OR4030 OPTIMIZATION Chapter 1
Some Optimization Models

1.1 Least Squares Data Fitting

Suppose that in a model, the relationship between two variables b and t are governed by a quadratic function:

$$b(t) = x_1 + x_2 t + x_3 t^2.$$

We need to use some experimental data $(t_1, b_1), (t_2, b_2), \ldots, (t_m, b_m)$ to determine x_1, x_2 and x_3.

If $m = 3$, and

$$\begin{cases} (t_1, b_1) = (2, 1) \\ (t_2, b_2) = (3, 6) \\ (t_3, b_3) = (5, 4) \end{cases}$$

then

$$\begin{bmatrix} 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 5 & 25 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 6 \\ 4 \end{bmatrix}.$$

Solving the above linear equations gives

$$(x_1, x_2, x_3) = (-21, 15, -2),$$

so that

$$b(t) = -21 + 15t - 2t^2.$$

It is common that we collect more data points than the number of variables in the model. And it is expected that each of the measurements will be in error.

For example, if we also have

$$(t_4, b_4) = (7, -15),$$

and if we use the first three data points to determine (x_1, x_2, x_3) and $b(t)$, then

$$b(7) = -21 + 15 \cdot 7 - 2 \cdot 7^2 = -14 \neq -15.$$

Since each data point may have some error, we must use them "collectively"

Define the residual vector

$$r = b - Ax = \begin{bmatrix} b_1 - (x_1 + x_2 t_1 + x_3 t_1^2) \\ b_2 - (x_1 + x_2 t_2 + x_3 t_2^2) \\ \cdot \\ \cdot \\ \cdot \\ b_m - (x_1 + x_2 t_m + x_3 t_m^2) \end{bmatrix},$$

where

$$r = \begin{bmatrix} r_1 \\ \cdot \\ \cdot \\ \cdot \\ r_m \end{bmatrix} \quad \text{and} \quad A = \begin{bmatrix} 1 & t_1 & t_1^2 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 1 & t_m & t_m^2 \end{bmatrix},$$

and make r as small as possible. As r is a vector, we specify the request of making r small as letting some norm of r small.

Least Squares Data Fitting

Least squares data fitting is the most commonly used approach to make r small, i.e., to make $\|r\|_2$ the smallest:

$$\min_x \sum_{i=1}^{m} r_i^2 = \sum_{i=1}^{m} [b_i - (x_1 + x_2 t_i + x_3 t_i^2)]^2.$$

The right hand side of the above equality is a non-linear function of x. Hence, the problem is a nonlinear optimization problem.

In this model, in the function

$$b(t) = x_1 + x_2 t + x_3 t^2,$$

all x_j occur linearly, and hence the model is called a linear least squares model.

Other Examples of Linear Model

$$b(t) = x_1 + x_2 \sin t + x_3 \sin 2t + \ldots + x_{k+1} \sin kt.$$

or

$$b(t) = x_1 + \frac{x_2}{1 + t^2}.$$

Note that even though the functions of t are nonlinear, variables x_1, x_2 and x_3 appear linearly.

Nonlinear Least Squares Model: Some $x_1, x_2, \ldots x_k$ in function $b(t)$ occur nonlinearly.

Examples Consider the following models where

$$b(t) = x_1 + x_2 e^{x_3 t} + x_4 e^{x_5 t}, \tag{1}$$

or

$$b(t) = x_1 + \frac{x_2}{1 + x_3 t^2}.$$

For the model (1), the least squares problem is

$$\min_x \sum_{i=1}^{m} r_i(x)^2,$$

where

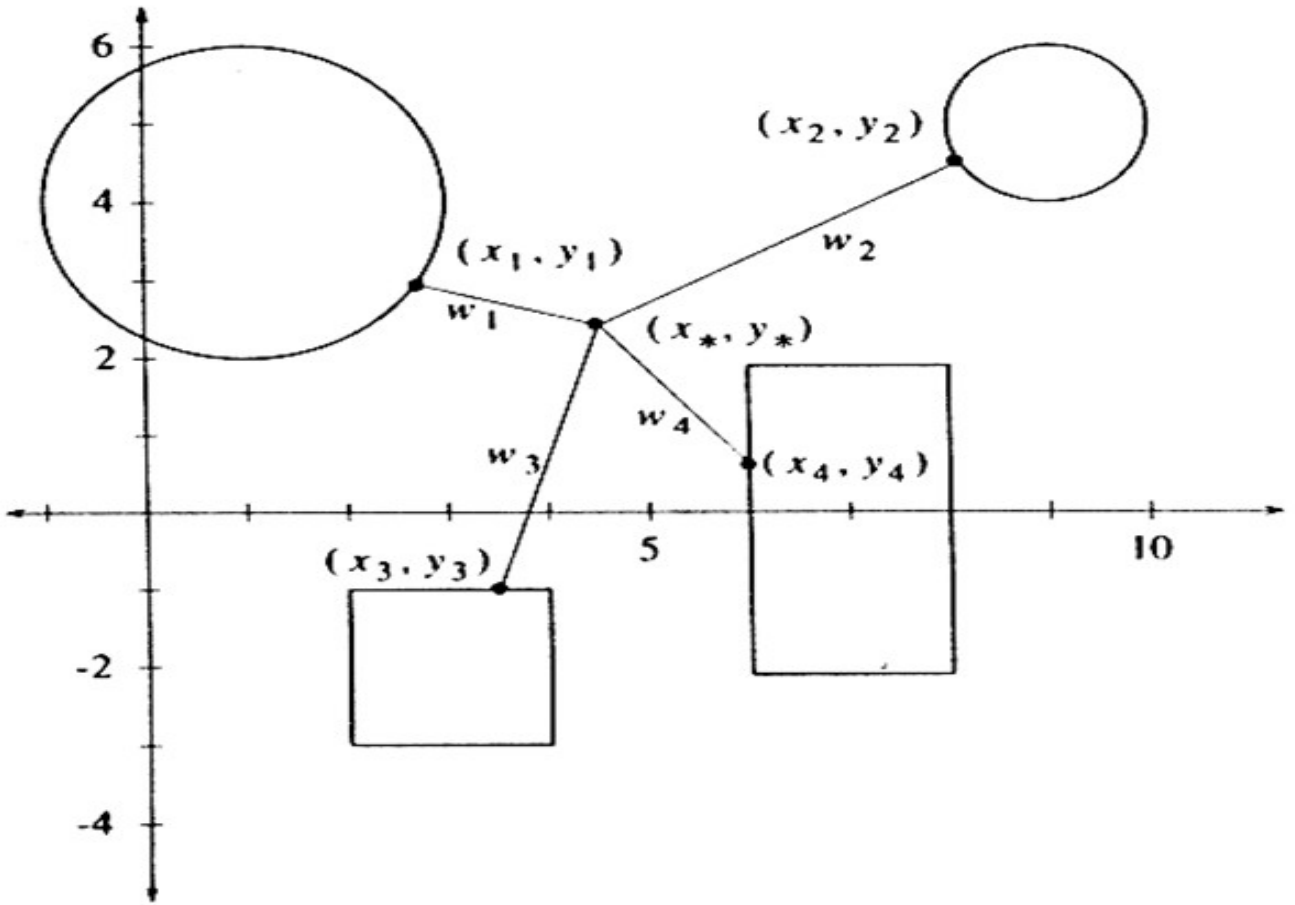
$$r_i(x) = b_i - (x_1 + x_2 e^{x_3 t_i} + x_4 e^{x_5 t_i}).$$

The minimization problem is a nonlinear optimization problem. Note that no matter the least squares (LS) problem is a linear LS model, or a nonlinear LS model, it is always a nonlinear optimization problem. But we shall see later that linear LS problem is easier to solve.

1.2 Nonlinear Programming

In this course, nonlinear programming means optimization of a function $f(x)$ subject to some constraints of the type $g_i(x) = 0$ or $h_j(x) \leq 0$, where at least one function (either the objective function f , or a constraint function g_i or h_j) is nonlinear.

Example 1 The figure below gives locations of two round buildings and two rectangular shaped buildings. Find a point (x^*, y^*) such that the total distance from (x^*, y^*) to all buildings is minimum. (here the distance from a point to a building means the distance from the point to the closest point in the building)


6
4
2
-2
-4
5
10
(x_1, y_1)
(x_2, y_2)
(x_3, y_3)
(x_4, y_4)
(x_*, y_*)
w_1
w_2
w_3
w_4

The model can be formulated as:

$$
\begin{aligned}
\min \quad & w_1 + w_2 + w_3 + w_4 \\
s.t. \quad & w_i = \sqrt{(x_i - x^*)^2 + (y_i - y^*)^2}, \quad i = 1, 2, 3, 4 \\
& (x_1 - 1)^2 + (y_1 - 4)^2 \leq 4, \\
& (x_2 - 9)^2 + (y_2 - 5)^2 \leq 1, \\
& 2 \leq x_3 \leq 4, \quad -3 \leq y_3 \leq -1, \\
& 6 \leq x_4 \leq 8, \quad -2 \leq y_4 \leq 2.
\end{aligned}
$$

In the model (x_i, y_i) represents the closest point in building i $(i = 1, 2, 3, 4)$ to the point (x^*, y^*). They are also unknown variables, and should be determined by this minimization model.

Example 2 Consider a portion (or say, a segment) of a sphere shown in the graph below. As we know, the volume of the segment is $\pi h^2(r - \frac{h}{3})$, and its surface area is $2\pi rh$. (without including the bottom part)

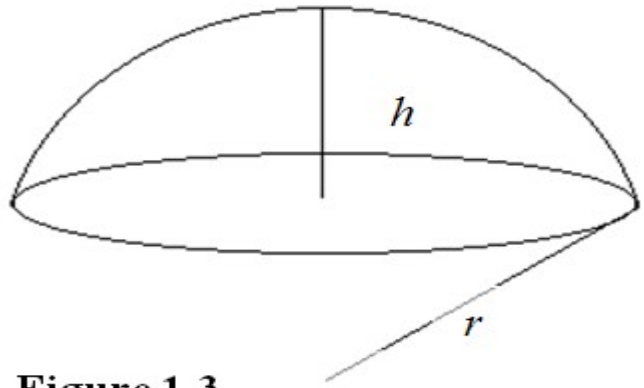


Figure 1.3
Archimedes' problem

The problem is how to choose r and h so as to maximize the volume of the segment, but where the surface area A of the segment is fixed., i.e., A is a given constant.

This problem can be formulated as the following model:

$$\begin{aligned} \max \quad & v(r,h) = \pi h^2 (r - \frac{h}{3}) \\ s.t. \quad & 2\pi rh = A. \end{aligned}$$

Here $v(r,h) = \pi h^2 (r - \frac{h}{3})$ is a nonlinear objective function of r and h, and $2\pi rh = A$ is a nonlinear equality constraint of r and h. So, in this problem both objective function and the constraint are nonlinear.

Example 3 (Portfolio selection). We know that
security: stock, bond, ... ;
portfolio: a collection of securities.

For each security j, its return r_j is in fact a random variable, and let

- μ_j: the expected annual return, i.e., the mean of the random variable r_j;
- v_{jj}: the variance of the random variable r_j
  (large variance means the return of security j is unstable);
- v_{ij}: the covariance of the returns between securities i and j
  (large v_{ij} means the performances of the two securities are closely related).

It is well known that diversifying investment can reduce risk. So, in stead of buying a single security, we now establish a portfolio by investing on several securities. Suppose all together there are N available securities, and denote

- x_j: the number of shares of security j to be invested;
- a_j: current price per share of security j.

Suppose a portfolio P consists of

$$x_j \text{ units of security } j, \quad j = 1, 2, \ldots, N,$$

and let

$\bar{r}_P$: the expected return of the portfolio P, which equals

$$\bar{r}_P = \sum_{j=1}^{N} x_j \mu_j = \mu^T x;$$

σ_P: the standard deviation of the portfolio P, i.e.,

$$\sigma_P = (\sum_{i=1}^{N} \sum_{j=1}^{N} x_i x_j v_{ij})^{\frac{1}{2}} = (x^T V x)^{\frac{1}{2}},$$

where

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1N} \\ v_{21} & v_{22} & \cdots & v_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ v_{N1} & v_{N2} & \cdots & v_{NN} \end{bmatrix}.$$

In the above model, σ_P or $\sigma_P^2 = x^T V x$ can measure the risk level.

Generally, investors want to have

- large $\bar{r}_P$ (high expected return)
- small σ_P^2 (low risk)

for their portfolios. But these are two conflict targets.

Let the total available budget be B, the portfolio selection model can be formulated in several ways:

(1) Maximize a combination of $\mu^T x$ and $-x^T V x$:

$$\begin{aligned} \max \quad & \mu^T x - \alpha x^T V x \\ s.t. \quad & a^T x \leq B \\ & x \geq 0, \end{aligned}$$

where α is a weight coefficient to measure the relative importance of the two parts $\mu^T x$ and $-x^T V x$. Note that

- $\alpha = 0$: risk is totally ignored;
- large value of α: risk is heavily concerned.

(2) Minimize $x^T Vx$ while setting a lower bound for the return:

$$\begin{aligned} \min \quad & x^T Vx \\ s.t. \quad & \mu^T x \geq p \\ & a^T x \leq B \\ & x \geq 0, \end{aligned}$$

where p stands for the minimum acceptable annual return.

(3) Maximize $\mu^T x$ while setting an upper bound for the risk level:

$$\begin{aligned} \max \quad & \mu^T x \\ s.t. \quad & x^T Vx \leq q \\ & a^T x \leq B \\ & x \geq 0, \end{aligned}$$

where q stands for the maximum acceptable risk level.

The model may have more restrictions. For example, if no more than 5% of total investment can be put into each individual security, then we should add constraints:

$$\frac{a_j x_j}{B} \leq 0.05, \quad j = 1, 2, \ldots, N$$

i.e.,

$$a_j x_j \leq 0.05B, \quad j = 1, 2, \ldots, N.$$

The three models are all nonlinear programming problems.

The above method for portfolio selection was first suggested by H.M. Markowitz in 1959, who won 1990 Nobel Prize for economics due to this contribution to the development of finance and economics.

Example 4 (Minimize the total travel time).
Let

- t_{ij}: the travel time between points i and j when the traffic is light;
- x_{ij}: the number of cars on the road between points i and j;
- c_{ij}: the capacity of the road between i and j;
- α_{ij}: a coefficient to measure how rapidly the travel time increases as the traffic gets heavier.

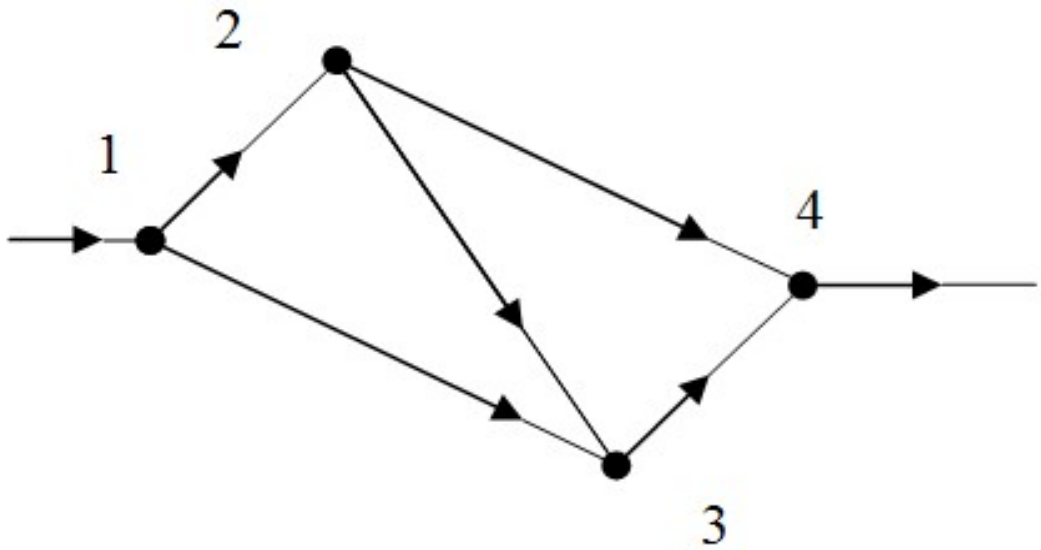
According to transportation science, the travel time between points i and j when the number of cars on the road is x_{ij} can be estimated by the formula:

$$T_{ij}(x_{ij}) = t_{ij} + \alpha_{ij} \frac{x_{ij}}{1 - \frac{x_{ij}}{c_{ij}}}.$$

Two extreme cases:
when $x_{ij} = 0$, the travel time is t_{ij};
when $x_{ij} = c_{ij}$, the travel time tends to $+\infty$.

Now suppose in the network below (see Figure 1.4), the total number of cars through the network (i.e., from point 1 to point 4) is X. How should we distribute the cars traveling on each road, so that the total travel time of all cars is minimized?

Figure 1.4
Traffic network



The model can be expressed as

$$
\begin{aligned}
\min \quad & f(x) = \sum_{(i,j)} x_{ij}\, T_{ij}(x_{ij}) \\
s.t. \quad & x_{1,2} + x_{1,3} = X \\
& x_{2,3} + x_{2,4} - x_{1,2} = 0 \\
& x_{3,4} - x_{1,3} - x_{2,3} = 0 \\
& x_{2,4} + x_{3,4} = X \\
& T_{ij}(x_{ij}) = t_{ij} + \alpha_{ij} \frac{x_{ij}}{1 - \frac{x_{ij}}{c_{ij}}}, \text{for } ij = 12, 13, 23, 24, 34 \\
& x_{ij} \geq 0, \text{for } ij = 12, 13, 23, 24, 34.
\end{aligned}
$$

This problem is again a nonlinear programming problem as the functions $T_{ij}(x_{ij})$ and $f(x)$ are nonlinear functions.

Example 5 (Maximum likelihood estimation).
The method of maximum likelihood has been regarded as one of the most effective methods of estimating distribution parameters.

Let $x^1, \ldots x^N$ be a set of samples from a population with probability density function $g(x, \theta)$, where $\theta = (\theta_1, \ldots, \theta_s)^T$ are distribution parameters. $\theta \in D$, where D is the parameter space in R^s.

We need to estimate the values of $\theta_1, \ldots, \theta_s$ by the samples. For the purpose, we introduce the likelihood function

$$L(\theta) = \prod_{i=1}^{N} g(x^i, \theta)$$

and find the maximum likelihood estimate $\hat{\theta}$ which maximizes the function $L(\theta)$:

$$L(\hat{\theta}) = \max_{\theta \in D} L(\theta).$$

or equivalently, we can maximize the logarithmic likelihood function

$$LL(\theta) = \sum_{i=1}^{N} \log g(x^i, \theta).$$

A special probability density function is a Weibull distribution with three parameters: location parameter δ, scale parameter β, and shape parameter α:

$$g(x;\delta,\beta,\alpha) = \frac{\alpha}{\beta^{\alpha}}(x-\delta)^{\alpha-1} exp\{-[(x-\delta)/\beta]^{\alpha}\}.$$

We see that maximizing its likelihood and logarithmic likelihood functions

$$L(\delta,\beta,\alpha) = \prod_{i=1}^{N} g(x^i;\delta,\beta,\alpha),$$

and

$$LL(\delta,\beta,\alpha) = \sum_{i=1}^{N} \log g(x^i;\delta,\beta,\alpha)$$

are both nonlinear optimization problems.

Example 6 (A two-class classification problem).

Suppose there is a set of historical data $\{(x^1, y_1), (x^2, y_2), \ldots, (x^N, y_N)\}$, where each $x^i \in R^m$ and $y_i \in R^1$, $i = 1, \ldots, N$. The m-dimensional vectors x represent some measurable factors, whereas each y has only two possible values: 1 or -1 which refers to two possible outcomes: either the result is in class one ($y = 1$) or in class two ($y = -1$).

We call these $\{(x^i, y_i)\}$ training data, and we want to find a line (if m=2), or a plane (if m=3), or generally a hyperplane in space R^m

$$(w, x) + b = 0$$

that separates the two classes of data points (here the symbol (w, x) represents the scalar product of the two vectors w and x).

That is,
$(w, x^i) + b > 0$, if the corresponding $y_i = 1$;
$(w, x^i) + b < 0$, if the corresponding $y_i = -1$.

The plane is called a separating plane. Once such a plane is found, if later we obtain another input x, what should the associate y be, 1 or -1?

