 \\ -2 \end{bmatrix}, \quad \nabla f(x^3) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$s_2 = \begin{bmatrix} -0.1848 \\ 0.2191 \\ -0.0924 \end{bmatrix}, \quad y_2 = \begin{bmatrix} -0.3679 \\ 0.6572 \\ -0.3679 \end{bmatrix},$$

and B_3. Now $\nabla f(x^3) = 0$, so we stop and the optimal solution is $x^* = x^3$.

4.5.4 BFGS Update Formula

- It is found that if we want to have a rank-one update formula that maintains the symmetric property, i.e., B_{k+1} is symmetric as long as B_k is so, then the above update formula is the unique choice, and people cannot find any more update formula.
- It is useful to ask all matrices B_k to be positive definite so that p_k are descent directions:

$$\nabla f(x^k)^T p_k = -\nabla f(x^k)^T B_k^{-1} \nabla f(x^k) < 0.$$

So, we often require that: if B_k is positive definite, an update formula can let B_{k+1} be positive definite, too.

- Unfortunately, the symmetric rank-one update does not have the property.
- The following BFGS (Broyden-Fletcher-Goldfarb-Shanno) update formula was proposed.

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

where again $s_k = x^{k+1} - x^k$, and $y_k = \nabla f(x^{k+1}) - \nabla f(x^k)$. This update formula also uses first order derivatives only.

- It is easy to verify that

$$B_{k+1} s_k = B_k s_k - B_k s_k + y_k = y_k,$$

that is, BFGS formula meets the secant condition. Also, we see that if B_k is symmetric, so is B_{k+1}.

- Each of the two update terms on the right hand side is a rank one matrix. So, this is a rank-two update formula (for a clear proof, see Appendix 3).
- It is proved that

Theorem Let B_k be a symmetric positive definite matrix and B_{k+1} be obtained using BFGS formula. Then B_{k+1} is positive definite if and only if $y_k^T s_k > 0$.

A proof of the theorem can be seen on page 354 of the textbook.

- The request $y_k^T s_k > 0$ can be guaranteed by performing an appropriate line search. For example, if we make exact line search, then it is quite easy to show that $y_k^T s_k > 0$. (see Appendix 5)
- There are many other update formulas, but BFGS formula is widely recognized as the one with best numerical performance.
- It is proved that with a lot of quasi-Newton updates, including the BFGS update formula, under mild conditions, the quasi-Newton method is globally convergent with a superlinear convergence order. So, this type of methods is desirable as it uses only the first order derivatives, but it is able to achieve superlinear convergence.

In the chapter we have learned three methods for unconstrained optimization: steepest descent method, Newton's method (and damped Newton's method) and quasi-Newton method (also called secant method). We may compare them from the following aspects. First, what order of partial derivatives these methods use, first order or second order? Second, when these methods are convergent, in general, what is the convergence rate, linear, superlinear, or quadratic?

Also, we should know that for these methods, if we do not use line search and let the step length equal one, they may not be globally convergent; and if an exact line search or an Amijo type approximate line search is used, most likely they are globally convergent.

Section 4.6 Appendix 1 - Directional Derivative and Steepest Descent Direction

4.6.1 Directional Derivative

We know that

$$\frac{\partial f}{\partial x}(x_0, y_0) = \lim_{\triangle x \to 0} \frac{f(x_0 + \triangle x, y_0) - f(x_0, y_0)}{\triangle x},$$

which is the rate of change of f at (x_0, y_0) if we approach the point (x_0, y_0) along the x- axis direction. Similarly, $\frac{\partial f}{\partial y}(x_0, y_0)$ represents the rate of change of f at (x_0, y_0) if we approach the point (x_0, y_0) along the y- axis.

Suppose we are given a unit vector $u = (u_1, u_2)^T$, *what is the rate of change of f at (x_0, y_0) if we approach the point along the direction u?*

We should consider

$$\lim_{t\to 0^+} \frac{f(x+tu)-f(x)}{t\|u\|} = \lim_{t\to 0^+} \frac{f(x+tu)-f(x)}{t}.$$

By the mean-value theorem,

$$f(x+tu)-f(x) = \nabla f(\xi)^T(tu),$$

where ξ is a point between x and $x+tu$. When $t \to 0^+$, $\xi \to x$. Therefore,

$$\lim_{t\to 0^+} \frac{f(x+tu)-f(x)}{t\|u\|} = \lim_{t\to 0^+} \nabla f(\xi)^T u = \nabla f(x)^T u.$$