We can answer that

$$y = sign\{(w, x) + b\},$$

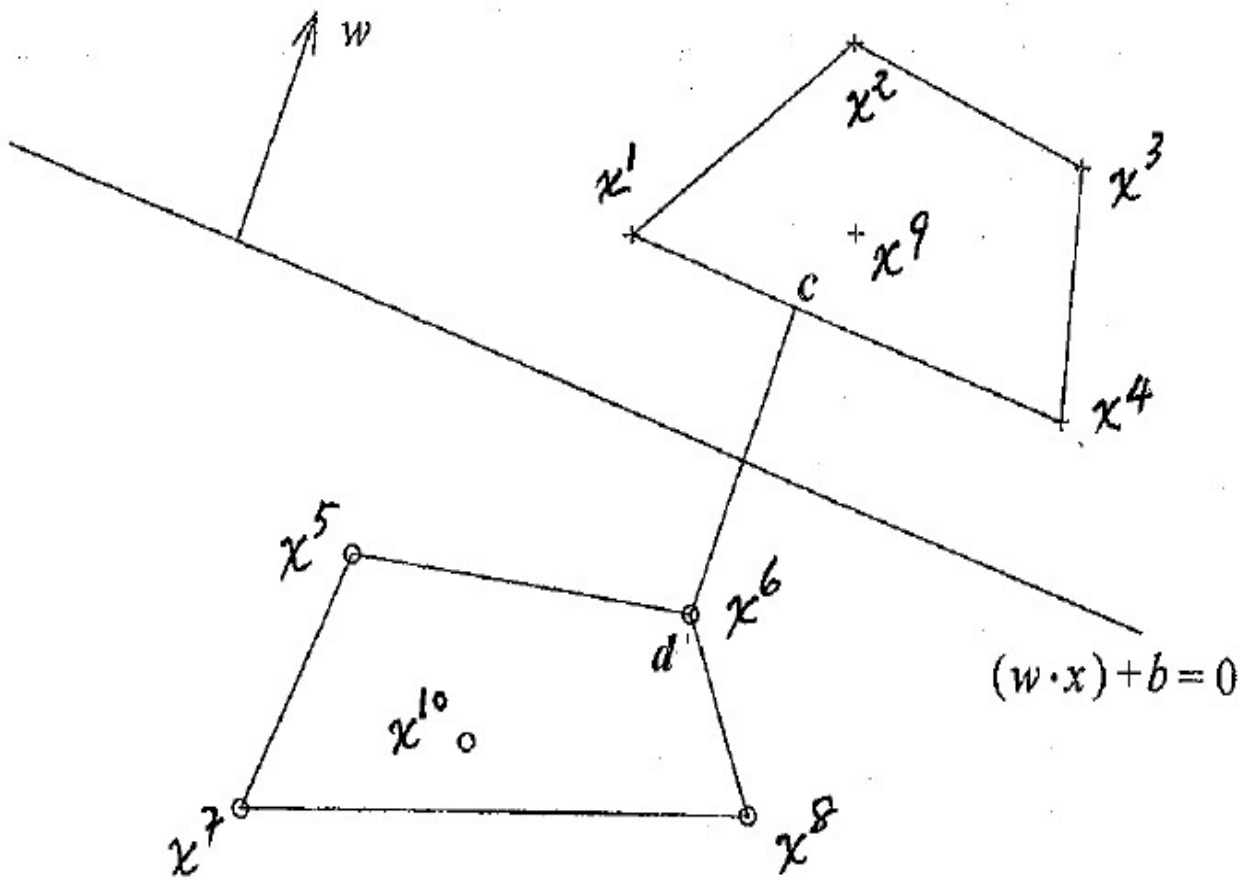
that is, $y = 1$ if $(w, x) + b > 0$ and -1 otherwise.

There are several methods to obtain such a separating plane. Here we introduce a method, called halving the shortest distance method. Its main idea is

1. find the closest points between convex hulls of the two classes of points, say points c and d;
2. link the two points to obtain line segment (c, d) and obtain its middle point M;
3. then take the plane vertical to line segment (c, d) and passing through point M as the wanted separating plane.

(we may use the graph on next page to understand the idea)

The method can be realized by the algorithm on page 31.


w
x1
x2
x3
x4
x5
x6
x7
x8
x9
x10
c
d
(w·x)+b=0

1. Formulate and solve the following optimization problem:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}||\sum_{y_i=1}\alpha_i x^i - \sum_{y_i=-1}\alpha_i x^i||^2, \\ s.t. \quad & \sum_{y_i=1}\alpha_i = 1, \quad \sum_{y_i=-1}\alpha_i = 1, \\ & 0 \le \alpha_i \le 1, \quad i = 1, \ldots, N \end{aligned}$$

   and obtain an optimal solution $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2, \ldots \hat{\alpha}_N)^T$.

2. Obtain the closest points between the two convex hulls:

$$c = \sum_{y_i=1}\hat{\alpha}_i x^i, \quad d = \sum_{y_i=-1}\hat{\alpha}_i x^i.$$

3. Construct the separating hyperplane

$$(\hat{w}, x) + \hat{b} = 0,$$

   where

$$\hat{w} = c - d = \sum_{i=1}^{N} y_i \hat{\alpha}_i x^i, \quad \hat{b} = -\frac{1}{2}((c-d),(c+d)).$$

We see that we have used the training set of data to establish the separating plane.

To check if a new input x belongs to class $y = 1$ or class $y = -1$, we just let

$$y = sign\{(\hat{w}, x) + \hat{b}\}.$$

This type of methods to find separating hyperplane or surface is called **support vector machine** which forms an important part of the subject **data mining**. Actually the key step is step 1, i.e., to solve a nonlinear optimization problem (more particularly, a quadratic optimization problem).

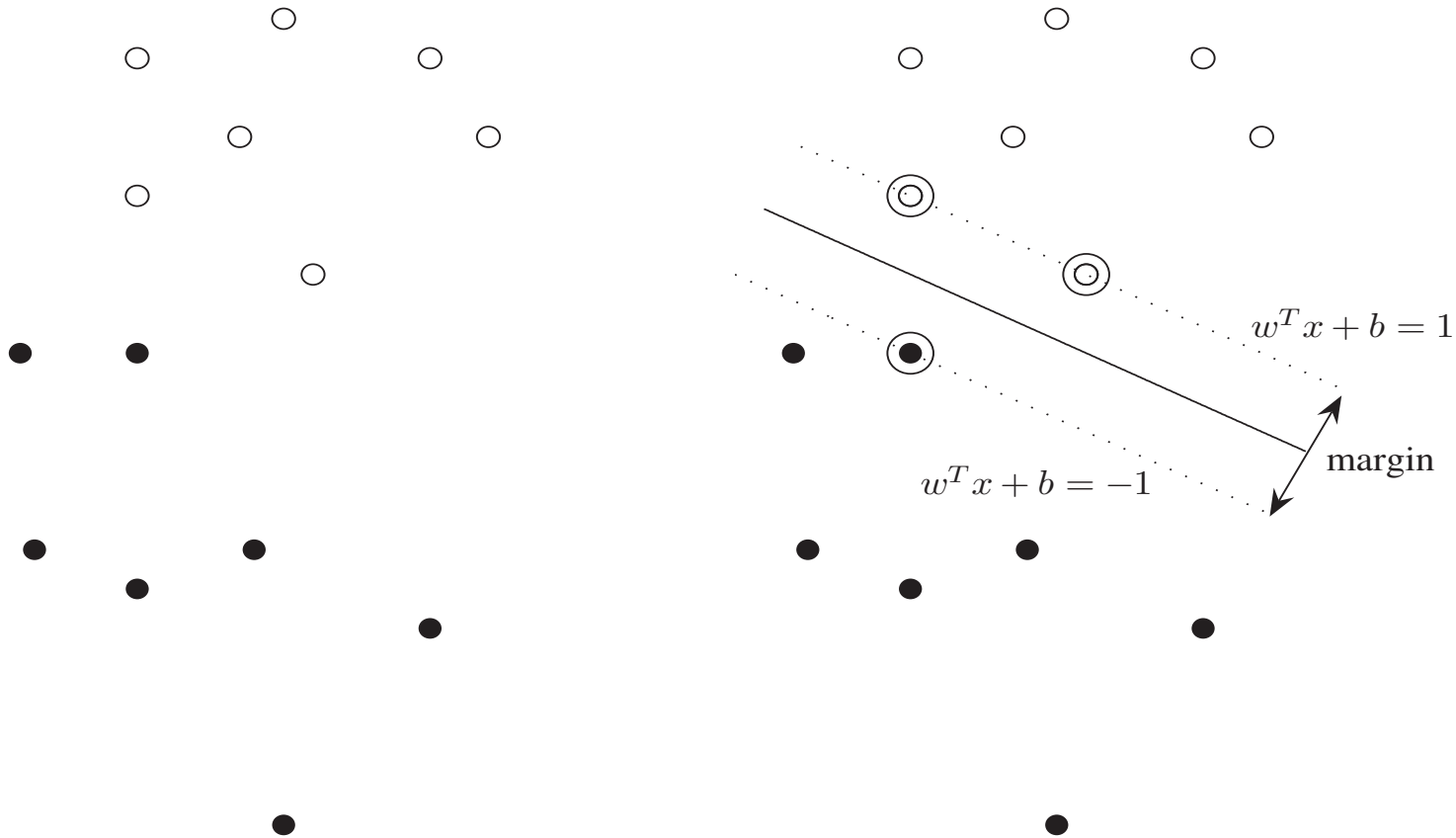
Another Model for the Classification Problem

To better separate the two groups of data points, we wish to have a band area centered at the separating plane $(\hat{w}, x_i) + \hat{b} = 0$, such that no data points would be in the area, see the graph on next page. That is, we wish that

$$\begin{aligned} (\hat{w}, x_i) + \hat{b} &\geq +k, \qquad \text{for } y_i = +1, \\ (\hat{w}, x_i) + \hat{b} &\leq -k, \qquad \text{for } y_i = -1. \end{aligned}$$

Without loss of generality, we may assume the number k to be 1, because if there is any number k satisfying the above two inequalities, then by changing $\hat{w}$ to a new one $w = \hat{w}/k$ and $\hat{b}$ to a new constant $b = \hat{b}/k$, the data points would satisfy

$$\begin{aligned} (w, x_i) + b &\geq +1, \qquad \text{for } y_i = +1, \\ (w, x_i) + b &\leq -1, \qquad \text{for } y_i = -1. \end{aligned}$$


$w^T x + b = 1$
$w^T x + b = -1$
margin

The width of the band is called its margin (see the graph). According to the fact that the normal directions of these planes are the same, which is w, we can obtain that the margin is $2/\|w\|$. If the margin is large, the two sets of data points would be more clearly separated. So, we wish to maximize $1/\|w\|$, or equivalently minimize $\|w\|^2$. Now the second model to determine the separating plane is:

$$\begin{array}{lll} \min & \|w\|^2 & \\ s.t. & (w, x_i) + b \geq +1, & \text{for } y_i = +1, \\ & (w, x_i) + b \leq -1, & \text{for } y_i = -1. \end{array}$$

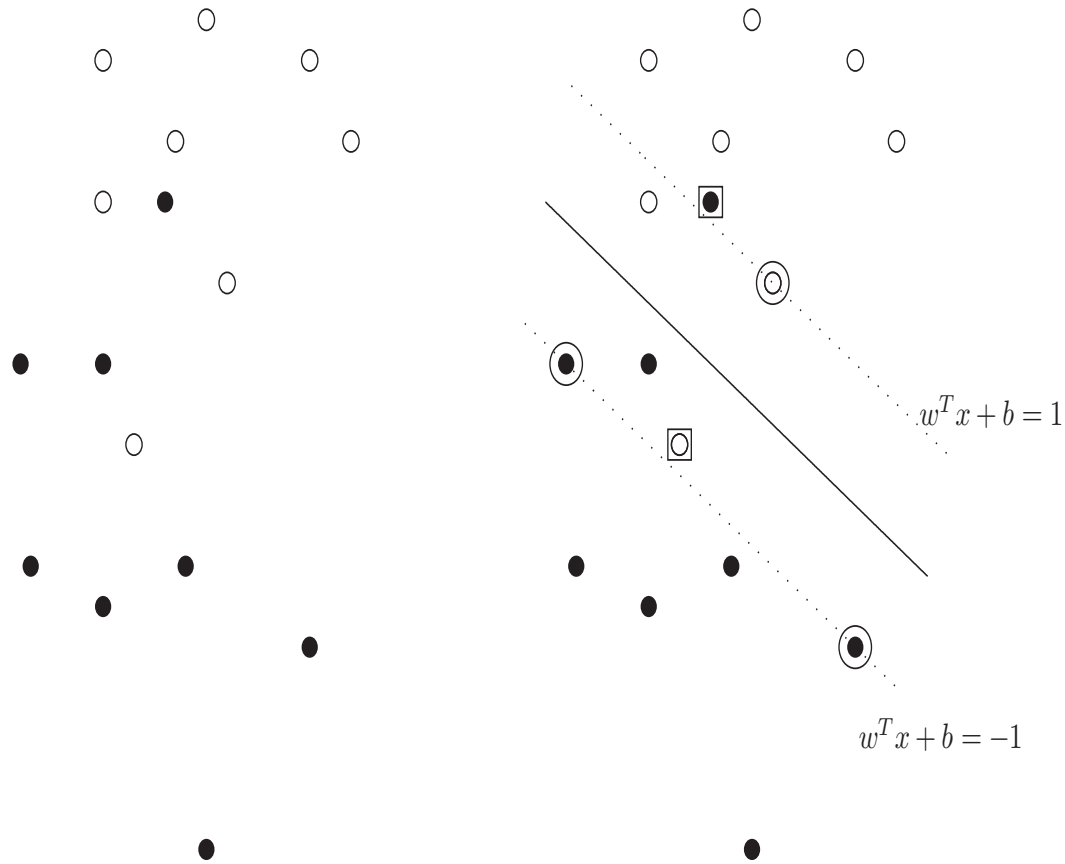
It can be proved that the two methods actually obtain the same plane.

Approximately Linearly Separable Case

So far we have assumed that the two data sets are linearly separable, i.e., there exists a plane to separate them exactly. But it is possible that such plane does not exist, see the next graph. Under such case, we need to relax the request, and let the plane separate as more data points correctly as possible, but with some misclassified data points.

We allow some points to violate the separation constraints, but impose a penalty for the violation. Let the nonnegative amount ξ_i be the amount by which the point x_i violates the constraint:

$$\begin{aligned} (w, x_i) + b &\geq +1 - \xi_i, \qquad \text{for } y_i = +1, \\ (w, x_i) + b &\leq -1 + \xi_i, \qquad \text{for } y_i = -1. \end{aligned}$$


$w^T x + b = 1$
$w^T x + b = -1$

The total violation can be measured by $\sum_i \xi_i$. As we want to minimize both $\|w\|^2$ and $\sum_i \xi_i$, the model can be formulated as:

$$\begin{aligned} \text{min} \quad & \|w\|^2 + C \sum_i \xi_i \\ s.t. \quad & (w, x_i) + b \geq +1 - \xi_i, \quad \text{for } y_i = +1, \\ & (w, x_i) + b \leq -1 + \xi_i, \quad \text{for } y_i = -1, \\ & \text{all } \xi_i \geq 0. \end{aligned}$$

C is a control parameter. The larger C is, the larger the penalty for violating the separation is imposed. In practice, we may try several values of C to get a suitable one.

Note that if in the solution, $\sum_i \xi_i > 0$, it means that there is no true separating plane, that is, the two data sets are in fact linearly inseparable, and the obtained plane $(w, x) + b = 0$ is only an approximate separating plane with some errors.

OR4030 OPTIMIZATION Chapter 2
Mathematical Background for Nonlinear Optimization
2.1 Sets

Neighborhood

The *ε-neighborhood* of a point $x = (x_1, x_2, \ldots, x_n)^T \in \Re^n$ is the set of points inside the ball with center x and radius $\varepsilon > 0$:

$$N_\varepsilon(x) = \{y \in \Re^n : \|y - x\| < \varepsilon\},$$

where $\|y - x\| = \sqrt{\sum_{j=1}^{n}(y_j - x_j)^2}$ for $y = (y_1, y_2, \ldots, y_n)^T$.

Open Sets

A set S is called *open* if around every point in S there is a neighborhood that is contained in S. For example, $\Re^n$, $\emptyset$ and the open interval (a, b) in $\Re^1$ (or simply written as $\Re$) are open sets.

Closed Sets

A set S in $\Re^n$ is called *closed* if its complement S' (i.e., the set $\Re^n \setminus S$) is an open set. For example, $\Re^n$, $\emptyset$ and the closed interval $[a, b]$ in $\Re$ are closed sets.

Bounded Sets

A set is *bounded* if it can be contained in a ball of a sufficiently large radius. For example, the open interval (a, b) and the closed interval $[a, b]$ in $\Re$ are both bounded sets (here a and b are two finite numbers).

Compact Sets

A set is *compact* if it is both closed and bounded. For example, the closed interval $[a, b]$ is a compact set.

2.2 Convex Sets

A set S in $\Re^n$ is *convex* if for any elements $\bar{x}$ and $\hat{x}$ of S,

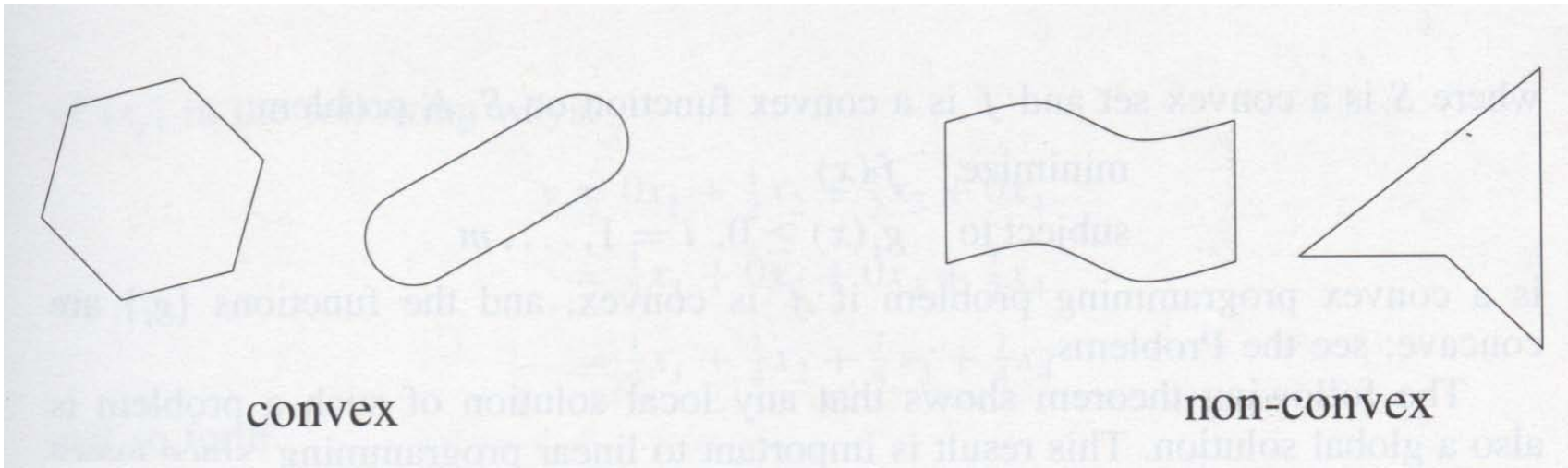
$$\lambda\bar{x} + (1-\lambda)\hat{x} \in S$$

for all values of λ such that $0 \leq \lambda \leq 1$.

The set

$$\begin{aligned} & \{\lambda\bar{x} + (1-\lambda)\hat{x} \mid 0 \leq \lambda \leq 1\} \\ = \; & \{\hat{x} + \lambda(\bar{x} - \hat{x}) \mid 0 \leq \lambda \leq 1\} \end{aligned}$$

is the line segment joining $\bar{x}$ and $\hat{x}$. So, S is a convex set if and only if for every pair of $\bar{x}$ and $\hat{x}$ in S, the line segment joining $\bar{x}$ and $\hat{x}$ also lies in S.



Convex and Non-convex Sets

It is easy to check by the definition that the following sets are all convex sets.

- Any line:

  $$L = \{x + \lambda v \mid \lambda \in \Re\}$$

  for a given point (or say a vector) x and a given vector v in $\Re^n$.

- Any half line or ray:

  $$L = \{x + \lambda v \mid \lambda \geq 0\}$$

  for a given point x and a given vector v in $\Re^n$.

- Any closed half-space

$$F^+ = \{y \in \Re^n \mid \bar{x}^T y \geq \alpha\}$$

for a given vector $\bar{x}$ in $\Re^n$ and a given real number α.

- Any open half-space

$$F = \{y \in \Re^n \mid \bar{x}^T y > \alpha\}$$

for a given vector $\bar{x}$ in $\Re^n$ and a given real number α.

- Any closed ball

$$B^+(\bar{x}, r) = \{y \in \Re^n \mid \|y - \bar{x}\| \leq r\}$$

or open ball

$$B(\bar{x}, r) = \{y \in \Re^n \mid \|y - \bar{x}\| < r\},$$

where $\bar{x}$ is a given point in $\Re^n$, and r is a positive number.

- The intersection of a number of convex sets C_i in $\Re^n$, $i = 1, 2, \ldots, k$:

$$C = \{\bigcap_{i=1}^{k} C_i \mid C_i, i = 1, \ldots, k, \text{ are convex sets}\}.$$

- The solution set to the linear equations

$$S = \{x \in \Re^n \mid Ax = b\},$$

where $A \in \Re^{m \times n}$ (i.e., a $m \times n$ real matrix), and $b \in \Re^m$.

- For p given points $x^1, x^2, \ldots, x^p$ in $\Re^n$, any vector

$$\sum_{i=1}^{p} \lambda_i x^i = \lambda_1 x^1 + \lambda_2 x^2 + \cdots + \lambda_p x^p$$

is called a *convex combination of* $x^1, \ldots, x^p$, if all numbers λ_i satisfy that

$$\lambda_i \geq 0, \ \ i = 1, \ldots, p; \quad \text{and} \ \sum_{i=1}^{p} \lambda_i = 1.$$

For p given points $x^1, x^2, \ldots, x^p$ in $\Re^n$, all their convex combinations:

$$C = \{\sum_{i=1}^{p} \lambda_i x^i \mid \lambda_i \geq 0, \sum_{i=1}^{p} \lambda_i = 1\}$$

form a convex set. This set is called the convex hull of points $x^1, x^2, \ldots, x^p$.

Note that for three points x^1, x^2 and x^3 in $\Re^2$, all their convex combinations form the triangular region with x^1, x^2 and x^3 as its vertexes.

Limits

The equation

$$\lim_{x \to c} f(x) = d$$

means that as the single variable x gets very close to the number c (but is not necessary equal to c), the value of $f(x)$ gets arbitrarily close to the value d.

It is also possible that

$$\lim_{x \to c} f(x)$$

may not exist.

Continuity

A function $f(x)$ is *continuous* at a point (or say a value) c if

$$\lim_{x \to c} f(x) = f(c).$$

If $f(x)$ is not continuous at c, we say that $f(x)$ is *discontinuous* (or has a discontinuity) at c.

Differentiation

The *derivative* of $f(x)$ at a point c is defined by

$$f'(c) = \frac{df(c)}{dx} = \left.\frac{df}{dx}\right|_{x=c} = \lim_{\Delta x \to 0} \frac{f(c + \Delta x) - f(c)}{\Delta x}$$

provided that the limit exists. When the limit exists, we say that $f(x)$ is *differentiable* at c. Note also that if $f(x)$ is differentiable at c, then $f(x)$ is continuous at c.

Higher Derivatives

Furthermore, we can define

$$f^{(2)}(c) = f''(c) = \frac{d^2 f(c)}{dx^2} = \left.\frac{d^2 f}{dx^2}\right|_{x=c}$$

to be the derivative of function $f'(x)$ at point $x = c$.

Similarly, we define (if it exists) $f^{(k)}(c)$ to be the derivative of $f^{(k-1)}(x)$ at $x = c$ for $k \geq 3$.

Continuously Differentiable Functions

- $f \in C^0(S)$
  $\sim$ f is a real-valued continuous function in its domain S.

- $f \in C^1(S)$
  $\sim$ f is a real-valued continuously differentiable function in S
  $\sim$ i.e., f' exists and is continuous everywhere in S.

- $f \in C^k(S)$, where $k = 1, 2, 3, \ldots$
  $\sim$ f is a real-valued kth-order continuously differentiable function in S
  $\sim$ i.e., $f^{(k)}$ exists and is continuous everywhere in its domain

Taylor's Theorem

Suppose $f^{(k+1)}$ exists for every point on the interval $[a, b]$. Let $c \in [a, b]$. Then for every $x \in [a, b]$, there exists h between c and x with

$$f(x) = P_k(x) + R_k(x),$$

where

$$\begin{aligned} P_k(x) &= f(c) + f'(c)(x-c) + \frac{f''(c)}{2!}(x-c)^2 + \cdots \\ &\quad + \frac{f^{(k)}(c)}{k!}(x-c)^k, \\ R_k(x) &= \frac{f^{(k+1)}(h)}{(k+1)!}(x-c)^{k+1}. \end{aligned}$$

Here

$P_k(x)$ is called the *kth-order Taylor series expansion* of f about c, and

$R_k(x)$ is called the *remainder term* (or *truncation error*) associated with $P_k(x)$.

The infinite series obtained by taking the limit of $P_k(x)$ as $k \to \infty$ is called the *Taylor series* of f about c.

We can use $P_k(x)$ to approximate $f(x)$: $f(x) \approx P_k(x)$ with an error of $R_k(x)$.

2.3.2 Differential Calculus of Several Variables

Partial Derivatives and the Gradient Vector

The *partial derivative* of f with respect to the variable x_j is

$$\frac{\partial f}{\partial x_j} = \lim_{\Delta x_j \to 0} \frac{f(x_1, \ldots, x_j + \Delta x_j, \ldots, x_n) - f(x_1, \ldots, x_j, \ldots, x_n)}{\Delta x_j}.$$

The *gradient* of f is defined by:

$$\nabla f = \begin{bmatrix} \dfrac{\partial f}{\partial x_1} \\ \vdots \\ \dfrac{\partial f}{\partial x_n} \end{bmatrix}.$$

Second-Order Partial Derivatives and the Hessian Matrix

We use the notation

$$\frac{\partial^2 f}{\partial x_i \, \partial x_j}$$

to denote a *second-order partial derivative* of f.

To find $\frac{\partial^2 f}{\partial x_i \, \partial x_j}$, we first find $\frac{\partial f}{\partial x_j}$ and then take its partial derivative with respect to x_i, i.e.,

$$\frac{\partial^2 f}{\partial x_i \, \partial x_j} = \frac{\partial}{\partial x_i} \left[\frac{\partial f}{\partial x_j} \right].$$

If the second-order partial derivatives exist and are everywhere continuous, then

$$\frac{\partial^2 f}{\partial x_i \, \partial x_j} = \frac{\partial^2 f}{\partial x_j \, \partial x_i}.$$

Suppose $f \in C^2(\Re^n)$. The *Hessian* matrix of f is defined by

$$\nabla^2 f = \begin{bmatrix} \dfrac{\partial^2 f}{\partial {x_1}^2} & \cdots & \dfrac{\partial^2 f}{\partial x_n \, \partial x_1} \\ \vdots & & \vdots \\ \dfrac{\partial^2 f}{\partial x_1 \, \partial x_n} & \cdots & \dfrac{\partial^2 f}{\partial {x_n}^2} \end{bmatrix}.$$

Note that the Hessian matrix of f is symmetric if all second order partial derivatives are continuous.

Taylor's Theorem

Suppose that $f \in C^3(S)$, where

$$S = \{x : a_j \leq x_j \leq b_j, j = 1, 2, \ldots, n\}.$$

Let $c = (c_1, c_2, \ldots, c_n)^T \in S$. For every $x \in S$,

$$f(x) \approx f(c) + \nabla f(c)^T (x - c) + \frac{1}{2}(x - c)^T \nabla^2 f(c)(x - c).$$

If we want to change the above approximate equality to an exact one, then

$$f(x) = f(c) + \nabla f(c)^T (x - c) + \frac{1}{2}(x - c)^T \nabla^2 f(\xi)(x - c),$$

where ξ is a point in the line segment linking x and c, but its exact location is unknown.

Example

For the function

$$f(x) = x_1^3 + 5x_1^2x_2 + 7x_1x_2^2 + 2x_2^3$$

where $x = (x_1, x_2)^T$, consider its approximate function values near the point $c = (-2, 3)^T$.

The gradient of this function is

$$\nabla f(x) = \begin{bmatrix} 3x_1^2 + 10x_1x_2 + 7x_2^2 \\ 5x_1^2 + 14x_1x_2 + 6x_2^2 \end{bmatrix}$$

and the Hessian matrix is

$$\nabla^2 f(x) = \begin{bmatrix} 6x_1 + 10x_2 & 10x_1 + 14x_2 \\ 10x_1 + 14x_2 & 14x_1 + 12x_2 \end{bmatrix}.$$

At the point c, $f, \nabla f$ and $\nabla^2 f$ become

$$f(c) = -20, \quad \nabla f(c) = \begin{bmatrix} 15 \\ -10 \end{bmatrix} \quad \text{and} \quad \nabla^2 f(c) = \begin{bmatrix} 18 & 22 \\ 22 & 8 \end{bmatrix}.$$

By Taylor's Theorem,

$$\begin{aligned} f(x) \quad &\approx \quad -20 + \begin{bmatrix} 15 & -10 \end{bmatrix} \begin{bmatrix} x_1 + 2 \\ x_2 - 3 \end{bmatrix} \\ &\qquad + \frac{1}{2} \begin{bmatrix} x_1 + 2 & x_2 - 3 \end{bmatrix} \begin{bmatrix} 18 & 22 \\ 22 & 8 \end{bmatrix} \begin{bmatrix} x_1 + 2 \\ x_2 - 3 \end{bmatrix} \end{aligned}$$

Thus, for example,

$$f(-1.9, 3.2) \approx -19.81.$$

The true value is $f(-1.9, 3.2) = -19.755$. So the approximation is accurate to three digits.

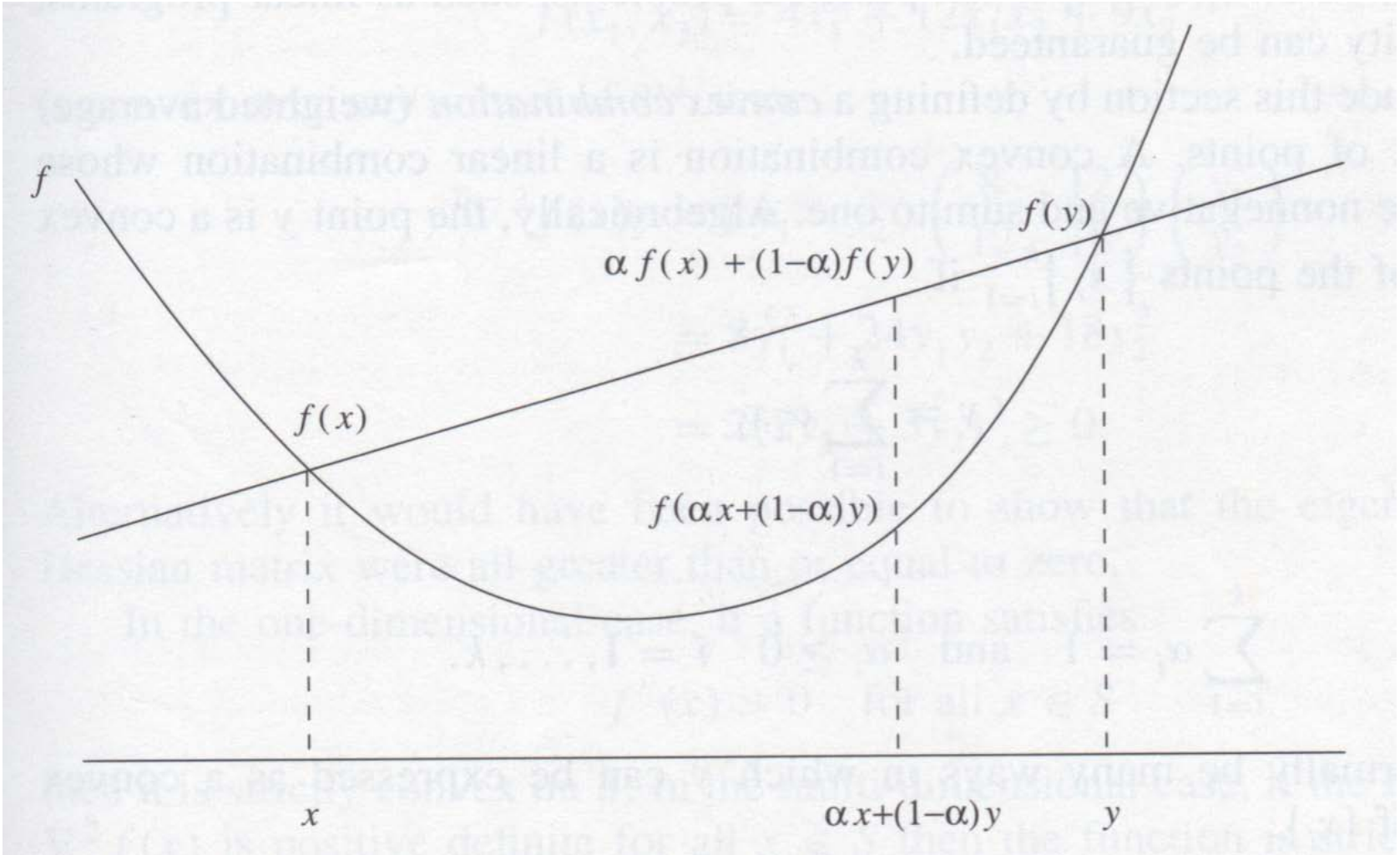
Definitions

A function $f(x)$ of a single real variable x, defined in an interval $[a, b]$, is a **convex function** if, for each pair of distinct values of x, say $\bar{x}$ and $\hat{x}$ in the interval, there holds

$$f(\lambda\bar{x} + (1-\lambda)\hat{x}) \leq \lambda f(\bar{x}) + (1-\lambda)f(\hat{x})$$

for all values of λ such that $0 < \lambda < 1$. It is a *strictly convex function* if $\leq$ can be replaced by $<$. It is a **concave function** (or a *strictly concave function*) if this statement holds when $\leq$ is replaced by $\geq$ (or by $>$).

Geometric meaning of a convex function: for each pair of $x < y$, over the interval $[x, y]$, the line segment joining the two points $(x, f(x))$ and $(y, f(y))$ either lies above, or coincides with the curve of f.



Graph of a Convex Function

Convexity Test for a Function of a Single Variable by Its First Order Derivatives

Consider a single variable function $f(x)$ that is defined in an interval S and possesses continuous first order derivative at each point x in the interval. Then $f(x)$ is

- *Convex* if and only if for each $\bar{x} \in S$,

$$f(y) \geq f(\bar{x}) + f'(\bar{x})(y - \bar{x}), \quad \text{for any } y \in S;$$

- *Strictly convex* if and only if for each $\bar{x} \in S$,

$$f(y) > f(\bar{x}) + f'(\bar{x})(y - \bar{x}), \quad \text{for any } y \in S \text{ and } y \neq \bar{x};$$

- *Concave* if and only if for each $\bar{x} \in S$,

$$f(y) \leq f(\bar{x}) + f'(\bar{x})(y - \bar{x}), \quad \text{for any } y \in S;$$

- *Strictly concave* if and only if for each $\bar{x} \in S$,

$$f(y) < f(\bar{x}) + f'(\bar{x})(y - \bar{x}), \quad \text{for any } y \in S \text{ and } y \neq \bar{x};$$

We will prove these properties in Section 2.4.2 for multi-variable convex functions.

Geometric Meaning of the Convexity Test. For a single variable function $f(x)$ over an interval S, the tangent line to the graph of $f(x)$ at $\bar{x}$ is given by

$$\{(x, y) \mid y = f(\bar{x}) + f'(\bar{x})(x - \bar{x})\}.$$

Hence, the first conclusion here says that $f(x)$ is a convex function if and only if every tangent line to $f(x)$ lies on or below the graph of $f(x)$;

The second conclusion here says that $f(x)$ is a strictly convex function if and only if every tangent line to $f(x)$ lies below the graph of $f(x)$ and contact the graph only at the point of tangency;

It is easy to state the geometric meaning of the last two conclusions.

Convexity Test for a Function of a Single Variable by Its Second Order Derivatives

Consider a single variable function $f(x)$ that is defined in an interval S and possesses continuous second order derivative at each point x in the interval. Then $f(x)$ is

- *Convex* if and only if $f''(x) \geq 0$ for all x in S;
- *Strictly convex* if $f''(x) > 0$ for all x in S;
- *Concave* if and only if $f''(x) \leq 0$ for all x in S;
- *Strictly concave* if $f''(x) < 0$ for all x in S.

We also leave the proof to the multi-variable case.

Note that in the second (and also the fourth) conclusion above, there is no "only if", that is, $f''(x) > 0$ is only a sufficient condition for strictly convex, but not a necessary condition.

Example. Function $f(x) = x^4$ is a strictly convex function for all $x \in (-\infty, \infty)$, but $f''(0) = 0$. So,

$$f \text{ is strictly convex over } (-\infty, \infty) \not\Longrightarrow f''(x) > 0, \ \forall x \in (-\infty, \infty)$$

(because at $x = 0$, $f''(x) = 0$).

2.4.2 Convex or Concave Functions of Several Variables

Definitions

A function of several variables $f(x)$, defined in a convex set S of $\Re^n$, is a *convex function* if, for each pair of distinct points in S, say

$$\bar{x} = (\bar{x}_1, \ldots, \bar{x}_n)^T, \quad \text{and} \quad \hat{x} = (\hat{x}_1, \ldots, \hat{x}_n)^T,$$

there holds

$$f(\lambda\bar{x} + (1-\lambda)\hat{x}) \leq \lambda f(\bar{x}) + (1-\lambda) f(\hat{x})$$

for all values of λ such that $0 < \lambda < 1$. It is a *strictly convex function* if $\leq$ can be replaced by $<$. It is a *concave function* (or a *strictly concave function*) if this statement holds when $\leq$ is replaced by $\geq$ (or by $>$).

In the above definition, we consider two points, $\bar{x}$ and $\hat{x}$. In fact for a convex or concave function, we can also consider p points $x^1, x^2, \ldots, x^p$, here p is a positive integer. That is, if $f(x)$ is a convex function over a convex set S, then for any p points $x^1, x^2, \ldots, x^p$ in S,

$$f(\sum_{i=1}^{p} \lambda_i x^i) \leq \sum_{i=1}^{p} \lambda_i f(x^i)$$

for any $\lambda_1, \ldots, \lambda_p$ satisfying

$$\lambda_i \geq 0, \ i = 1, \ldots, p; \text{ and } \sum_{i=1}^{p} \lambda_i = 1.$$

This conclusion can be proved by induction. For example, consider the case of $p = 3$. Then we can assume that $\lambda_1, \lambda_2, \lambda_3 > 0$ (otherwise it reduces to the case of $p < 3$). As the conclusion is true when $p = 2$, we know that

$$\begin{aligned}
& f(\lambda_1 x^1 + \lambda_2 x^2 + \lambda_3 x^3) \\
= \; & f(\lambda_1 x^1 + [\lambda_2 + \lambda_3][\frac{\lambda_2}{\lambda_2 + \lambda_3} x^2 + \frac{\lambda_3}{\lambda_2 + \lambda_3} x^3]) \\
\leq \; & \lambda_1 f(x^1) + (\lambda_2 + \lambda_3) f(\frac{\lambda_2}{\lambda_2 + \lambda_3} x^2 + \frac{\lambda_3}{\lambda_2 + \lambda_3} x^3) \\
\leq \; & \lambda_1 f(x^1) + (\lambda_2 + \lambda_3)[\frac{\lambda_2}{\lambda_2 + \lambda_3} f(x^2) + \frac{\lambda_3}{\lambda_2 + \lambda_3} f(x^3)] \\
= \; & \lambda_1 f(x^1) + \lambda_2 f(x^2) + \lambda_3 f(x^3).
\end{aligned}$$

Hence the conclusion is true if $p = 3$.

Examples of Convex Functions

- **Example 1** The linear function

$$f(x) = a^T x + b, \quad a, x \in \Re^n, \ b \in \Re.$$

In fact

$$\begin{aligned} f(\lambda x + (1-\lambda)y) &= a^T(\lambda x + (1-\lambda)y) + b \\ &= \lambda(a^T x + b) + (1-\lambda)(a^T y + b) \\ &= \lambda f(x) + (1-\lambda) f(y). \end{aligned}$$

So, $f(x)$ is convex, but not strictly convex.

Question: is this $f(x)$ also concave?

- **Example 2** The quadratic function

$$f(x) = (a^T x)^2, \quad a, x \in \Re^n.$$

In fact as $0 < \lambda < 1$,

$$\begin{aligned} & f(\lambda x + (1-\lambda)y) - [\lambda f(x) + (1-\lambda) f(y)] \\ = \; & [a^T(\lambda x + (1-\lambda)y)]^2 - [\lambda (a^T x)^2 + (1-\lambda)(a^T y)^2] \\ = \; & -\lambda(1-\lambda)(a^T x - a^T y)^2 \\ \leq \; & 0. \end{aligned}$$

Hence f is a convex function.

Convexity Test for a Function of Multi-Variable by Its First Order Derivatives

Consider a multi-variable function $f(x)$ that is defined in a convex set $S \subseteq \Re^n$ and possesses continuous gradient at each point x in S. Then $f(x)$ is

- *Convex* if and only if for each $\bar{x} \in S$,

$$f(y) \geq f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}), \quad \text{for any } y \in S; \qquad (1)$$

- *Strictly convex* if and only if for each $\bar{x} \in S$,

$$f(y) > f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}), \quad \text{for any } y \in S \text{ and } y \neq \bar{x}; \quad (2)$$

- *Concave* if and only if for each $\bar{x} \in S$,

$$f(y) \leq f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}), \quad \text{for any } y \in S;$$

- *Strictly concave* if and only if for each $\bar{x} \in S$,

$$f(y) < f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}), \quad \text{for any } y \in S \text{ and } y \neq \bar{x};$$

Here we explain the first conclusion in detail (other conclusions can be proved similarly).

First, suppose $f(x)$ is convex. Then for any $y \in S$ and any $\lambda \in (0, 1]$,

$$f(\bar{x} + \lambda(y - \bar{x})) = f(\lambda y + (1 - \lambda)\bar{x}) \leq \lambda f(y) + (1 - \lambda) f(\bar{x}),$$

hence

$$\frac{f(\bar{x} + \lambda(y - \bar{x})) - f(\bar{x})}{\lambda} \leq f(y) - f(\bar{x}).$$

Using Taylor's formula on the left hand side, we have

$$\nabla f(\bar{x} + \xi(y - \bar{x}))^T (y - \bar{x}) \leq f(y) - f(\bar{x}),$$

where ξ is a number between 0 and λ: $0 \leq \xi \leq \lambda$. Let $\lambda \to 0^+$ (hence $\xi \to 0^+$) and take limit, we obtain

$$\nabla f(\bar{x})^T (y - \bar{x}) \leq f(y) - f(\bar{x}),$$

i.e.,the inequality (1) is true.

Conversely, suppose (1) holds for every pair of points $(\bar{x}, y)$ in S. Now for any $u, v \in S$ and any $\lambda \in (0,1)$, let $w = \lambda u + (1-\lambda)v$. For the two pairs (w, u) and (w, v), by the condition (1),

$$f(u) - f(w) \geq \nabla f(w)^T (u - w),$$
$$f(v) - f(w) \geq \nabla f(w)^T (v - w).$$

Multiplying the first inequality by λ, multiplying the second one by $1-\lambda$, and then adding them, we obtain:

$$\lambda f(u) + (1-\lambda) f(v) - f(w) \geq \nabla f(w)^T [\lambda(u-w) + (1-\lambda)(v-w)] = 0,$$

which means that

$$f(w) = f(\lambda u + (1-\lambda)v) \leq \lambda f(u) + (1-\lambda) f(v).$$

Hence by definition f is a convex function.

The geometric meaning of these conclusions is similar to the single variable case. That is, in the two-dimensional case, $f(x)$ is a convex function if and only if every tangent plane to $f(x)$ lies on or below the surface of $f(x)$; and $f(x)$ is a strictly convex function if and only if every tangent plane to $f(x)$ lies below the surface of $f(x)$ and contact the surface only at the point of tangency.

Example 3 Let

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

Then

$$\nabla f(x) = (2x_1, 4x_2)^T.$$

Let $\bar{x} = (\bar{x}_1, \bar{x}_2)$, then for any $x = (x_1, x_2) \neq (\bar{x}_1, \bar{x}_2)$,

$$\begin{aligned}
& f(x) - f(\bar{x}) - \nabla f(\bar{x})^T (x - \bar{x}) \\
= \; & x_1^2 + 2x_2^2 - \bar{x}_1^2 - 2\bar{x}_2^2 - (2\bar{x}_1, 4\bar{x}_2)^T (x_1 - \bar{x}_1, x_2 - \bar{x}_2) \\
= \; & x_1^2 + 2x_2^2 - \bar{x}_1^2 - 2\bar{x}_2^2 - 2\bar{x}_1(x_1 - \bar{x}_1) - 4\bar{x}_2(x_2 - \bar{x}_2) \\
= \; & [x_1^2 + \bar{x}_1^2 - 2x_1\bar{x}_1] + 2[x_2^2 + \bar{x}_2^2 - 2x_2\bar{x}_2] \\
= \; & (x_1 - \bar{x}_1)^2 + 2(x_2 - \bar{x}_2)^2 > 0
\end{aligned}$$

So, the inequality (2) holds, and hence $f(x)$ is a strictly convex function over $\mathcal{R}^2$.