Let u be a unit vector. The above limit is called directional derivative of f in the direction of u at (x_0, y_0), denoted by $D_u f(x_0, y_0)$. We have the formula:

$$D_u f(x_0, y_0) = \nabla f(x_0, y_0)^T u.$$

Obviously, if $u = (1, 0)$, then

$$D_u f(x_0, y_0) = f_x(x_0, y_0) \cdot 1 + f_y(x_0, y_0) \cdot 0 = f_x(x_0, y_0);$$

and if if $u = (0, 1)$, then

$$D_u f(x_0, y_0) = f_x(x_0, y_0) \cdot 0 + f_y(x_0, y_0) \cdot 1 = f_y(x_0, y_0).$$

So, partial derivatives are special cases of directional derivative.

Note that if u is not a unit vector, then when we consider the directional derivative along direction u, we should consider the unit vector in the direction, i.e., take vector $\frac{u}{\|u\|}$.

Example 1

Find the directional derivative of $f(x, y) = 3x^2y$ at point $(1, 2)$ in the direction $v = (3, 4)$.

Solution:

$$f_x(x, y) = 6xy, \quad f_y(x, y) = 3x^2,$$

$$f_x(1, 2) = 12, \quad f_y(1, 2) = 3,$$

$$u = \frac{v}{\|v\|} = (\frac{3}{5}, \frac{4}{5}).$$

So,

$$\begin{aligned} D_u f(1, 2) &= f_x(1, 2)u_1 + f_y(1, 2)u_2 \\ &= 12 \cdot \frac{3}{5} + 3 \cdot \frac{4}{5} = \frac{48}{5}. \end{aligned}$$

4.6.2 Steepest Ascent and Descent Directions

Suppose $\nabla f(x_0, y_0) \neq 0$, and consider all unit vectors u.

1. Along which u, $D_u f(x_0, y_0)$ has the largest value?
2. Along which u, $D_u f(x_0, y_0)$ has the smallest value?

Solution:

$$\begin{aligned} D_u f(x_0, y_0) &= \nabla f(x_0, y_0)^T u \\ &= \|\nabla f(x_0, y_0)\| \cdot \|u\| \cdot \cos\alpha \\ &= \|\nabla f(x_0, y_0)\| \cdot \cos\alpha, \end{aligned}$$

where α is the angle between the vectors $\nabla f(x_0, y_0)$ and u.

- If $\alpha = 0$, i.e., u and $\nabla f(x_0, y_0)$ point to the same direction, then the directional derivative has the largest value:

$$D_u f(x_0, y_0) = \|\nabla f(x_0, y_0)\|.$$

  — call $\nabla f(x_0, y_0)$ *the steepest ascent direction.*

- If $\alpha = \pi$, i.e., u and $\nabla f(x_0, y_0)$ point to the opposite directions, then the directional derivative has the smallest value:

$$D_u f(x_0, y_0) = -\|\nabla f(x_0, y_0)\|.$$

  — call $-\nabla f(x_0, y_0)$ *the steepest descent direction.*

Example 2

For the function $f(x,y) = x^2 e^y$, find the maximum value of a directional derivative at $(-2,0)$, and give a unit vector along which the maximum value is reached.

Solution: $f_x = 2xe^y, \quad f_y = x^2 e^y$, and $\nabla f(-2,0) = (-4,4)^T$. The maximum value of directional derivative is

$$\|\nabla f(-2,0)\| = \sqrt{(-4)^2 + 4^2} = 4\sqrt{2},$$

which occurs in the direction $\nabla f(-2,0) = (-4,4)^T$, and the unit vector in that direction is

$$u = \frac{\nabla f(-2,0)}{\|\nabla f(-2,0)\|} = \begin{bmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}.$$

Section 4.7 Appendix 2 - Graphs of Quadratic Functions

1. Consider a quadratic function

$$f(x) = x^T Q x - b^T x,$$

where Q is a positive definite symmetric matrix. We ask: **what are contours of $f(x)$, i.e., what are graphs of $f(x) = c$?**

To answer this question, we need to change the coordinate system. If we move the origin of the coordinate system suitably, then under the new coordinate system, say the variables become $\bar{x}$, the function may contain only the quadratic terms:

$$f(x) = \bar{x}^T Q \bar{x}.$$

For example, in Example 1 of Section 4.2,

$$\begin{aligned} f(x) &= x_1^2 + x_2^2 - 8x_1 + 16 \\ &= (4 - x_1)^2 + x_2^2. \end{aligned}$$

If we let $\bar{x}_1 = x_1 - 4$ and $\bar{x}_2 = x_2$, then $f(x) = \bar{x}_1^2 + \bar{x}_2^2$, which does not have linear and constant terms.