Convexity Test for a Function of Multi-Variable by Its Second Order Derivatives

Consider a multi-variable function $f(x)$ that is defined in an convex set $S \subseteq \Re^n$ and possesses all continuous second order partial derivatives at each point x in S. Then $f(x)$ is

- *Convex* if and only if $\nabla^2 f(x)$ is positive semi-definite at all points of S;
- *Strictly convex* if $\nabla^2 f(x)$ is positive definite at all points of S.
- *Concave* if and only if $\nabla^2 f(x)$ is negative semi-definite at all points of S;
- *Strictly concave* if $\nabla^2 f(x)$ is negative definite at all points of S

In fact for Example 3, $\nabla^2 f(x) = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$, which is positive definite everywhere. Hence this function is strictly convex. What are the Hessian matrices of the functions in Examples 1 and 2?

Definition

An $n \times n$ real symmetric matrix A is called positive semi-definite if

$$v^T A v \geq 0, \quad \text{for all non-zero vectors } v \in \Re^n.$$

A is *positive definite* if $\geq$ can be replaced by $>$.

We now can see that for a function f having continuous second order partial derivatives, **f is convex in a convex set S if and only if $\nabla^2 f(x)$ is positive semi-definite on S.**

First suppose $\nabla^2 f$ is positive semi-definite on S. Then at each $\bar{x} \in S$ and for any $y \in S$,

$$\begin{aligned} f(y) &= f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}) + \frac{1}{2}(y - \bar{x})^T \nabla^2 f(\xi)(y - \bar{x}) \\ &\geq f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}). \end{aligned}$$

Hence f is convex over the set S. In the above, ξ is a point between $\bar{x}$ and y. Hence $\xi \in S$, and $\nabla^2 f(\xi)$ is positive semi-definite.

Conversely, assume f is convex over S. We will see that $\nabla^2 f(x)$ must be positive semi-definite on S.

In fact if it is not true, then $\nabla^2 f$ is not positive semi-definite at some point $\bar{x}$ in S, which means that there exists a vector $z \neq 0$ such that

$$z^T \nabla^2 f(\bar{x}) z < 0.$$

As $\nabla^2 f$ is continuous, it means that for all ξ near $\bar{x}$, $z^T \nabla^2 f(\xi) z < 0$. Now consider $f(\bar{x} + \lambda z)$ for very small positive λ,

$$\begin{aligned} f(\bar{x} + \lambda z) &= f(\bar{x}) + \lambda \nabla f(\bar{x})^T z + \frac{1}{2}\lambda^2 z^T \nabla^2 f(\xi) z \\ &< f(\bar{x}) + \lambda \nabla f(\bar{x})^T z, \end{aligned}$$

where ξ is between $\bar{x}$ and $\bar{x} + \lambda z$, and hence near $\bar{x}$. The above inequality is against the convexity of f. Therefore, $\nabla^2 f(x)$ must be p.s.d. over S.

Eigenvalue Test

A real symmetric matrix A is positive semi-definite if and only if all its eigenvalues are non-negative, i.e., all the solutions, λ, to the equation $|A - \lambda I| = 0$ are real and non-negative.

Similarly, A is positive definite if and only if all its eigenvalues are positive.

Principle Minor Test

A real symmetric matrix A is positive semi-definite if and only if all its principal minors are non-negative, i.e.

$$a_{11} \geq 0, \quad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \geq 0, \quad \cdots, \quad \begin{vmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{vmatrix} \geq 0.$$

Similarly, A is positive definite if and only if all its principal minors are positive.

2.5.2 Negative Definite Matrices

Definition

An $n \times n$ real symmetric matrix A is called *negative semi-definite* if

$$v^T A v \leq 0, \quad \text{for all non-zero vectors } v \in \Re^n.$$

A is *negative definite* if $\leq$ can be replaced by $<$.

Eigenvalue Test

A real symmetric matrix A is negative semi-definite if and only if all its eigenvalues are non-positive.

Similarly, A is negative definite if and only if all its eigenvalues are negative.

Principle Minor Test

A real symmetric matrix A is negative definite if and only if its principal minors alternate in sign starting with a minus sign:

$$a_{1,1} < 0, \quad \begin{vmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{vmatrix} > 0, \quad \cdots$$

$$\begin{vmatrix} a_{1,1} & \cdots & a_{1,2k-1} \\ \vdots & & \vdots \\ a_{2k-1,1} & \cdots & a_{2k-1,2k-1} \end{vmatrix} < 0, \quad \begin{vmatrix} a_{1,1} & \cdots & a_{1,2k} \\ \vdots & & \vdots \\ a_{2k,1} & \cdots & a_{2k,2k} \end{vmatrix} > 0, \quad \cdots.$$

2.5.3 Indefinite Matrices

An $n \times n$ real symmetric matrix A is said to be *indefinite* if the matrix A is neither positive semi-definite nor negative semi-definite.

2.6 Some Properties of Convex Function

- If $f(x)$ and $g(x)$ are two convex functions defined on a convex set $S \subseteq \Re^n$, then $f(x) + g(x)$ is also a convex function in S; Moreover, if at least one of $f(x)$ and $g(x)$ is strictly convex, then $f(x) + g(x)$ is a strictly convex function;
- If $f(x)$ is a convex function over a convex set S, then $\alpha f(x)$ is a convex function if $\alpha > 0$ and a concave function if $\alpha < 0$;
- If $f(x)$ is a convex function defined on a convex set $S \subseteq \Re^n$ and $g(u)$ is a single variable function which is convex and increasing on $\Re$, then $h(x) \equiv g(f(x))$ is convex on S.

We now explain the last property. For any $y, z \in S$, as f is convex, for any $\lambda \in [0, 1]$,

$$f(\lambda y + (1 - \lambda)z) \leq \lambda f(y) + (1 - \lambda)f(z).$$

Since g is increasing,

$$\begin{aligned} g(f(\lambda y + (1 - \lambda)z)) &\leq g(\lambda f(y) + (1 - \lambda)f(z)) \\ &\leq \lambda g(f(y)) + (1 - \lambda)g(f(z)), \end{aligned}$$

i.e.,

$$h(\lambda y + (1 - \lambda)z) \leq \lambda h(y) + (1 - \lambda)h(z).$$

So, $h(x)$ is a convex function.

We may consider the condition to further strengthen the last property to a strictly convex function, that is, under what conditions, $h(x) \equiv g(f(x))$ is strictly convex?

Example Show that

$$h(x_1, x_2, x_3) = e^{x_1^2 + x_2^2 + x_3^2}$$

is a convex function in $\Re^3$.

Let $f(x) = x_1^2 + x_2^2 + x_3^2$ and $g(u) = e^u$ for $x \in \Re^3$ and $u \in \Re$. As

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

which is positive definite, $f(x)$ is strictly convex. On the other hand, as

$$g''(u) = e^u > 0 \text{ for every } u,$$

$g(u)$ is also strictly convex. Also, g is an increasing function. Therefore, $h(x)$ is a convex function. In fact it is strictly convex.

Example Show that

$$f(x_1, x_2) = x_1^2 - 4x_1x_2 + 5x_2^2 - \ln x_1x_2$$

is a strictly convex function on $S = \{x \in \Re^2 \mid x_1 > 0, x_2 > 0\}$. Let

$$g(x_1, x_2) = x_1^2 - 4x_1x_2 + 5x_2^2$$

and

$$h(x_1, x_2) = -\ln x_1x_2 = -\ln x_1 - \ln x_2.$$

Hence

$$f(x_1, x_2) = g(x_1, x_2) + h(x_1, x_2).$$

We now consider convexity of g and h respectively.

As

$$\nabla^2 g(x) = \begin{bmatrix} 2 & -4 \\ -4 & 10 \end{bmatrix}$$

is positive definite, $g(x)$ is strictly convex. Let

$$\phi(t) = -\ln t, \quad t > 0.$$

As $\phi''(t) = \frac{1}{t^2} > 0$, $\phi(t)$ is a strictly convex function for all $t > 0$. So, $h(x) = \phi(x_1) + \phi(x_2)$ is strictly convex on S.

Therefore, f is a strictly convex function.

General Optimization Problems

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in S \subset \Re^n. \end{array}$$

Weierstrass Theorem

A continuous function $f(x)$ defined on a **compact set** S has a minimum point in S.

Local Minimizers

A point $x^* \in S$ is said to be a local minimizer (or local minimum point) of f over S if there is an $\varepsilon > 0$ such that $f(x) \geq f(x^*)$ for all $x \in S \cap N_\varepsilon(x^*)$, where $N_\varepsilon(x^*)$ is the ε-neighborhood of x^*.

If $f(x) > f(x^*)$ for all $x \in S \cap N_\varepsilon(x^*)$ and $x \neq x^*$, then x^* is said to be a strict local minimizer of f over S.

Global Minimizers

A point $x^* \in S$ is said to be a global minimizer (or global minimum point) of f over S if $f(x) \geq f(x^*)$ for all $x \in S$.

If $f(x) > f(x^*)$ for all $x \in S$ and $x \neq x^*$, then x^* is said to be a strict global minimizer of f over S.

Remarks

- It is preferred to find a global minimizer when formulating an optimization problem.
- In most situations, however, optimization theory and methodologies only enable us to locate local minimum points or stationary points.

Recall that a point x^* is called a **stationary point** of f if $\nabla f(x^*) = 0$.

Minimum Point for Convex Functions

For a convex function $f(x)$ defined on a convex set S, its minimum points have the following nice properties.

- Every stationary point is a global minimizer.
  Let $\bar{x}$ be a stationary point of f, i.e., $\nabla f(\bar{x}) = 0$. Then for every $y \in S$,

$$f(y) \geq f(\bar{x}) + \nabla f(\bar{x})^T (y - \bar{x}) = f(\bar{x}).$$

  Hence $\bar{x}$ is the global minimizer of f over the convex set S.
- Every local minimizer of f is also a global minimizer of f over the convex set S.

Suppose x^* is a local minimizer of f. Then there is a small $r > 0$ such that for every $x \in S$ satisfying $\|x - x^*\| < r$, $f(x^*) \leq f(x)$. We now consider an arbitrary $y \in S$. As S is a convex set, the line segment $\{x^* + \lambda(y - x^*) \mid 0 \leq \lambda \leq 1\}$ joining x^* and y is in S. We choose a sufficiently small positive $\bar{\lambda} < 1$ such that $\|\bar{\lambda}(y - x^*)\| < r$. Thus,

$$\begin{aligned} f(x^*) &\leq f(x^* + \bar{\lambda}(y - x^*)) \\ &= f(\bar{\lambda}y + (1 - \bar{\lambda})x^*) \\ &\leq \bar{\lambda}f(y) + (1 - \bar{\lambda})f(x^*). \end{aligned}$$

It follows that

$$\bar{\lambda}f(x^*) \leq \bar{\lambda}f(y),$$

i.e., $f(x^*) \leq f(y)$. As y can be any point in S, x^* is a global minimizer of f over the convex set S.

- If a strictly convex function $f(x)$ over a convex set S has a global minimizer, then it must be the unique global minimizer (i.e., there is no other global minimizer).
  Suppose x^* is a global minimizer on S, then for any $y \in S$, $y \neq x^*$, as f is strictly convex, we can obtain from the above reasoning that

$$f(x^*) \leq f(\bar{\lambda}y + (1-\bar{\lambda})x^*) < \bar{\lambda}f(y) + (1-\bar{\lambda})f(x^*)$$

  for any $\bar{\lambda}$ such that $0 < \bar{\lambda} < 1$. Therefore,

$$f(x^*) < f(y), \text{ for every } y \in S, y \neq x^*,$$

  which means that x^* is the unique global minimizer over S.

For most optimization problems, we fail to obtain an explicit analytic solution, and must use numerical method to get it by a sequence of computation. Such methods are often called iterative methods.

A General Scheme of an Iterative Solution Procedure

Step 1. Start from a feasible solution $x \in S$.

Step 2. Check if the stopping criteria (such as the optimality conditions) are met.
If the answer is YES, stop.
If the answer is NO, continue.

Step 3. Move to a better feasible solution and return to Step 2.

Feasible Directions

Along any given direction, the objective function can be regarded as a function of a single variable.

Given $x \in S$, a vector $d \in \Re^n$ is a *feasible direction* at x if there is an $\bar{\alpha} > 0$ such that $x + \alpha d \in S$ for all α such that $0 \leq \alpha \leq \bar{\alpha}$.

2.8 Appendix 1 - Gradients and Hessians for Linear and Quadratic Functions

2.8.1 Linear Function

Let

$$f(x) = a^T x = (a_1, a_2, \ldots, a_n) \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix}.$$

Then,

$$\frac{\partial f}{\partial x_1} = a_1, \quad \frac{\partial f}{\partial x_2} = a_2, \quad \ldots, \quad \frac{\partial f}{\partial x_n} = a_n.$$

So,

$$\nabla f(x) = a, \text{ and } \nabla^2 f(x) = 0.$$

2.8.2 Quadratic Function

Let

$$h(x) = \frac{1}{2} x^T A x,$$

where A is an $n \times n$ symmetric matrix. Then

$$h(x) = \frac{1}{2} \sum_{i,j=1}^{n} a_{ij} x_i x_j,$$

and the following terms of $h(x)$ contain x_1:

$$\frac{1}{2}(a_{11}x_1^2 + \sum_{j \neq 1} a_{1j} x_1 x_j + \sum_{i \neq 1} a_{i1} x_i x_1).$$

Hence,

$$\begin{aligned}
\frac{\partial h}{\partial x_1} &= \frac{1}{2}(2a_{11}x_1 + \sum_{j\neq 1} a_{1j}x_j + \sum_{i\neq 1} a_{i1}x_i) \\
&= a_{11}x_1 + \sum_{j\neq 1} a_{1j}x_j \text{ (note that } a_{i1} = a_{1i}) \\
&= \sum_{j=1}^{n} a_{1j}x_j.
\end{aligned}$$

Similarly,

$$\frac{\partial h}{\partial x_2} = \sum_{j=1}^{n} a_{2j}x_j, \quad \cdots \quad , \frac{\partial h}{\partial x_n} = \sum_{j=1}^{n} a_{nj}x_j.$$

Therefore,

$$\nabla h(x) = Ax.$$

For $i, j = 1, \ldots, n$, from

$$\frac{\partial h}{\partial x_i} = \sum_{k=1}^{n} a_{ik} x_k$$

we know that

$$\frac{\partial^2 h}{\partial x_i \partial x_j} = a_{ij}.$$

Hence,

$$\nabla^2 h(x) = A.$$

Let us return to **Example 2**:

$$f(x) = (a^T x)^2, \quad a, x \in \Re^n.$$

What are $\nabla f(x)$ and $\nabla^2 f(x)$?

In fact

$$f(x) = (a^T x)^2 = a^T x a^T x = x^T a a^T x = \frac{1}{2} x^T A x,$$

where $A = 2aa^T$. So,

$$\nabla f(x) = Ax = 2aa^T x = 2(a^T x)a,$$

and

$$\nabla^2 f(x) = A = 2aa^T.$$

2.9 Appendix 2 - Gradients and Hessians for Product and Composite Functions

2.9.1 Product Function

Let

$$f(x) = g(x)h(x),$$

where $x = (x_1, x_2, \ldots, x_n)$, and suppose that $g(x)$ and $h(x)$ are both continuously differentiable. We need to calculate $\nabla f(x)$ and $\nabla^2 f(x)$. We know that

$$\frac{\partial f}{\partial x_1} = \frac{\partial g}{\partial x_1} h(x) + \frac{\partial h}{\partial x_1} g(x)$$

$$\ldots\ldots\ldots$$

$$\frac{\partial f}{\partial x_n} = \frac{\partial g}{\partial x_n} h(x) + \frac{\partial h}{\partial x_n} g(x).$$

Hence

$$\nabla f(x) = h(x)\nabla g(x) + g(x)\nabla h(x).$$

2.9.1 Product Function

For $\nabla^2 f$, it can be verified that

$$\nabla^2 f(x) = h(x)\nabla^2 g(x) + g(x)\nabla^2 h(x) + \nabla g(x)\nabla h(x)^T + \nabla h(x)\nabla g(x)^T.$$

Note that

$$\nabla g(x)\nabla h(x)^T = \begin{bmatrix} \frac{\partial g}{\partial x_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial g}{\partial x_n} \end{bmatrix} \begin{bmatrix} \frac{\partial h}{\partial x_1} & \cdot & \cdot & \cdot & \frac{\partial h}{\partial x_n} \end{bmatrix}$$

is an $n \times n$ matrix, and so is $\nabla h(x)\nabla g(x)^T$.

2.9.1 Product Function

Example Consider the function

$$f(x) = (a^T x)(b^T x),$$

where $a, b, x \in R^n$.

We can let $g(x) = a^T x$, $h(x) = b^T x$, and use the above general formula:

$$\begin{aligned} \nabla f(x) &= h(x)\nabla g(x) + g(x)\nabla h(x) \\ &= (b^T x)a + (a^T x)b. \end{aligned}$$

Since $\nabla^2 g(x) = \nabla^2 h(x) = 0$ (zero matrix),

$$\begin{aligned} \nabla^2 f(x) &= \nabla g(x)\nabla h(x)^T + \nabla h(x)\nabla g(x)^T \\ &= ab^T + ba^T. \end{aligned}$$

2.9.2 Composite Function - Chain Rule

Suppose

$$y = g(x) = g(x_1, x_2, \ldots, x_n)$$

and

$$x_i = x_i(t_1, t_2, \ldots, t_m), \quad i = 1, \ldots, n$$

where g is a continuously differential function of $x \in R^n$, and each x_i is a continuously differentiable function of $t = (t_1, \ldots, t_m)$. By the chain rule, we know that

$$\begin{aligned}
\frac{\partial y}{\partial t_1} &= \frac{\partial g}{\partial x_1}\frac{\partial x_1}{\partial t_1} + \cdots + \frac{\partial g}{\partial x_n}\frac{\partial x_n}{\partial t_1} \\
&\vdots \\
\frac{\partial y}{\partial t_m} &= \frac{\partial g}{\partial x_1}\frac{\partial x_1}{\partial t_m} + \cdots + \frac{\partial g}{\partial x_n}\frac{\partial x_n}{\partial t_m}.
\end{aligned}$$

2.9.2 Composite Function - Chain Rule

So,

$$\begin{bmatrix} \frac{\partial y}{\partial t_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial y}{\partial t_m} \end{bmatrix} = \begin{bmatrix} \frac{\partial x_1}{\partial t_1} & \cdots & \frac{\partial x_n}{\partial t_1} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{\partial x_1}{\partial t_m} & \cdots & \frac{\partial x_n}{\partial t_m} \end{bmatrix} \begin{bmatrix} \frac{\partial g}{\partial x_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial g}{\partial x_n} \end{bmatrix},$$

i.e.,

$$\nabla y(t) = \nabla x(t) \nabla g(x),$$

where the matrix

$$\nabla x(t) = [\nabla x_1(t), \cdots, \nabla x_n(t)].$$

2.9.2 Composite Function - Chain Rule

Example Let

$$\begin{aligned} y &= x_1^2 - x_1 x_2, \\ x_1 &= t_1 + 2t_2, \\ x_2 &= t_1^2 + t_2. \end{aligned}$$

Then, by the chain rule,

$$\begin{aligned} \nabla y(t) &= \nabla x(t) \nabla y(x) \\ &= \begin{bmatrix} 1 & 2t_1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2x_1 - x_2 \\ -x_1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 2t_1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 2(t_1 + 2t_2) - (t_1^2 + t_2) \\ -(t_1 + 2t_2) \end{bmatrix}. \end{aligned}$$

OR4030 OPTIMIZATION Chapter 3
One-Dimensional Unconstrained Optimization — Line Search Methods
3.1 Introduction

3,1.1 Problem Description

Problem:

$$\begin{aligned} \min \quad & f(x), \\ \text{s.t.} \quad & x \in S = [x_s, x_t] \subset \Re, \end{aligned}$$

where

1. $f : \Re \mapsto \Re$;
2. $[x_s, x_t]$ is a given interval of uncertainty;

3.1.2 A Trivial Example

Problem:

$$\begin{aligned} \min \quad & f(x) = 12x^6 + 3x^4 - 12x + 7, \\ \text{s.t.} \quad & 0 \le x \le 1. \end{aligned}$$

We know that

$$f'(x) = 72x^5 + 12x^3 - 12, \quad \text{and} \quad f''(x) = 360x^4 + 36x^2.$$

If we can solve the equation

$$f'(x) = 72x^5 + 12x^3 - 12 = 0, \tag{1}$$

and if a root x^* of equation (1) is in the interval of $[0, 1]$, then it is a stationary point. Furthermore, if $x^* \neq 0$, then $f''(x^*) > 0$, and it must be a minimizer. In fact the optimal solution is

$$x^* = 0.65435605252093 \quad \text{with} \quad f(x^*) = 0.63977916586877.$$

However, as (1) is an order 5 equation, it is not easy to obtain the solution.

3.1.3 Remarks

- When analytical solution is not achievable, numerical methods are essential to the success of finding an optimum solution.
- The process of determining the minimum point on a given line (i.e. with only one variable) is called *line search* or *one dimensional search*. The line search techniques are procedures for solving one-dimensional minimization problems.

3.2 One-Dimensional Search Methods that Use Function Values Only

3.2.1 Unimodal Functions

In this section we consider minimization for unimodal functions.

Definition Let x^* be the minimum point of f over an interval $[x_s, x_t]$. f is called a *unimodal* function on $[x_s, x_t]$ if for any x_1, x_2, x_3 and x_4 in $[x_s, x_t]$ with the arrangement:

$$x_s < x_1 < x_2 < x^* < x_3 < x_4 < x_t$$

then their function values must have the the following relationship:

$$f(x_s) > f(x_1) > f(x_2) > f(x^*) < f(x_3) < f(x_4) < f(x_t).$$

Or simply speaking, to the right of x^*, f is an increasing one, whereas to the left of x^*, f is a decreasing one.

Consider a unimodal function defined on an interval $[x_s, x_t]$. Assume that two function evaluations are carried out at x_1 and x_2 with $x_s < x_1 < x_2 < x_t$.

- If $f(x_1) < f(x_2)$, then $[x_2, x_t]$ can be discarded, because for each $x \in [x_2, x_t]$, its function value $f(x) \geq f(x_2)$. Hence $[x_2, x_t]$ cannot contain the minimum point. By the same reasoning,
- If $f(x_1) > f(x_2)$, then $[x_s, x_1]$ can be discarded.
- If $f(x_1) = f(x_2)$, then both $[x_s, x_1]$ and $[x_2, x_t]$ can be discarded.

The above conclusions mean that by comparing two function values in $[x_s, x_t]$, we can reduce the search interval by discarding a part of the interval. Based on this property, we may design a class of methods for locating the minimum point for a unimodal function.

Output of this class of line search methods

The output is the final *interval* of uncertainty, not the exact minimum point. But we often approximately take the middle point of the final interval as the minimum point.

To design algorithms for minimizing unimodal functions, we should consider to calculate function values *at what points*, so that we can let the final interval of uncertainty meet the required accuracy while the number of function evaluations can be as small as possible.

3.2.2 Exhaustive Search Method

- Evaluating the objective function at a predetermined number (N) of equally spaced points in the interval (x_s, x_t),

$$x_s < x_1 < \cdots < x_N < x_t.$$

- If the minimum value among the N function values is x_K, then the final interval of uncertainty is $[x_{K-1}, x_{K+1}]$ with length of

$$L_N = x_{K+1} - x_{K-1} = \left(\frac{2}{N+1}\right) L_0,$$

where $L_0 = x_t - x_s$.

- If we ask the length of the final interval of uncertainty to be bounded by $\varepsilon > 0$, then we should let

$$(\frac{2}{N+1})L_0 < \varepsilon \Longrightarrow N+1 > \frac{2L_0}{\varepsilon} \Longrightarrow N > \frac{2L_0}{\varepsilon} - 1.$$

Advantage

- The algorithm is simple.

Disadvantages

- N is predetermined. That is, after computing the function values for N times, the procedure cannot continue by one extra function value calculation.
- This is a simultaneous search method (i.e., no decision can be made by only one of these N function evaluations) and is relatively inefficient.

Algorithm (Exhaustive Search Method)

Step 1. Input and Initialization:

(a) Input $[x_s, x_t]$ = the initial interval of uncertainty.

(b) Input N (≥ 2) = the total number of function evaluations to be conducted.

(c) Let $L_0 := x_t - x_s$.

Step 2. Main procedure:

(a) Generate N equally spaced points in the interval (x_s, x_t), i.e.,

$$x_k := x_s + \left(\frac{k}{N+1}\right) L_0, \text{ for } k = 1, 2, \ldots, N.$$

(b) Calculate all $f(x_k)$ and find J such that $f(x_J) = \min_k f(x_k)$.

Step 3. Output $[x_{J-1}, x_{J+1}]$ = the final interval of uncertainty.

Golden Section Search Method

Idea

- We know that the positive root to the equation

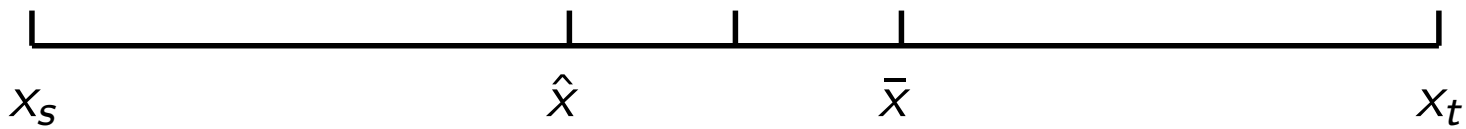
$$\tau^2 + \tau - 1 = 0$$

is $\tau = \frac{-1+\sqrt{5}}{2} \approx 0.618$. This τ is called the golden ratio.

- Let the interval of uncertainty be $[x_s, x_t]$, in which we choose the first test point

$$\bar{x} = x_s + \tau L_0, \tag{2}$$

where L_0 is the length of the interval: $L_0 = x_t - x_s$. We next choose the second test point $\hat{x}$ as the point symmetric to $\bar{x}$ with respect to the middle point of the interval $[x_s, x_t]$. How to express the location of $\hat{x}$ analytically?



Due to symmetry,

$$|x_s\hat{x}| = |\bar{x}x_t| = (1-\tau)L_0.$$

So,

$$\hat{x} = x_s + |x_s\hat{x}| = x_s + (1-\tau)L_0. \tag{3}$$

- We calculate $f(\bar{x})$ and $f(\hat{x})$, and compare their values.
  Case A If $f(\bar{x}) \geq f(\hat{x})$, we discard $[\bar{x}, x_t]$ and let the remaining interval of uncertainty be

$$[x_s^1, x_t^1] = [x_s, \bar{x}]; \tag{4}$$

  Case B otherwise, $f(\bar{x}) < f(\hat{x})$, we discard $[x_s, \hat{x}]$ and let

$$[x_s^1, x_t^1] = [\hat{x}, x_t]. \tag{5}$$

- The first iteration is finished and we obtain the second interval of uncertainty $[x_s^1, x_t^1]$. Let its length be L_1. L_1=? Note that in either of the above two cases, the removed length is $(1-\tau)L_0$. Hence

$$L_1 = L_0 - (1-\tau)L_0 = \tau L_0. \tag{6}$$

That is, after this iteration, the length of the shortened interval is reduced to the τ times of the original length.

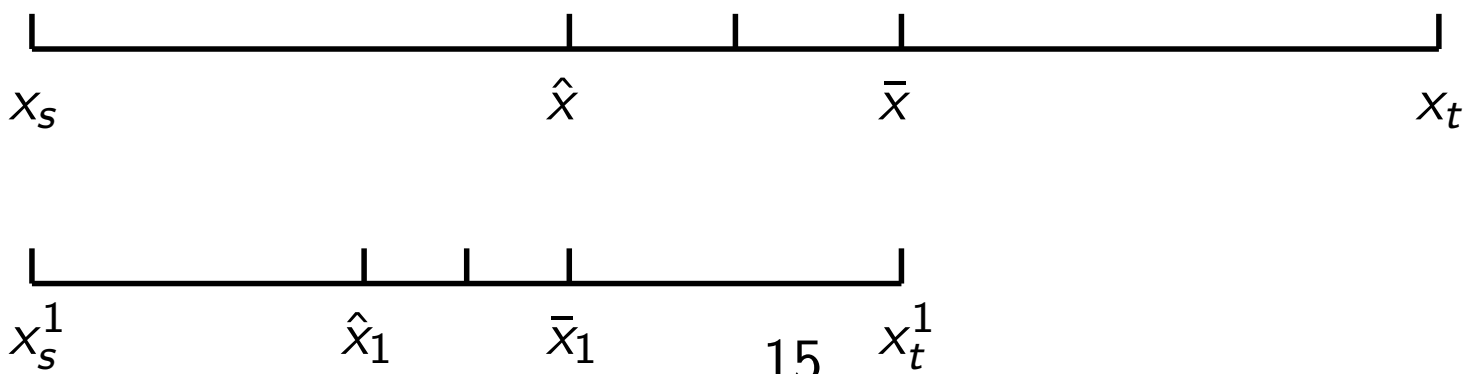
- Consider the second iteration. Suppose it is the Case A:

$$[x_s^1, x_t^1] = [x_s, \bar{x}],$$

and we need to arrange two test points, $\bar{x}_1$ and $\hat{x}_1$, in $[x_s^1, x_t^1]$ by the same method, that is, we let

$$\bar{x}_1 = x_s^1 + \tau L_1$$

and $\hat{x}_1$ be the point symmetric to $\bar{x}_1$ with respect to the mid-point of the interval $[x_s^1, x_t^1]$.



- By formula (2), we see that

$$\begin{aligned} \bar{x}_1 &= x_s^1 + \tau L_1 \\ &= x_s + \tau^2 L_0 \\ &= x_s + (1-\tau) L_0 \quad (\text{because } \tau^2 = 1 - \tau) \\ &= \hat{x}. \quad (\text{see formula (3) }) \end{aligned}$$

Hence, $\bar{x}_1$ and $\hat{x}$ coincide so that $f(\bar{x}_1) = f(\hat{x})$, which has already been calculated in the last iteration. On the other hand, by formula (3),

$$\hat{x}_1 = x_s^1 + (1-\tau) L_1,$$

which is a new point. We calculate $f(\hat{x}_1)$, then by comparing the values $f(\bar{x}_1)$ and $f(\hat{x}_1)$, we can reduce the interval of uncertainty to a new one $[x_s^2, x_t^2]$ (like the formulas (4) and (5)).

We have completed the second iteration. Like (6), the length L_2 of the interval $[x_s^2, x_t^2]$ is

$$L_2 = \tau L_1. \tag{7}$$

Note that the **important advantage** of the Golden Section method is that starting with the second iteration, between the two test points, one is used in the previous iteration so that its function value is readily available. Hence in order to compare the two values, we only need to calculate ONE, not two, function values. This idea helps the method enhance efficiency.

An equivalent way to express this advantage is that if we set point $\bar{x}$ by formula (2) as the initial test point before all iterations, then in each iteration (including the first one), the Golden Section method need to generate and evaluate only one test point.

- The above analysis shows that in the method the lengths of intervals will decrease with a constant rate:

$$L_k = \tau L_{k-1}, \quad k = 1, 2, \ldots$$

So, if we repeat the procedure for N times, then the length of final interval of uncertainty is

$$L_N = \tau L_{N-1} = \tau^2 L_{N-2} = \cdots = \tau^N L_0.$$

Using this formula, we may determine the number N of iterations in order to let the length of final interval be within a given accuracy ε:

$$L_N \leq \varepsilon \Longrightarrow \tau^N \leq \frac{\varepsilon}{L_0} \Longrightarrow N \geq \log_\tau(\frac{\varepsilon}{L_0}).$$

And we know that to conduct N iterations, we need to compute function values for $N + 1$ times.

Note that another advantage of the Golden Section method, comparing to the exhaustive method, is that even if we have finished N iterations, if we wish to make more iterations to raise accuracy further, we can repeat the same procedure for any number of times.

Below we state the algorithm. In the algorithm, Step 1 and Step 2 are initial steps in which the first test point, denoted by $\bar{x}^1$, is given by formula (2). Step 3 is the main step which shall repeat for N times, giving N iterations. In the k-th iteration, $\bar{x}^k$ is the test point which we already know in the last iteration, and x_{k+1} is the new test point which is symmetric to $\bar{x}^k$ in the current interval of uncertainty.

Algorithm (Golden Section Method)

Step 1. Input and Initialization:

(a) Input $[x_s^0, x_t^0]$ = the initial interval of uncertainty.

(b) Input N (≥ 2) = the total number of iterations to be conducted.

(c) Let $L_0 := x_t^0 - x_s^0$.

(d) Let $\tau := \dfrac{\sqrt{5}-1}{2} \approx 0.618$ = the golden ratio.

Step 2. Place the first point:

(a) Let $\bar{x}^1 := x_s^0 + \tau L_0$.

(b) Let $x_1 := \bar{x}^1$.

(c) calculate $f(x_1)$.

Step 3. For $k = 1, 2, 3, \ldots, N$, do the following:

(a) Place a point x_{k+1} symmetrically in the interval $[x_s^{k-1}, x_t^{k-1}]$ with respect to $\bar{x}^k$. Calculate $f(x_{k+1})$ and compare it with $f(\bar{x}^k)$.

(b) Use the elimination scheme for unimodal function to discard $1 - \tau$ of the interval of uncertainty obtained at iteration $k - 1$.

(c) Let $[x_s^k, x_t^k] :=$ the interval of uncertainty obtained at iteration k.

(d) Let $\bar{x}^{k+1} :=$ the test point remaining inside $[x_s^k, x_t^k]$.

Step 4. Output $[x_s^N, x_t^N]$ as the final interval of uncertainty.

Example We consider the following problem:

$$\begin{aligned} \min \quad & f(x) = x^2 + 2x \\ s.t. \quad & -3 \leq x \leq 5. \end{aligned}$$

We use the Golden Section method until the length of the uncertain interval is at most 0.2. In iteration k, let x_s^k and x_t^k be respectively the left and right endpoint of the search interval, and let $\hat{x}^k$ and $\bar{x}^k$ be respectively the left and right test point. The first two test points are chosen as

$$\hat{x}^1 = -3 + 0.382 \times 8 = 0.056, \qquad \bar{x}^1 = -3 + 0.618 \times 8 = 1.944.$$

The computations are summarized in the table below. After eight iterations involving nine function evaluations, the search interval becomes [-1.112, -0.936] which meets the accuracy requirement, so that the minimum point can be estimated to be the midpoint -1.024. Note that the true minimum point is -1.0.

Table: Summary of Computations for the Golden Section Method

Iteration k	x_s^k	x_t^k	$\hat{x}^k$	$\bar{x}^k$	$f(\hat{x}^k)$	$f(\bar{x}^k)$
1	-3.000	5.000	0.056	1.944	0.115*	7.667*
2	-3.000	1.944	-1.112	0.056	-0.987*	0.115
3	-3.000	0.056	-1.832	-1.112	-0.308*	-0.987
4	-1.832	0.056	-1.112	-0.664	-0.987	-0.887*
5	-1.832	-0.664	-1.384	-1.112	-0.853*	-0.987
6	-1.384	-0.664	-1.112	-0.936	-0.987	-0.996*
7	-1.112	-0.664	-0.936	-0.840	-0.996	-0.974*
8	-1.112	-0.840	-1.016	-0.936	-1.000*	-0.996
9	-1.112	-0.936				

* — this function value should be calculated in the iteration

3.2.4 Comparison of the Two Derivative Free Methods

Suppose the length of initial interval of uncertainty $L_0 = 1$ and after N iterations, the length of the final interval is L_N. If we take the mid-point of the final interval as an approximate minimum point, then the maximum error E should be no more than the half-length of the final interval: $E \leq \frac{L_N}{2}$. In the following table we give the number of function evaluations n required to reach different accuracies for the two methods. We see that the Golden Section method apparently outperforms the Exhaustive method.

Method	$E \leq 0.1$	$E \leq 0.01$	$E \leq 0.001$
Exhaustive Search	$n \geq 9$	$n \geq 99$	$n \geq 999$
Golden Section Search	$n \geq 5$	$n \geq 10$	$n \geq 14$

n — the number of function evaluations.

3.3 A One-Dimensional Search Method that Uses First Order Derivatives - Bisection Method

We want to find

$$\begin{array}{ll} \min & f(x) \\ s.t. & x \in [x_s, x_t] \end{array}$$

Idea

- We turn to find a stationary point: $f'(x^*) = 0$.
- Assume that $f \in C^1[x_s, x_t]$.
- Let $g(x) = f'(x)$. Then $g \in C^0[x_s, x_t]$.
- **Intermediate Value Theorem.** If $g \in C^0[x_s, x_t]$ and K is any number between $g(x_s)$ and $g(x_t)$, then there exists c in (x_s, x_t) for which $g(c) = K$.

- Suppose that $g(x_s)g(x_t) < 0$. Then, there exists x^* in (x_s, x_t) such that $g(x^*) = f'(x^*) = 0$.
- To begin, set $[x_s^0, x_t^0] := [x_s, x_t]$ and let x^1 be the mid-point of $[x_s^0, x_t^0]$.
- If $g(x^1) = 0$, then set $x^* := x^1$ and stop. If not, then $g(x^1)$ has a different sign from either $g(x_s^0)$ or $g(x_t^0)$.
- If $g(x^1)$ and $g(x_s^0)$ have different signs, then $x^* \in (x_s^0, x^1)$, and we set $[x_s^1, x_t^1] := [x_s^0, x^1]$. If $g(x^1)$ and $g(x_t^0)$ have different signs, then $x^* \in (x^1, x_t^0)$, and we set $[x_s^1, x_t^1] := [x^1, x_t^0]$.

- We then reapply the procedure to the interval $[x_s^1, x_t^1]$.
- In general, we have

$$L_k = \frac{L_{k-1}}{2} = \cdots = \frac{L_0}{2^k}.$$

The length of the final interval of uncertainty is

$$L_N = \frac{L_0}{2^N}.$$

- Totally, we need to calculate $f'(x)$ for $N+2$ times, because to start with, we need to know the values $f'(x_s)$ and $f'(x_t)$, and then in each iteration, we need to calculate derivative $f'(x)$ at the middle point of the search interval.

Algorithm

Step 1. Input and Initialization:

(a) Input $[x_s^0, x_t^0]$ = the initial interval of uncertainty.

(b) Calculate $f'(x_s^0)$ and $f'(x_t^0)$. If $f'(x_s^0)f'(x_t^0) > 0$, then ERROR.

(c) Input N = the total number of iterations required.

(d) Let $L_0 := x_t^0 - x_s^0$.

(In (b), ERROR means a correct search interval has not been located yet so that we cannot start using the method.)

Step 2. For $k = 1, 2, \ldots, N$, do the following:

(a) Let $x^k := \frac{1}{2}(x_s^{k-1} + x_t^{k-1})$ and calculate $f'(x^k)$.

(b) If $f'(x^k) = 0$, then output $x^* = x^k$ and STOP.

(c) If $f'(x^k)f'(x_s^{k-1}) < 0$, then let $[x_s^k, x_t^k] := [x_s^{k-1}, x^k]$, else let $[x_s^k, x_t^k] := [x^k, x_t^{k-1}]$.

(d) Let $L_k := \frac{1}{2}L_{k-1}$.

Step 3. Output $[x_s^N, x_t^N]$ as the final interval of uncertainty which has length L_N.

3.4.1 Newton's Method

Idea

- Assume that $f \in C^2[x_s, x_t]$ is a convex function (if f is not convex, the method can find only a stationary point, not a minimum point).
- Assume that an initial point $x^1 \in [x_s, x_t]$ is given and is sufficiently close to the minimum solution x^* with $f'(x^*) = 0$ and $f''(x^*) > 0$.
- Assume that $f(x^k)$, $f'(x^k)$ and $f''(x^k)$ are known at point x^k in iteration k $(k = 1, 2, \ldots)$.

- Construct a quadratic function which at x^k agrees with f up to second derivatives, i.e.,

$$f(x) \approx q(x) = f(x^k) + f'(x^k)(x - x^k) + \frac{1}{2} f''(x^k)(x - x^k)^2.$$

(It is easy to see that $q(x^k) = f(x^k), q'(x^k) = f'(x^k)$, and $q''(x^k) = f''(x^k)$).

- Let the next guess of the minimum point, x^{k+1}, be the minimum point of $q(x)$, i.e.,

$$q'(x) = 0 \Longrightarrow f'(x^k) + f''(x^k)(x - x^k) = 0 \Longrightarrow x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)}.$$

- The process repeats until $|f'(x^{k+1})| < \varepsilon$ or other termination criterion is satisfied.

In fact Newton's method can be used to solve equation $g(x) = 0$ with the formula:

$$x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)}.$$

In a minimization problem, we replace $g(x)$ by $f'(x)$.

Advantage

- Fast if x^1 is near x^*. (need fewer iterations than the bisection method does)

Disadvantage

- Need second-order derivatives.
- The sequence $\{x^k\}$ may not converge to the solution x^* if the initial point x^1 is far away from x^*.

Algorithm

Step 1. Input $x^1 =$ the initial point. Set $k = 1$.

Step 2. Repeat the following formula:

$$x^{k+1} := x^k - \frac{f'(x^k)}{f''(x^k)}$$

until $|f'(x^{k+1})| < \varepsilon$ or other stopping criterion is met.

Step 3. Output the last x^{k+1} as x^*.

Geometric Meaning of Newton's Method.

See the figure on page 39. Suppose the current iterative point is x^{k-1}. We approximate the curve $f'(x)$ by its tangent line at the point $(x^{k-1}, f'(x^{k-1}))$. What we need to find is the intersection point x^* of the curve $f'(x)$ with x-axis, and we use the intersection point of the tangent line with x-axis to approximate x^*. The latter is just x^k according to the formula of Newton's method.

3.4.2 Secant Method

Idea

- Assume that $f \in C^2[x_s, x_t]$ is a convex function.
- Assume that two distinct initial points $x^0, x^1 \in [x_s, x_t]$ are given, and they are sufficiently close to the minimum solution x^* with $f'(x^*) = 0$ and $f''(x^*) > 0$.
- Assume that $f'(x^{k-1})$ and $f'(x^k)$ are known at iteration k $(k = 1, 2, \ldots)$.