2. It is known that for any positive definite and symmetric matrix Q, there exists an (orthogonal) matrix P such that

$$PQP^T = D = diag(d_1, d_2, \ldots, d_n),$$

where all d_i are eigenvalues of Q, and matrix P has the property that $P^T = P^{-1}$.

So, if we let $y = P\bar{x}$, then $\bar{x} = P^{-1}y = P^T y$, and

$$\begin{aligned} f(x) &= \bar{x}^T Q \bar{x} \\ &= y^T P Q P^T y \\ &= y^T D y \\ &= d_1 y_1^2 + d_2 y_2^2 + \cdots + d_n y_n^2. \end{aligned}$$

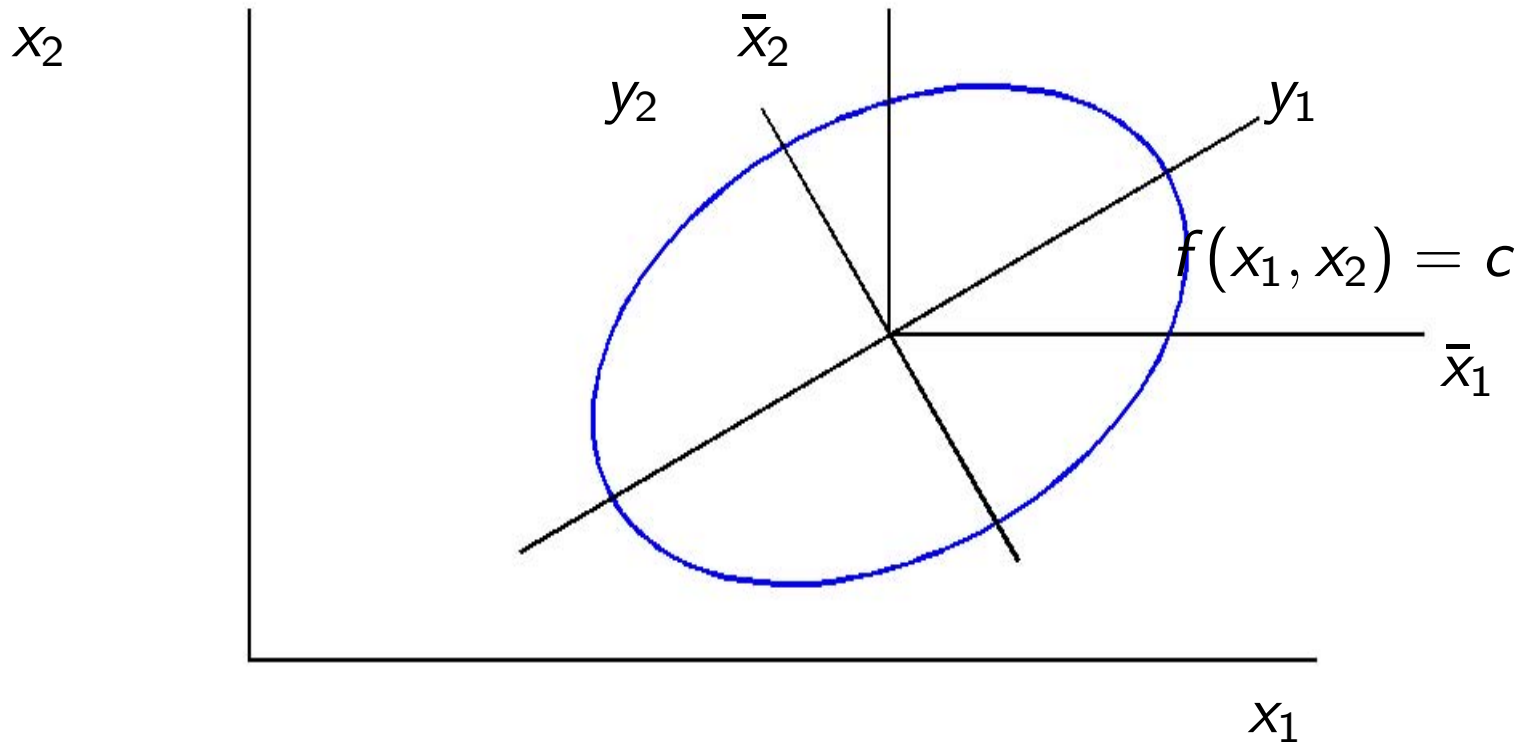
That is, when we change variables x into y, function f contains only pure square terms y_i^2 and has no cross product terms such as $\beta y_i y_j$ for $i \neq j$. Actually, changing variables from $\bar{x}$ to y represents a rotation and enlargement/reduction of the coordinate system.