- Construct a quadratic function:

$$\begin{aligned} q(x) &= f(x^k) + f'(x^k)(x - x^k) \\ &\quad + \frac{1}{2}\left[\frac{f'(x^k) - f'(x^{k-1})}{x^k - x^{k-1}}\right](x - x^k)^2. \end{aligned}$$

Note that $q(x^k) = f(x^k)$ and $q'(x^k) = f'(x^k)$, but $q''(x^k) \neq f''(x^k)$. In fact

$$q''(x^k) = \left[\frac{f'(x^k) - f'(x^{k-1})}{x^k - x^{k-1}}\right] \approx f''(x^k).$$

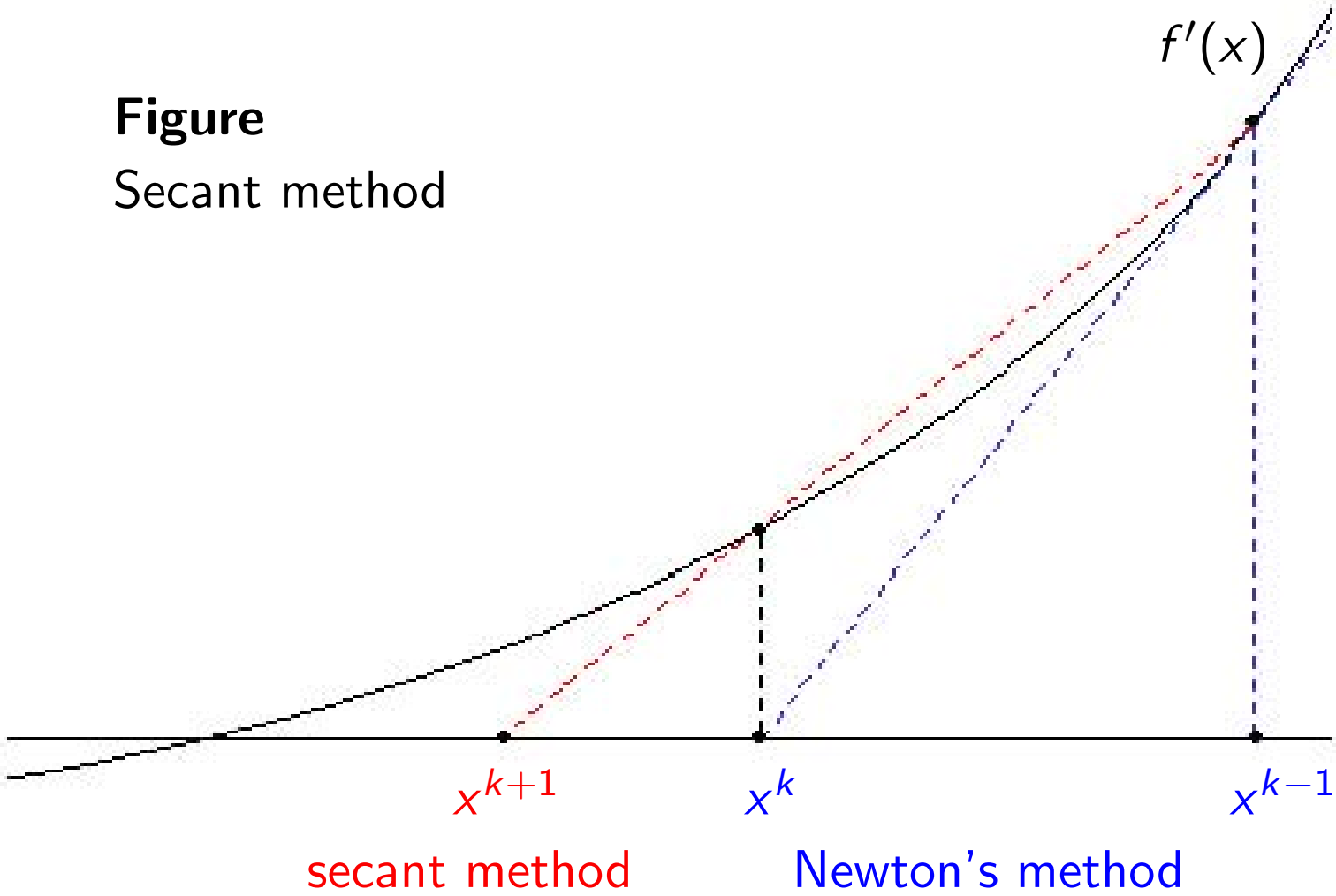
The geometric meaning is that in the graph of $g(x) = f'(x)$, the slope of the tangent line at x^k, i.e. $f''(x^k)$, is approximately replaced by the amount

$$\left[\frac{f'(x^k) - f'(x^{k-1})}{x^k - x^{k-1}}\right],$$

which is in fact the slope of the secant line passing through the two points $(x^{k-1}, f'(x^{k-1}))$ and $(x^k, f'(x^k))$ in the graph of function $f'(x)$. So, we used the secant line to replace the tangent line. That is the reason for the name of the method.

Figure

Secant method



- Let x^{k+1} be the minimum point of $q(x)$, i.e.,

$$x^{k+1} = x^k - \left[\frac{x^k - x^{k-1}}{f'(x^k) - f'(x^{k-1})}\right] f'(x^k).$$

Advantage

- Use first order derivatives only, and do NOT need to use second-order derivatives.

Disadvantage

- May be slower than Newton's Method.

Algorithm

Step 1. Input x^0, x^1 as two initial points.

Step 2. Repeat the following formula for $k = 1, 2, \ldots$

$$x^{k+1} := x^k - \left[\frac{x^{k-1} - x^k}{f'(x^{k-1}) - f'(x^k)}\right] f'(x^k)$$

until $|f'(x^{k+1})| < \varepsilon$ or other stopping criterion is met.

Step 3. Output the last x^{k+1} as x^*.

3.5 Speed of Convergence

- We often need to study how fast an algorithm converges to a minimum solution.
- **Definition.** Let the sequence $\{r_k\}$ converge to r^*. The *order of convergence* of $\{r_k\}$ is defined as the non-negative number p satisfying

$$0 < \lim_{k \to \infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^p} = \beta < \infty.$$

- If the sequence has a convergence order p, we have asymptotically,

$$|r_{k+1} - r^*| \approx \beta |r_k - r^*|^p.$$

Note that if a sequence $\{r_k\}$ converges to r^* with a convergence order p', then p' must be unique. That is, $\{r_k\}$ cannot have two different convergence orders.

In fact if $\{r_k\}$ has converges order p', then

$$\lim_{k\to\infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^{p'}} = \beta \quad (0 < \beta < \infty).$$

Then for any $p \neq p'$,

$$\begin{aligned} \lim_{k\to\infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^{p}} &= \lim_{k\to\infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^{p'}} \frac{|r_k - r^*|^{p'}}{|r_k - r^*|^{p}} \\ &= \begin{cases} 0 & \text{if } p < p' \\ \infty & \text{if } p > p' \end{cases} \end{aligned}$$

Therefore, p cannot be another convergence order.

Example 3.5.1:

- The sequence with $r_k = \frac{1}{k}$ converges to zero with order 1.
- The sequence with $r_k = a^k$, where $0 < a < 1$, converges to zero with order 1.
- The sequence with $r_k = a^{2^k}$, where $0 < a < 1$, converges to zero with order two.

Remark 3.5.1 Some convergent sequences may not have a convergence order. For example, in the above example, we know that

- sequence $r_k = \frac{1}{k}$ converges to 0 with order 1;
- sequence $r'_k = a^{2^k}$ $(0 < a < 1)$ converges to 0 with order 2.

Now if we mix the two sequences as

$$S = \{r_1, r'_1, r_2, r'_2, r_3, r'_3, \ldots\ldots\},$$

then the resulting sequence has no convergence order. In fact let the sequence be $S = \{s_1, s_2, s_3, \ldots\}$, then

$$s_{2k} = r'_k, \quad k = 1, 2, 3 \ldots$$

$$s_{2k-1} = r_k, \quad k = 1, 2, 3 \ldots$$

and

$$\lim_{k\to\infty} \frac{|s_{2k} - 0|}{|s_{2k-1} - 0|^p} = \lim_{k\to\infty} \frac{|r'_k|}{|r_k|^p} = \lim_{k\to\infty} \frac{a^{2^k}}{(\frac{1}{k})^p} = 0, \quad \forall p > 0,$$

$$\lim_{k\to\infty} \frac{|s_{2k+1} - 0|}{|s_{2k} - 0|^p} = \lim_{k\to\infty} \frac{|r_{k+1}|}{|r'_k|^p} = \lim_{k\to\infty} \frac{\frac{1}{k+1}}{(a^{2^k})^p} = \infty, \quad \forall p > 0.$$

Therefore, for any $p > 0$,

$$\lim_{k\to\infty} \frac{|s_{k+1} - 0|}{|s_k - 0|^p}$$

does not exist.

The following three convergence speeds are often met:

1. if $p = 1$ and $0 < \beta < 1$, then we say that r_k converges to r^* *linearly*;
2. if $p = 2$ and $0 < \beta$, then we say that r_k converges to r^* *quadratically*;
3. if $1 < p$, or if $p = 1$ and $\beta = 0$, then we say that r_k converges to r^* *superlinearly*.

Linear rate is a relatively slow rate. For a linearly convergent sequence $\{r_k\}$, when k is large, the errors are approximately

$$|r_{k+1} - r^*| \approx \beta |r_k - r^*|.$$

Especially, if β is close to 1, $\{r_k\}$ converges very slowly.

Quadratic convergence is very quick. We know that the third example above has order 2. If $a = \frac{1}{10}$, then the sequence $r_k = a^{2^k}$ becomes

$$\frac{1}{10^2}, \frac{1}{10^4}, \frac{1}{10^8}, \frac{1}{10^{16}}, \ldots$$

which tends to 0 very quickly.

For superlinear convergence, in the first case in the definition, as $p > 1$,

$$\frac{|r_{k+1} - r^*|}{|r_k - r^*|} = \frac{|r_{k+1} - r^*|}{|r_k - r^*|^p} \cdot |r_k - r^*|^{p-1} \to \beta \cdot 0 = 0,$$

which becomes the second case. Hence we can directly define superlinear convergence as

$$\lim_{k \to \infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|} = 0.$$

Superlinear convergence is considered as a quick convergence, even though it may not converge as fast as quadratic convergence.

Example 3.5.2 The sequence $\{r_k\}$ defined by $r_1 = 4$ and

$$r_{k+1} = 1 + (r_k - 1)/2^k, \quad \text{for } k = 1, 2, \ldots$$

i.e.,

$$\{r_k\} = 4, 2.5, 1.375, 1.0469, \ldots$$

meets the condition:

$$\lim_{k \to \infty} \frac{|r_{k+1} - 1|}{|r_k - 1|} = \lim_{k \to \infty} \frac{1}{2^k} = 0.$$

Hence $\{r_k\}$ converges to 1 superlinearly.

Remark 3.5.2 The sequence $\{r_k\}$ in Example 3.5.2 converges to 1 superlinearly, but it does not have a definite convergence order.

In fact we have seen that when $p = 1$,

$$\lim_{k\to\infty} \frac{|r_{k+1} - 1|}{|r_k - 1|^p} = \lim_{k\to\infty} \frac{r_{k+1} - 1}{r_k - 1} = 0.$$

As the limit is 0, $p = 1$ is not the convergence order. It seems that the convergence order is higher than 1.

But this is not true. We can prove that for any $p > 1$,

$$\lim_{k\to\infty} \frac{|r_{k+1} - 1|}{|r_k - 1|^p} = \infty. \tag{8}$$

So, the sequence also cannot have a convergence order $p > 1$.

We now prove (8). The sequence satisfies:

$$r_2 - 1 = \frac{1}{2}(r_1 - 1) = \frac{1}{2} \cdot 3,$$

$$r_3 - 1 = \frac{1}{2^2}(r_2 - 1) = \frac{1}{2} \cdot \frac{1}{2^2} \cdot 3,$$

$$\cdots\cdots$$

In general,

$$r_{k+1} - 1 = \frac{1}{2} \cdot \frac{1}{2^2} \cdots \frac{1}{2^k} \cdot 3.$$

Let

$$\alpha_k = \frac{|r_{k+1} - 1|}{|r_k - 1|^p} = \frac{r_{k+1} - 1}{(r_k - 1)^p} \quad (p > 1).$$

Then we may obtain

$$\alpha_k = \frac{(2 \cdot 2^2 \cdots 2^{k-1})^{p-1}}{2^k} \cdot 3^{1-p}.$$

As

$$2 \cdot 2^2 \cdots 2^{k-1} = 2^{1+2+\cdots+(k-1)} = 2^{\frac{k(k-1)}{2}},$$

$$\alpha_k = 2^{\frac{k(k-1)}{2}(p-1)-k} \cdot 3^{1-p}.$$

Since $p > 1$,

$$\frac{k(k-1)}{2}(p-1) - k = k[\frac{k-1}{2}(p-1) - 1] \to \infty.$$

Therefore,

$$\alpha_k \to \infty.$$

So far we have introduced the concept of convergence order for a sequence of numbers $\{r_k\}$. For a sequence of points (vectors) x^k in R^n converging to a point (vector) x^*, we may define convergence order of $\{x^k\}$ by the convergence order of the sequence $\{\|x^k - x^*\|\}$. For example, we say that x^k converges to x^* superlinearly if $\|x^k - x^*\|$ converges to 0 superlinearly, i.e.,

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

Later we shall see that **different algorithms may have different convergence orders**. Here for the single-variable Newton's method and secant method, we give their convergence orders.

Theorem 1. Let function f have a continuous second order derivative, and let x^* satisfy $f'(x^*) = 0$, $f''(x^*) \neq 0$. Then, provided x^0 is sufficiently close to x^*, the sequence $\{x^k\}_{k=0}^{\infty}$ generated by Newton's method converges to x^* with an order of convergence at least two.

Theorem 2. Let function f have a continuous second order derivative, and let x^* satisfy $f'(x^*) = 0$, $f''(x^*) \neq 0$. Then, provided x^0 and x^1 are sufficiently close to x^*, the sequence $\{x^k\}_{k=0}^{\infty}$ generated by the secant method converges to x^* with order $\tau_1 \approx 1.618$.

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.

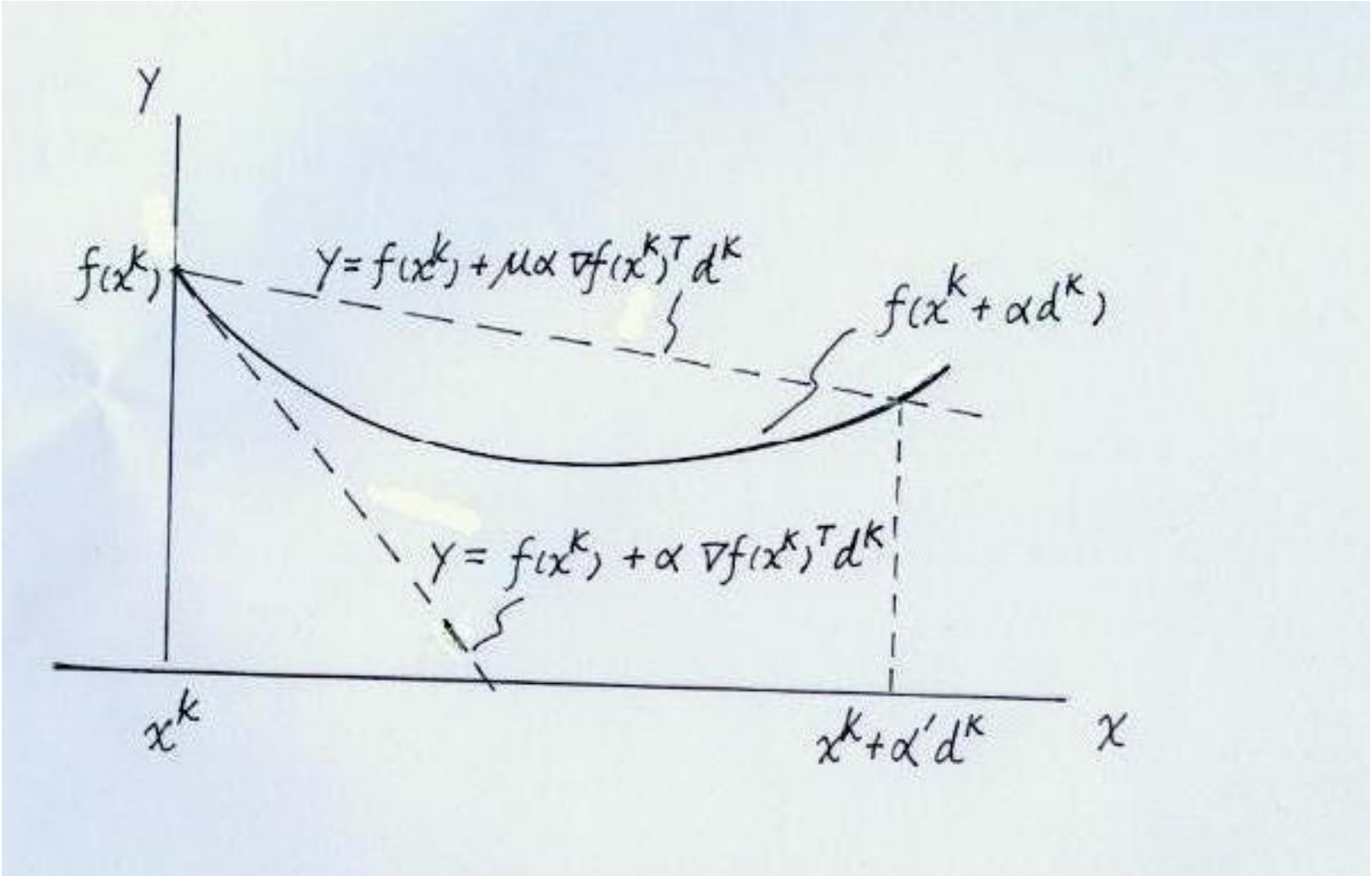
It is found that in order to guarantee global convergence, letting $f(x^k + \alpha_k^* d^k)$ be smaller than $f(x^k)$ is not enough (sometimes it is successful such as the example on Pages 43-44, but not always so), and we need to ask $f(x^k + \alpha_k^* d^k)$ to be smaller than $f(x^k)$ by a certain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

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

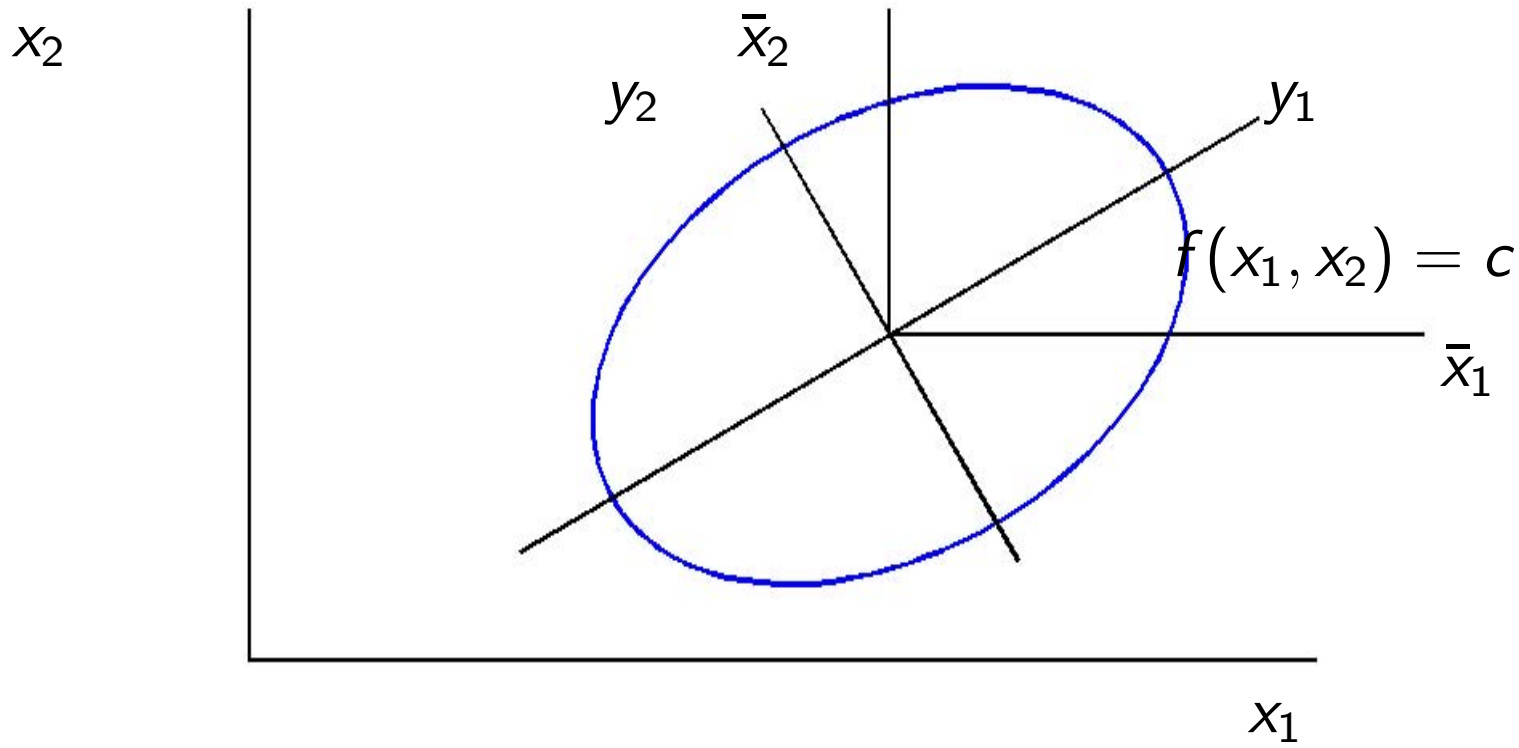
That is, when we change variables x into y, function f contains only pure square terms y_i^2 and has no cross product terms such as $\beta y_i y_j$ for $i \neq j$. Actually, changing variables from $\bar{x}$ to y represents a rotation and enlargement/reduction of the coordinate system.

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

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

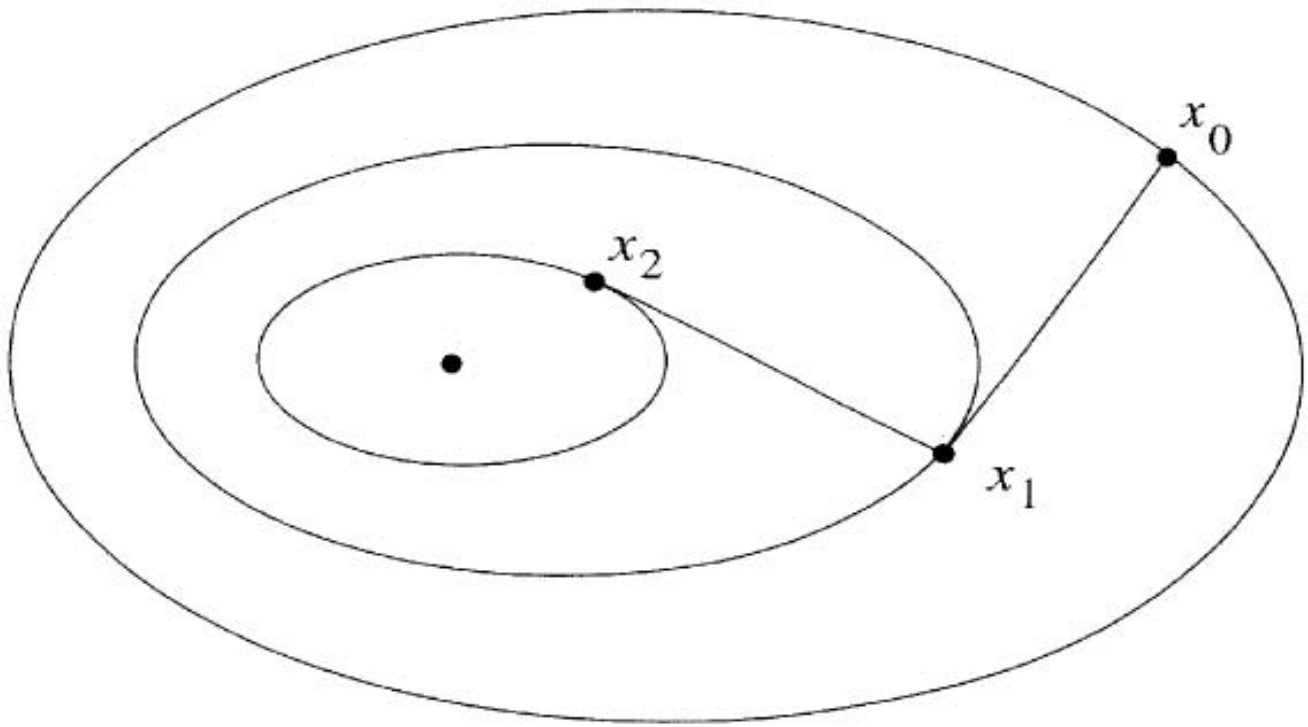


So, if all eigenvalues d_i $(i = 1, \ldots, n)$ are equal, the surface is a spherical surface, whereas if the largest and the smallest eigenvalues of Q have big difference, the surface would be very flat in certain direction.

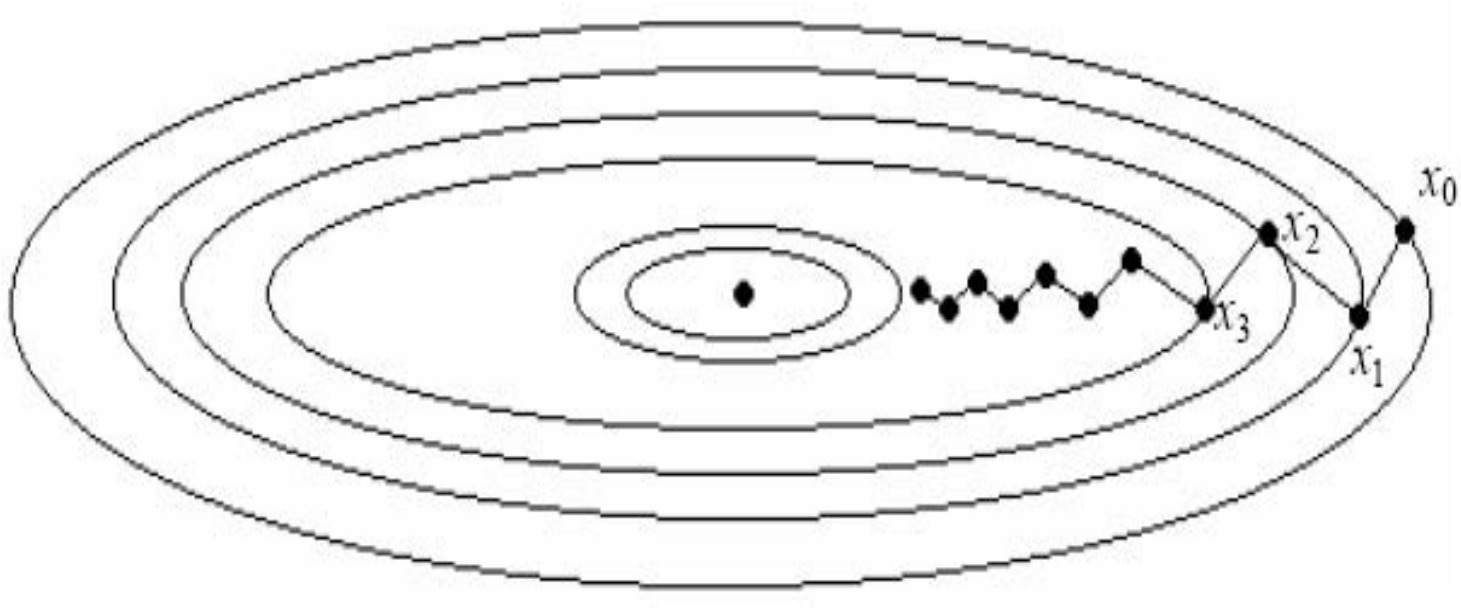
4. For any continuously differentiable function $f(x)$, at each point x^* on the contour $f(x) = c$, the normal vector of the contour at point x^* is the vector $\nabla f(x^*)$ that points to the increasing direction of $f(x)$.

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.

OR4030 OPTIMIZATION Chapter 6

Optimality Conditions for Constrained Optimization Problems

6.1 Equality Constrained Optimization

6.1.1 Problem Description

Problem:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & h_i(x) = 0, \quad i = 1, \ldots, m \\ & x \in \Re^n \end{array}$$

where

- $m \leq n$;
- $f, h_1, \ldots, h_m : \Re^n \mapsto \Re$;
- $f, h_1, \ldots, h_m \in C^2(\Re^n)$.

In this chapter, we shall learn necessary or sufficient optimality conditions for above optimization problem without giving proofs.

6.1.2 First-Order Necessary Conditions

Lagrange Theorem

- Let x^* be a local extreme (either a maximum or a minimum) point of f subject to the constraints $h(x) = (h_1(x), \ldots, h_m(x))^T = 0$.
- Assume further that x^* satisfies certain regularity conditions. (e.g., all $\nabla h_i(x^*)$ are linearly independent. In this course it is not required to know these conditions in detail.)
- Then there exists

$$\lambda^* \in \Re^m$$

(called *Lagrange Multipliers*) such that

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) = 0.$$

- A feasible point that satisfies the above condition is called a **stationary point**. Note that such a stationary point may be a constrained minimum point, or a constrained maximum point, or neither (i.e., only a saddle point).
- It should be noted that the first-order necessary conditions

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) = 0 \quad (n \text{ equations})$$

together with the constraints

$$h(x^*) = 0 \quad (m \text{ equations})$$

gives $n + m$ (generally nonlinear) equations in the $n + m$ variables comprising x^* and λ^*.

- Define the *Lagrangian function* as follows:

$$L(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i h_i(x).$$

The first-order necessary conditions together with the constraints can then be expressed in the form:

$$\nabla_x L(x^*, \lambda^*) = 0 \quad \text{and} \quad \nabla_\lambda L(x^*, \lambda^*) = 0,$$

or equivalently,

$$\nabla L(x^*, \lambda^*) = 0.$$

6.1.3 Second-Order Necessary Conditions

- Let x^* be a local minimum point of f subject to $h(x) = 0$. Assume that x^* satisfies certain regularity conditions. Then there exists

$$\lambda^* \in \Re^m$$

such that

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) = 0,$$

and

$$y^T \left[\nabla^2 f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 h_i(x^*) \right] y \geq 0$$

for all $y \in \Re^n$ satisfying

$$\begin{aligned} y &\neq 0 \\ \nabla h_i(x^*)^T y &= 0 \qquad \text{for} \quad i = 1, \ldots, m. \end{aligned}$$

- The second-order necessary conditions can be expressed in the form:

$$\nabla L(x^*, \lambda^*) = 0,$$

and

$$y^T \nabla^2_{xx} L(x^*, \lambda^*) y \geq 0 \tag{1}$$

for all $y \in \Re^n$ satisfying

$$\begin{aligned} y &\neq 0 \\ \nabla h_i(x^*)^T y &= 0 \qquad \text{for} \quad i = 1, \ldots, m. \end{aligned}$$

Remark. The above condition for $\nabla^2_{xx} L(x^*, \lambda^*)$ is *weaker* than asking the matrix $\nabla^2_{xx} L(x^*, \lambda^*)$ to be positive semidefinite. Why?

- If x^* is a local maximum point of f subject to $h(x) = 0$, then in the above conditions, (1) is changed to

$$y^T \nabla^2_{xx} L(x^*, \lambda^*) y \leq 0.$$

6.1.4 Second-Order Sufficient Conditions

Just like in the unconstrained case, if, additionally, x^* satisfies

$$y^T \nabla^2_{xx} L(x^*, \lambda^*) y > 0$$

for all $y \in \Re^n$ satisfying

$$\begin{aligned} y &\neq 0 \\ \nabla h_i(x^*)^T y &= 0 \qquad \text{for} \quad i = 1, \ldots, m, \end{aligned}$$

then x^* is a strict local minimum point of f subject to $h(x) = 0$.

For the maximum case, if the above inequality is changed to

$$y^T \nabla^2_{xx} L(x^*, \lambda^*) y < 0$$

for such y, then x^* is a strict local maximum point of f subject to $h(x) = 0$.

6.1.5 Examples

Example 6.1 Solve the problem:

$$\begin{array}{lrcccccc} \text{minimize} & f(x) & = & x_1^2 & + & x_2^2 & & \\ \text{subject to} & h(x) & = & x_1 & + & x_2 & - \ 1 & = \ 0. \end{array}$$

Solution: The Lagrangian function is

$$L(x, \lambda) = x_1^2 + x_2^2 + \lambda(x_1 + x_2 - 1).$$

Thus,

$$\nabla L(x, \lambda) = \begin{bmatrix} 2x_1 + \lambda \\ 2x_2 + \lambda \\ x_1 + x_2 - 1 \end{bmatrix} \quad \text{and} \quad \nabla_{xx}^2 L(x, \lambda) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

The first-order necessary conditions are

$$\begin{aligned} 2x_1^* + \lambda^* &= 0, \\ 2x_2^* + \lambda^* &= 0, \\ x_1^* + x_2^* - 1 &= 0. \end{aligned}$$

We may solve the system of equations as

$$\begin{bmatrix} x_1^* \\ x_2^* \\ \lambda^* \end{bmatrix} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \\ -1 \end{bmatrix}.$$

But usually we use the elimination method to solve these equations more quickly. In fact from the first two equations we know that $x_1^* = x_2^*$ (i.e., eliminate λ^*). Substituting this result into the third equation, we obtain $x_1^* = x_2^* = 0.5$. Finally, from the first or the second equation, we obtain $\lambda^* = -1$.

Since

$$\nabla^2_{xx} L(x^*, \lambda^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

is positive definite, (x^*, λ^*) satisfies the second-order sufficient conditions. Therefore

$$\begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

is a strict local minimum solution to the problem.

Example 6.2 Solve the problem:

$$\begin{array}{lrcl} \text{minimize} & f(x) & = & -x_1x_2 - x_2x_3 - x_1x_3 \\ \text{subject to} & h(x) & = & x_1 + x_2 + x_3 - 3 = 0. \end{array}$$

Solution The Lagrangian function is

$$L(x,\lambda) = -x_1x_2 - x_2x_3 - x_1x_3 + \lambda(x_1 + x_2 + x_3 - 3).$$

Thus,

$$\nabla L(x,\lambda) = \begin{bmatrix} -x_2 - x_3 + \lambda \\ -x_1 - x_3 + \lambda \\ -x_1 - x_2 + \lambda \\ x_1 + x_2 + x_3 - 3 \end{bmatrix}, \nabla^2_{xx} L(x,\lambda) = \begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & 0 \end{bmatrix}.$$

The first-order necessary conditions are

$$\begin{aligned} -x_2^* - x_3^* + \lambda^* &= 0, \\ -x_1^* - x_3^* + \lambda^* &= 0, \\ -x_1^* - x_2^* + \lambda^* &= 0, \\ x_1^* + x_2^* + x_3^* - 3 &= 0, \end{aligned}$$

which has a solution $\begin{bmatrix} x_1^*, & x_2^*, & x_3^*, & \lambda^* \end{bmatrix}^T = \begin{bmatrix} 1, & 1, & 1, & 2 \end{bmatrix}^T$.

Since

$$\nabla^2_{xx} L(x^*, \lambda^*) = \begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & 0 \end{bmatrix}$$

is not positive definite, we cannot make conclusion immediately.

Let $y \in \Re^3$ satisfying $y \neq 0$ and

$$0 = \nabla h(x^*)^T y = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = y_1 + y_2 + y_3.$$

For such y,

$$\begin{aligned} y^T \nabla^2_{xx} L(x^*, \lambda^*) y &= \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \\ &= -y_1(y_2 + y_3) - y_2(y_1 + y_3) - y_3(y_1 + y_2) \\ &= y_1^2 + y_2^2 + y_3^2 > 0. \end{aligned}$$

So, (x^*, λ^*) satisfies the second order sufficient conditions, and

$$\begin{bmatrix} x_1^*, & x_2^*, & x_3^* \end{bmatrix}^T = \begin{bmatrix} 1, & 1, & 1 \end{bmatrix}^T$$

is a strict local minimum solution to the problem.

Example 6.3 Solve the problem:

$$\begin{array}{lrcrcrcrcl} \text{minimize} & f(x) & = & x_1^2 & + & x_2^2 & + & x_3^2 & & \\ \text{subject to} & h_1(x) & = & x_1 & + & x_2 & + & 3x_3 & - \quad 2 & = \quad 0 \\ & h_2(x) & = & 5x_1 & + & 2x_2 & + & x_3 & - \quad 5 & = \quad 0. \end{array}$$

Solution The Lagrangian function is

$$\begin{aligned} L(x,\lambda) &= x_1^2 + x_2^2 + x_3^2 + \lambda_1(x_1 + x_2 + 3x_3 - 2) \\ &\quad + \lambda_2(5x_1 + 2x_2 + x_3 - 5). \end{aligned}$$

Thus,

$$\nabla L(x,\lambda) = \begin{bmatrix} 2x_1 + \lambda_1 + 5\lambda_2 \\ 2x_2 + \lambda_1 + 2\lambda_2 \\ 2x_3 + 3\lambda_1 + \lambda_2 \\ x_1 + x_2 + 3x_3 - 2 \\ 5x_1 + 2x_2 + x_3 - 5 \end{bmatrix} \text{ and } \nabla_{xx}^2 L(x,\lambda) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

The first-order necessary conditions are

$$\begin{aligned} 2x_1^* + \lambda_1^* + 5\lambda_2^* &= 0, \\ 2x_2^* + \lambda_1^* + 2\lambda_2^* &= 0, \\ 2x_3^* + 3\lambda_1^* + \lambda_2^* &= 0, \\ x_1^* + x_2^* + 3x_3^* - 2 &= 0, \\ 5x_1^* + 2x_2^* + x_3^* - 5 &= 0. \end{aligned}$$

From the first three equations we eliminate variables λ_1^* and λ_2^*, and obtain the equation $5x_1^* - 14x_2^* + 3x_3^* = 0$. Then together with the last two equations we obtain x_1^*, x_2^* and x_3^*. Finally,

$$(x_1^*,\ x_2^*,\ x_3^*,\ \lambda_1^*,\ \lambda_2^*) \approx (0.8043, 0.3478,\ 0.2826, -0.0870, -0.3043).$$

Since

$$\nabla^2_{xx} L(x^*, \lambda^*) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

is positive definite, (x^*, λ^*) satisfies the second-order sufficient conditions. Therefore

$$\begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \end{bmatrix} \approx \begin{bmatrix} 0.8043 \\ 0.3478 \\ 0.2826 \end{bmatrix}$$

is a strict local minimum solution to the problem.

6.2.1 Problem Description

Problem:

$$\begin{array}{lllll} \min & f(x) & & & \\ \text{s.t.} & h_i(x) & = & 0, & i = 1, \ldots, m \\ & g_j(x) & \leq & 0, & j = 1, \ldots, r \\ & x \in \Re^n & & & \end{array}$$

where

1. $m \leq n$;
2. $f, h_1, \ldots, h_m, g_1, \ldots, g_r : \Re^n \mapsto \Re$;
3. $f, h_1, \ldots, h_m, g_1, \ldots, g_r \in C^2(\Re^n)$.

6.2.2 First-Order Necessary Conditions

Karosh-Kuhn-Tucker (KKT) Conditions

- Let x^* be a local minimum point of f subject to the constraints
  $h(x) = (h_1(x), \ldots, h_m(x))^T = 0$ and
  $g(x) = (g_1(x), \ldots, g_r(x))^T \leq 0$.
- Assume further that x^* satisfies certain regularity conditions.
- Then there exists

$$\lambda_i^* \in \Re \quad \text{for} \quad i = 1, \ldots, m$$
$$\mu_j^* \geq 0 \quad \text{for} \quad j = 1, \ldots, r$$

(λ^* and μ^* are called *KKT multipliers*) such that

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) = 0$$
$$\mu_j^* g_j(x^*) = 0 \quad \text{for} \quad j = 1, \ldots, r$$

- A feasible point x^* that satisfies the above KKT conditions (together with a set of multipliers λ^* and μ^*) is called a **stationary point**, or a **KKT point**.
- Define the *Lagrangian function* as follows:

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{m} \lambda_i h_i(x) + \sum_{j=1}^{r} \mu_j g_j(x).$$

The first-order necessary conditions together with the equality constraints can then be expressed as the following system of equations:

$$\begin{aligned} \nabla_x L(x^*, \lambda^*, \mu^*) &= 0 && (n \text{ equations}) \\ h(x^*) &= 0 && (m \text{ equations}) \\ \mu_j^* g_j(x^*) &= 0 \quad \text{for} \quad j = 1, \ldots, r && (r \text{ equations}) \end{aligned}$$

which contains $n + m + r$ equations for $n + m + r$ unknown variables $x^* \in R^n$, $\lambda^* \in R^m$, and $\mu^* \in R^r$.

- Note that the conditions also ask that

$$g_j(x^*) \leq 0, \ \mu_j^* \geq 0 \ \text{ and } \mu_j^* g_j(x^*) = 0, \quad \text{for } \ j = 1, \ldots, r. \tag{2}$$

  The last request is called the *complementary slackness condition* which asks that in each pair of μ_j^* and $g_j(x^*)$, at least one of them must be 0.
  (This condition is a natural extension of the complementary slackness condition in linear programming.)
- If in (2), it is requested further that in each pair of μ_j^* and $g_j(x^*)$, *exactly one* is zero, it is called the *strict complementary slackness condition*. That is, if $g_j(x^*) < 0$, then $\mu_j^* = 0$, and if $g_j(x^*) = 0$, then $\mu_j^* > 0$.

6.2.3 Second-Order Necessary Conditions

- Suppose that x^* is a local minimum point of f subject to the constraints $h(x) = 0$ and $g(x) \leq 0$.
- Assume that x^* satisfies certain regularity conditions.
- Then there exists

$$\begin{aligned} \lambda_i^* &\in \Re \qquad \text{for} \qquad i = 1, \ldots, m, \\ \mu_j^* &\geq 0 \qquad \text{for} \qquad j = 1, \ldots, r, \end{aligned}$$

such that

$$\begin{aligned} \nabla_x L(x^*, \lambda^*, \mu^*) &= 0 && (n \text{ equations}) \\ h(x^*) &= 0 && (m \text{ equations}) \\ \mu_j^* g_j(x^*) &= 0 \quad \text{for } j = 1, \ldots, r && (r \text{ equations}) \end{aligned}$$

and

$$y^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) y \geq 0 \tag{3}$$

for all $y \in \Re^n$ satisfying

$$\begin{aligned} y &\neq 0 \\ \nabla h_i(x^*)^T y &= 0 \qquad \text{for} \qquad i = 1, \ldots, m \\ \nabla g_j(x^*)^T y &= 0 \qquad \text{for} \qquad j \in A(x^*) \end{aligned}$$

where $A(x^*) = \{j : g_j(x^*) = 0\}$.

We call $g_j(x) \leq 0$ an *active or binding constraint* at x^* if the index $j \in A(x^*)$. For example, suppose the problem has two inequality constraints:

$$\begin{aligned} g_1(x) &= x_1 + 2x_2 - 3 \leq 0, \\ g_2(x) &= 2x_1 + x_2 - 5 \leq 0. \end{aligned}$$

At point $x^* = (1, 1)$,

$$\begin{aligned} g_1(x^*) &= x_1^* + 2x_2^* - 3 = 0, \\ g_2(x^*) &= 2x_1^* + x_2^* - 5 < 0. \end{aligned}$$

We see that at x^*, the first constraint is active, but the second one is not. Hence $A(x^*) = \{1\}$.

6.2.4 Second-Order Sufficient Conditions

The above conditions become sufficient if

1. the inequality (3) is further strengthened to

$$y^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) y > 0$$

for all $y \in \Re^n$ satisfying

$$\begin{aligned} y &\neq 0 \\ \nabla h_i(x^*)^T y &= 0 \qquad \text{for} \qquad i = 1, \ldots, m \\ \nabla g_j(x^*)^T y &= 0 \qquad \text{for} \qquad j \in A(x^*), \end{aligned}$$

and

2. $\mu^*_j > 0$ for $j \in A(x^*)$.

Note that Condition 2 is in fact the strict complementary slackness condition.

6.2.5 Examples

Example 6.4 Solve the problem:

$$\begin{array}{lrcrcrcrl} \text{minimize} & f(x) & = & x_1^2 & + & x_2^2 & & & \\ \text{subject to} & g(x) & = & -x_1 & - & x_2 & + & 1 & \leq 0. \end{array}$$

Solution The Lagrangian function is

$$L(x, \mu) = x_1^2 + x_2^2 + \mu(-x_1 - x_2 + 1).$$

Thus,

$$\nabla_x L(x, \mu) = \begin{bmatrix} 2x_1 - \mu \\ 2x_2 - \mu \end{bmatrix} \quad \text{and} \quad \nabla_{xx}^2 L(x, \mu) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

The KKT necessary conditions are

$$\begin{aligned} 2x_1^* - \mu^* &= 0, && (4)\\ 2x_2^* - \mu^* &= 0, && (5)\\ \mu^*(-x_1^* - x_2^* + 1) &= 0, && (6)\\ \mu^* &\geq 0. && (7) \end{aligned}$$

We now consider the following two cases.