3. For any $c > 0$, if all $d_i > 0$, then the contour of

$$f(x) = d_1 y_1^2 + d_2 y_2^2 + \cdots + d_n y_n^2 = c$$

is an elliptic surface centered at the origin of the coordinate system y, and the half-lengths of the axes of the elliptic surface are $\sqrt{c/d_i}$. This is because the surface equation can be expressed equivalently as

$$\sum_{i=1}^{n} (\frac{y_i}{\sqrt{\frac{c}{d_i}}})^2 = 1.$$

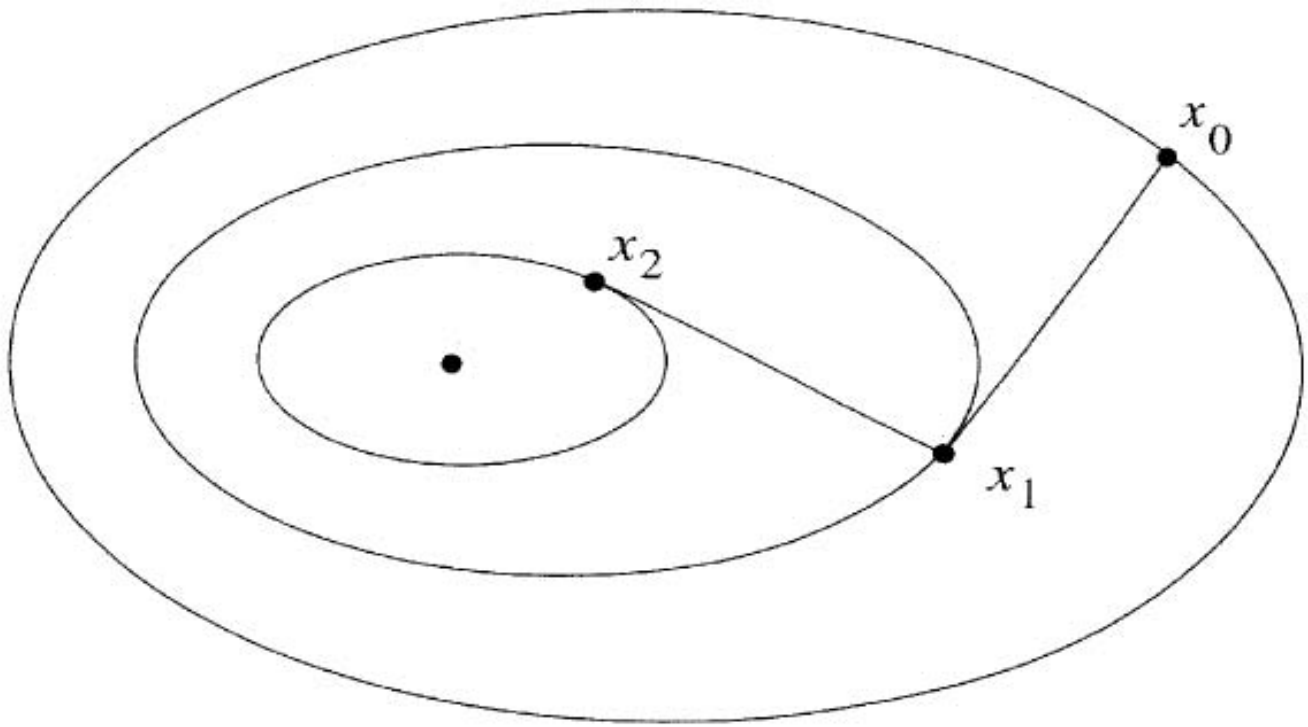


So, if all eigenvalues d_i $(i = 1, \ldots, n)$ are equal, the surface is a spherical surface, whereas if the largest and the smallest eigenvalues of Q have big difference, the surface would be very flat in certain direction.

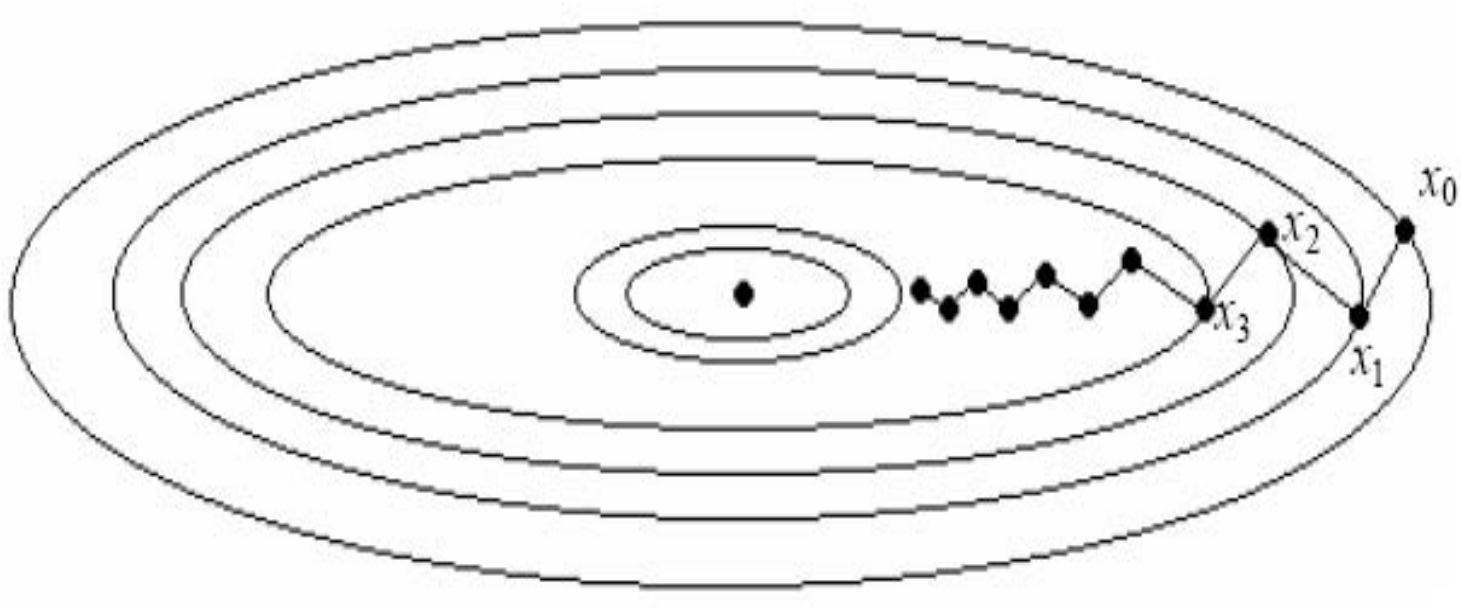
4. For any continuously differentiable function $f(x)$, at each point x^* on the contour $f(x) = c$, the normal vector of the contour at point x^* is the vector $\nabla f(x^*)$ that points to the increasing direction of $f(x)$.

5. With the above facts, we know that when the steepest descent method is applied to minimize a positive definite quadratic function $f(x)$, the iterative points shall move along the path shown in the graphs below.

In the two graphs below, the contours $f(x) = c$ are a set of ellipses, and an inside ellipse corresponds to a smaller value of c. If we start from x_0, the progress of the steepest descent method will be shown in the two graphs. If the ellipses are flat, like the second graph shows, then the progress is slow by taking a zigzag path.



Steepest-descent in Two Dimensions



Steepest Descent Method in Two Dimensions

Section 4.8 Appendix 3 - A Rank Two Matrix

Let matrices

$$A = taa^T, \quad \text{and} \quad B = \tau bb^T,$$

where $a = (a_1, a_2, \ldots, a_n)^T$, $b = (b_1, b_2, \ldots, b_n)^T$ are two non-zero vectors, and t and τ are two non-zero constants. As we have seen in subsection 4.5.2, such A and B can be expressed as

$$A = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \alpha_1 a & \alpha_2 a & \cdot & \alpha_n a \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}, \qquad B = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \beta_1 b & \beta_2 b & \cdot & \beta_n b \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix},$$

where α_i and β_i $(i = 1, \ldots, n)$ are real values. Now consider matrix $C = A + B$. Obviously,

$$C = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \alpha_1 a + \beta_1 b & \alpha_2 a + \beta_2 b & \cdot & \alpha_n a + \beta_n b \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}.$$

We now show that such matrix C has a rank of two or less. Starting with the first two columns, we first find two columns that are linearly independent (if we cannot find two linearly independent columns, then the rank of C is less than 2). Without loss of generality suppose the first two columns $\alpha_1 a + \beta_1 b$ and $\alpha_2 a + \beta_2 b$ are linearly independent. It means that vectors $(\alpha_1, \beta_1)^T$ and $(\alpha_2, \beta_2)^T$ are not proportional, hence

$$\det \begin{bmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{bmatrix} \neq 0.$$

Now any other column of C can be expressed as a linear combination of the first two columns. For example. the third column $\alpha_3 a + \beta_3 b$ can be expressed as

$$\alpha_3 a + \beta_3 b = \gamma_1(\alpha_1 a + \beta_1 b) + \gamma_2(\alpha_2 a + \beta_2 b)$$

with certain real numbers γ_1 and γ_2. Why? In fact to find such γ_1 and γ_2, we can solve the equations

$$\begin{aligned} \alpha_3 &= \gamma_1\alpha_1 + \gamma_2\alpha_2 \\ \beta_3 &= \gamma_1\beta_1 + \gamma_2\beta_2 \end{aligned}$$

As the coefficient matrix of the above equations

$$\begin{bmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{bmatrix}$$

is nonsingular, the equations must have a solution (γ_1, γ_2).

Therefore, we know that the rank of C is two.

Section 4.8 Appendix 4 - Motivation of the BFGS Update Formula

We want to have a symmetric rank-2 update formula. So we consider the following form:

$$B_{k+1} = B_k + \alpha a a^T + \beta b b^T,$$

where α and β are two unknown scalars, and a and b are two vectors. We choose:

$$a = y_k, \quad b = B_k s_k,$$

where $s_k = x^{k+1} - x^k$ and $y_k = \nabla f(x^{k+1}) - \nabla f(x^k)$. As we know, B_{k+1} should satisfy the quasi-Newton equation:

$$B_{k+1} s_k = y_k.$$

We observe what α and β should be in order to meet the above condition.

Now

$$B_k s_k + \alpha y_k y_k^T s_k + \beta (B_k s_k)(B_k s_k)^T s_k = y_k,$$

that is,

$$\begin{aligned} \alpha (y_k^T s_k) y_k + [1 + \beta (B_k s_k)^T s_k] B_k s_k &= y_k \\ &= 1 \cdot y_k + 0 \cdot B_k s_k. \end{aligned}$$

To meet the above equation, the easiest way is to ask:

$$\alpha (y_k^T s_k) = 1,$$

and

$$1 + \beta (s_k^T B_k s_k) = 0,$$

i.e.,

$$\alpha = \frac{1}{y_k^T s_k}$$

and

$$\beta = -\frac{1}{s_k^T B_k s_k}.$$

So, the update formula becomes

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

which is just the BFGS formula.

Section 4.8 Appendix 5 - Why $y_k^T s_k > 0$ under exact line search for quasi-Newton method?

We know that

$$B_k p_k = -\nabla f(x^k), \quad s_k = x^{k+1} - x^k, \quad y_k = \nabla f(x^{k+1}) - \nabla f(x^k),$$

and

$$x^{k+1} = x^k + \alpha_k p_k,$$

where α_k is the minimum solution to the line search

$$\min_{\alpha>0} \phi(\alpha) = f(x^k + \alpha p_k).$$

So,

$$\begin{aligned} \phi'(\alpha_k) &= \nabla f(x^k + \alpha_k p_k)^T p_k \\ &= \nabla f(x^{k+1})^T p_k \\ &= \frac{1}{\alpha_k} \nabla f(x^{k+1})^T s_k = 0. \end{aligned}$$

Hence

$$\nabla f(x^{k+1})^T s_k = 0.$$

Now we see that

$$\begin{aligned} y_k^T s_k &= [\nabla f(x^{k+1}) - \nabla f(x^k)]^T s_k \\ &= -\nabla f(x^k)^T s_k \\ &= -\alpha_k \nabla f(x^k)^T p_k \\ &= -\alpha_k \nabla f(x^k)^T (-B_k^{-1} \nabla f(x^k)) \\ &= \alpha_k \nabla f(x^k)^T B_k^{-1} \nabla f(x^k) \\ &> 0, \end{aligned}$$

as B_k is positive definite.