Case 1: Suppose

$$-x_1^* - x_2^* + 1 < 0. \quad (8)$$

(8) and (6) imply

$$\mu^* = 0. \quad (9)$$

Substituting (9) into (4) and (5) gives

$$(x_1^*, x_2^*) = (0, 0). \quad (10)$$

Since (10) contradicts (8), case 1 is rejected.

Case 2: Suppose

$$-x_1^* - x_2^* + 1 = 0. \tag{11}$$

Rewrite (4), (5) and (11) as

$$\begin{bmatrix} 2 & 0 & -1 \\ 0 & 2 & -1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ \mu^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

We thus have

$$(x_1^*, x_2^*, \mu^*) = (0.5, 0.5, 1). \tag{12}$$

Since $\mu^* \geq 0$,

$$(x_1^*, x_2^*) = (0.5, 0.5)$$

is a KKT point.

This point may be a potential minimum solution to the problem.

Since

$$\nabla^2_{xx} L(x^*, \mu^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

is positive definite and $\mu^* > 0$, (x^*, μ^*) satisfies the second-order sufficient conditions. Therefore

$$\begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

is a strict local minimum solution to the problem.

Example 6.5 Solve the problem:

$$\begin{array}{lrcrcrcl} \text{minimize} & f(x) & = & x_1^3 & + & x_2^2 & & \\ \text{subject to} & g(x) & = & -x_1 & & & - \quad 1 \quad \leq & 0. \end{array}$$

Solution The Lagrangian function is

$$L(x, \mu) = x_1^3 + x_2^2 + \mu(-x_1 - 1).$$

Thus,

$$\nabla_x L(x, \mu) = \begin{bmatrix} 3x_1^2 - \mu \\ 2x_2 \end{bmatrix} \quad \text{and} \quad \nabla_{xx}^2 L(x, \mu) = \begin{bmatrix} 6x_1 & 0 \\ 0 & 2 \end{bmatrix}.$$

The KKT necessary conditions are

$$\begin{aligned} 3(x_1^*)^2 - \mu^* &= 0, && (13) \\ x_2^* &= 0, && (14) \\ \mu^*(-x_1^* - 1) &= 0, && (15) \\ \mu^* &\geq 0. && (16) \end{aligned}$$

We now consider the following two cases.

Case 1: Suppose

$$-x_1^* - 1 < 0. \tag{17}$$

(17) and (15) imply

$$\mu^* = 0. \tag{18}$$

Substituting (18) into (13) gives

$$x_1^* = 0. \tag{19}$$

As (19) satisfies (17), $(x_1^*, x_2^*) = (0, 0)$ with $\mu^* = 0$ is a potential minimum solution to the problem.

Since at (x^*, y^*), there is no any active constraint, the second order optimality condition can be simplified.

- necessary condition: $\nabla^2_{xx} L(x^*, \mu^*)$ is p.s.d.;
- sufficient condition: $\nabla^2_{xx} L(x^*, \mu^*)$ is p.d.

We consider

$$\nabla^2_{xx} L(x^*, \mu^*) = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}. \tag{20}$$

As (20) is positive semi-definite, (x^*, μ^*) satisfies the second-order necessary conditions but does not satisfy the sufficient conditions to be a strict local minimizer, the identity of the point $(x_1^*, x_2^*) = (0, 0)$ is unknown from the above theorems. But in this particular example, it is easy to see that x^* is not a local minimizer. In fact if we take $\bar{x} = (-\epsilon, 0)^T$ with very small positive value ϵ, it is obvious that $\bar{x}$ is very close to x^* and feasible, but $f(\bar{x}) < f(x^*)$.

Case 2: Suppose

$$x_1^* = -1. \tag{21}$$

Substituting (21) into (13) gives

$$\mu^* = 3. \tag{22}$$

Since (22) satisfies (16),

$$(x_1^*, x_2^*) = (-1, 0)$$

with $\mu^* = 3$ is a potential minimum solution to the problem.

Note that $\mu^* > 0$ satisfies the sufficient condition to be a strict local minimum solution to the problem.

Consider

$$\nabla_{xx}^2 L(x^*, \mu^*) = \begin{bmatrix} -6 & 0 \\ 0 & 2 \end{bmatrix}. \tag{23}$$

Since (23) is indefinite, we cannot make conclusion immediately.

Since the constraint is active, we consider $y = (y_1, y_2) \neq (0, 0)$ such that

$$0 = \nabla g(x^*)^T y = \begin{bmatrix} -1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = -y_1. \tag{24}$$

This implies

$$y_2 \neq 0. \tag{25}$$

For such y,

$$\begin{aligned} y^T \nabla^2_{xx} L(x^*, \mu^*) y &= \begin{bmatrix} 0 & y_2 \end{bmatrix} \begin{bmatrix} -6 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ y_2 \end{bmatrix} \\ &= 2y_2^2 > 0. \end{aligned}$$

Therefore $(x_1^*, x_2^*) = (-1, 0)$ is a strict local minimum solution to the problem.

Example 6.6 Solve the problem:

$$\begin{array}{lrcrcrcrcl} \text{minimize} & f(x) & = & x_1^2 & + & x_2^2 & & & & \\ \text{subject to} & g_1(x) & = & -x_1 & - & x_2 & + & 1 & \leq & 0, \\ & g_2(x) & = & -x_1 & & & + & 2 & \leq & 0. \end{array}$$

Solution The Lagrangian function is

$$L(x, \mu) = x_1^2 + x_2^2 + \mu_1(-x_1 - x_2 + 1) + \mu_2(-x_1 + 2).$$

Thus,

$$\nabla_x L(x, \mu) = \begin{bmatrix} 2x_1 - \mu_1 - \mu_2 \\ 2x_2 - \mu_1 \end{bmatrix} \quad \text{and} \quad \nabla_{xx}^2 L(x, \mu) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

The KKT necessary conditions are

$$
\begin{aligned}
2x_1^* - \mu_1^* - \mu_2^* &= 0, && (26)\\
2x_2^* - \mu_1^* &= 0, && (27)\\
\mu_1^*(-x_1^* - x_2^* + 1) &= 0, && (28)\\
\mu_2^*(-x_1^* + 2) &= 0, && (29)\\
\mu_1^* &\geq 0, && (30)\\
\mu_2^* &\geq 0. && (31)
\end{aligned}
$$

We now consider the following four cases.

Case 1: Suppose that

$$-x_1^* - x_2^* + 1 < 0, \text{ and} \tag{32}$$

$$-x_1^* + 2 < 0. \tag{33}$$

(32) and (28) imply

$$\mu_1^* = 0. \tag{34}$$

(33) and (29) imply

$$\mu_2^* = 0. \tag{35}$$

Rewrite (26), (27), (34) and (35) as

$$\begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ \mu_1^* \\ \mu_2^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

and we thus have

$$(x_1^*, x_2^*, \mu_1^*, \mu_2^*) = (0, 0, 0, 0). \tag{36}$$

Since (36) contradicts (32) and (33), case 1 is rejected.

Case 2: Suppose that

$$-x_1^* - x_2^* + 1 = 0, \text{ and} \tag{37}$$

$$-x_1^* + 2 < 0. \tag{38}$$

(38) and (29) imply

$$\mu_2^* = 0. \tag{39}$$

Rewrite (26), (27), (37) and (39) as

$$\begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ \mu_1^* \\ \mu_2^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}.$$

We thus have

$$(x_1^*, x_2^*, \mu_1^*, \mu_2^*) = (0.5, 0.5, 1, 0). \tag{40}$$

Since (40) contradicts (38), case 2 is rejected.

Case 3: Suppose that

$$-x_1^* - x_2^* + 1 < 0, \text{ and} \tag{41}$$

$$x_1^* = 2. \tag{42}$$

(41) and (28) imply

$$\mu_1^* = 0. \tag{43}$$

Rewrite (26), (27), (42) and (43) as

$$\begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ \mu_1^* \\ \mu_2^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}.$$

We thus have

$$(x_1^*, x_2^*, \mu_1^*, \mu_2^*) = (2, 0, 0, 4). \tag{44}$$

Since (44) satisfies (31) and (41),

$$\begin{bmatrix} x_1^* \\ x_2^* \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad \text{with} \quad \begin{bmatrix} \mu_1^* \\ \mu_2^* \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$$

is a potential minimum solution to the problem.

As

$$\nabla_{xx}^2 L(x^*, \mu^*) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

is positive definite, and only the second constraint is active which corresponds to the multiplier $\mu_2^* > 0$, we see that (x^*, μ^*) satisfies the sufficient conditions. Therefore $(x_1^*, x_2^*) = (2, 0)$ is a strict local minimum solution to the problem.

Case 4: Suppose that

$$-x_1^* - x_2^* + 1 = 0, \text{ and} \tag{45}$$

$$x_1^* = 2. \tag{46}$$

Rewrite (26), (27), (45) and (46) as

$$\begin{bmatrix} 2 & 0 & -1 & -1 \\ 0 & 2 & -1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1^* \\ x_2^* \\ \mu_1^* \\ \mu_2^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 2 \end{bmatrix}.$$

We thus have

$$(x_1^*, x_2^*, \mu_1^*, \mu_2^*) = (2, -1, -2, 6). \tag{47}$$

As (47) violates (30), case 4 is rejected.

To summarize, this example has a unique local minimum solution $x^* = (2, 0)^T$.

So far we assume that all inequality constraints are $g_j(x) \leq 0$ type. If all inequality constraints are $g_j(x) \geq 0$ type, i.e., the problem is

$$\begin{array}{lll} \min & f(x) & \\ \text{s.t.} & h_i(x) = 0, & i = 1, \ldots, m, \\ & g_j(x) \geq 0, & j = 1, \ldots, r, \\ & x \in \Re^n, & \end{array}$$

how to revise the optimality conditions? We may rewrite the constraints as

$$-g_j(x) \leq 0, \quad j = 1, \ldots, r,$$

then use the result of this section. In fact now the first order necessary condition asks that

$$\nabla f(x^*) - \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) - \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) = 0$$

$$\mu_j^* g_j(x^*) = 0 \quad \text{for} \quad j = 1, \ldots, r.$$

Here the sign of the first sum can be written either as "+", or "-", because the signs of multipliers λ_i^* have no restriction. Or we can define the *Lagrangian function* as follows:

$$\bar{L}(x, \lambda, \mu) = f(x) - \sum_{i=1}^{m} \lambda_i h_i(x) - \sum_{j=1}^{r} \mu_j g_j(x),$$

then in the first order necessary condition, $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$ should be replaced by:

$$\nabla_x \bar{L}(x^*, \lambda^*, \mu^*) = 0,$$

and in the second order conditions, the matrix $\nabla_{xx}^2 L$ should be changed to $\nabla_{xx}^2 \bar{L}(x^*, \lambda^*, \mu^*)$.

Suppose a minimization problem has inequality constraints only, for example,

$$\begin{array}{ll} \min & f(x) \\ s.t. & g_1(x) \leq 0; \\ & g_2(x) \leq 0. \end{array}$$

Here the feasible region is

$$S = \{x \mid g_1(x) \leq 0;\ g_2(x) \leq 0\}.$$

If at the optimal solution x^*, there is no active constraint, i.e.,

$$g_1(x^*) < 0,\ g_2(x^*) < 0,$$

then by KKT condition, we know that

$$\mu_1^* g_1(x^*) = 0 \Longrightarrow \mu_1^* = 0,$$

$$\mu_2^* g_2(x^*) = 0 \Longrightarrow \mu_2^* = 0.$$

So,

$$\nabla f(x^*) + \sum_{j=1}^{2} \mu_j^* \nabla g_j(x^*) = 0 \Longrightarrow \nabla f(x^*) = 0.$$

This result is correct because in this case there is a neighborhood $N(x^*, \epsilon)$ of x^* such that

$$x \in N(x^*, \epsilon) \Longrightarrow g_1(x) < 0,\ g_2(x) < 0,$$

that is,

$$N(x^*, \epsilon) \subset S.$$

Since x^* is a minimum point of f over the feasible region S,

$$f(x) \geq f(x^*) \text{ for all } x \in N(x^*, \epsilon).$$

This means that x^* is an local **unconstrained** minimum point. Therefore,

$$\nabla f(x^*) = 0.$$

It tells us that for an inequality constrained minimization problem, if at the optimal solution x^*, $\nabla f(x^*) \neq 0$, then there must be some active constraints, i.e., x^* is on some boundary of the feasible region.

OR4030 OPTIMIZATION Chapter 7

Penalty and Barrier Methods

7.1 A Brief Introduction

Main Idea of the Methods:

1. Solve a constrained optimization problem by solving a sequence of unconstrained optimization problems, and in the limit, the solutions of unconstrained problems will converge to the solution of the constrained problem.
2. Use an auxiliary function that incorporates the objective function together with "penalty" terms that measure violations of the constraints.

Two groups of classical methods:

- Barrier methods: impose a penalty for reaching the boundary of an inequality constraint from the interior area (prevent the iterative points from being out of the boundary).
- Penalty methods: impose a penalty for violating a constraint (force the iterative points to return to the feasible region gradually).

Common idea of the two groups of methods:

Consider the constrained problem

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & x \in S, \end{aligned} \tag{1}$$

where S is the feasible region of the problem.

Define

$$\sigma(x) = \begin{cases} 0, & \text{if } x \in S \\ +\infty, & \text{if } x \notin S \end{cases}$$

Function σ is an infinite penalty for violating feasibility.

Problem (1) can be transformed equivalently to

$$\min \ f(x) + \sigma(x). \tag{2}$$

But it is not practical to solve problem (2), because the objective function is not defined outside S, and discontinuous on the boundary.

Barrier and penalty methods solve a sequence of unconstrained sub-problems that gradually approximate problem (2) in which $\sigma(x)$ is replaced by a continuous function that gradually approaches $\sigma(x)$.

Barrier method generates a sequence of iterates that converge to a solution of the constrained problem (1) from the interior of the feasible region. – interior penalty method

Penalty method generates a sequence of iterates that converge to a solution of the constrained problem (1) from the exterior of the feasible region. – exterior penalty method

7.2 Barrier Methods

Consider the nonlinear inequality constrained problem

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g_i(x) \geq 0, i = 1, \ldots, m \end{aligned} \tag{3}$$

1. Assumption

1. f and g_i are twice continuously differentiable.
2. The feasible region has a nonempty interior

$$S^0 = \{x | g_i(x) > 0, \quad i = 1, \ldots, m\},$$

i.e., there exists a point $\bar{x}$ such that

$$g_i(\bar{x}) > 0, \quad i = 1, \ldots, m.$$

3. Any point on the boundary can be approached by a sequence of interior points.

2. Barrier Terms and Barrier Functions

(1) We choose a continuous function ϕ defined on S^0 satisfying

$$\phi(x) \to \infty \quad \text{if any } g_i(x) \to 0_+.$$

Two examples of such a function ϕ are :

- Logarithmic function: $\phi(x) = -\sum_{i=1}^{m} \log(g_i(x))$.
- Inverse function: $\phi(x) = \sum_{i=1}^{m} \frac{1}{g_i(x)}$.

(2) Barrier terms:

For each feasible point x, the logarithmic function and the inverse function satisfy (as shown in Figure 7.1) that

$$\mu\phi(x) \text{ approaches } \sigma(x) \text{ as } \mu \to 0_+.$$

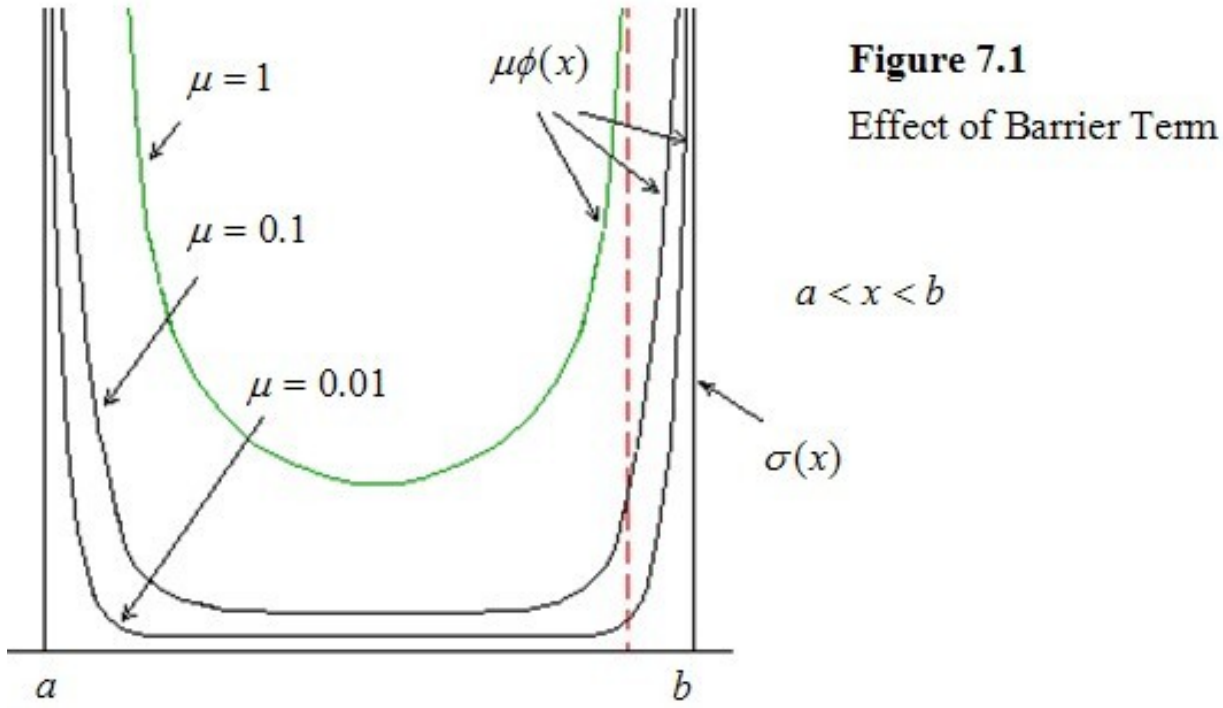


Figure 7.1
Effect of Barrier Term

We call
$\mu\phi(x)$: a barrier term;
μ : a barrier parameter.
Two frequently used barrier terms are

$$-\mu \sum_{i=1}^{m} \log(g_i(x)) \quad \text{and} \quad \mu \sum_{i=1}^{m} \frac{1}{g_i(x)}.$$

$\beta(x, \mu) = f(x) + \mu\phi(x)$: a barrier function.

- Logarithmic barrier function:

$$\beta(x, \mu) = f(x) - \mu \sum_{i=1}^{m} \log(g_i(x)).$$

- Inverse barrier function:

$$\beta(x, \mu) = f(x) + \mu \sum_{i=1}^{m} \frac{1}{g_i(x)}.$$

Barrier methods solve a sequence of "unconstrained" minimization problems

$$\min_{x \in S^0} \beta(x, \mu_k) \tag{4}$$

for a sequence of $\{\mu_k\}$ that decreases monotonically to zero.

The method transfers the constrained optimization problem (3) to problem (4). But (4) looks still a constrained optimization problem. What is the advantage to make this change?

Note that the region S^0 is an open set, that is, every point of S^0 is an interior point. So, if $\bar{x}$ is a minimum point of problem (4), it must be a unconstrained local minimum point of function $\beta(x, \mu_k)$, and hence

$$\nabla_x \beta(\bar{x}, \mu_k) = 0.$$

Therefore, we can use unconstrained minimization methods, such as the steepest descent method, or Newton's method, to solve this problem.

If we make a line search along a descent direction, as when the point is close to the boundary, the function value of $\beta(x, \mu_k)$ approaches $+\infty$, the minimum point along this direction would remain in S^0.

3. Why Don't Solve a Single Unconstrained Problem Using a Small Value of μ?

(1) No matter how small μ is, $\mu\phi(x)$ is different from $\sigma(x)$, and solving problem (4) cannot find a solution x_* of problem (1) if x_* is on the boundary.

(2) When μ is small, problem (4) is difficult to solve, especially if the initial point is far from the solution.

We need to start with an appropriate value of μ (not very small), and solve a sequence of problems (4) with decreasing μ,. The solution of problems (4) with $\mu = \mu_k$ is used as starting point for problems (4) with $\mu = \mu_{k+1}$ to facilitate computation.

Example 7.1 (Barrier Method) Consider the problem

$$\begin{array}{ll} \min & f(x) = x_1 - 2x_2 \\ s.t. & 1 + x_1 - x_2^2 \geq 0 \\ & x_2 \geq 0. \end{array}$$

The unconstrained problem by the logarithmic barrier function is:

$$\min_x \beta(x, \mu) = x_1 - 2x_2 - \mu \log(1 + x_1 - x_2^2) - \mu \log x_2.$$

For any fixed value $\mu > 0$, the first order necessary conditions for optimality are:

$$\begin{cases} 1 - \frac{\mu}{1+x_1-x_2^2} = 0, \\ -2 + \frac{2\mu x_2}{1+x_1-x_2^2} - \frac{\mu}{x_2} = 0. \end{cases}$$

$$\Longrightarrow \qquad -2 + 2x_2 - \frac{\mu}{x_2} = 0$$

$$\Longrightarrow \qquad x_2^2 - x_2 - \frac{1}{2}\mu = 0$$

$$\Longrightarrow \qquad x_2(\mu) = \frac{1 + \sqrt{1 + 2\mu}}{2}$$

(another solution $x_2 < 0$ and hence is discarded),

$$\Longrightarrow \qquad x_1(\mu) = \frac{\sqrt{1 + 2\mu} + 3\mu - 1}{2}$$

(note that $x_1 = x_2^2 - 1 + \mu$).

When $\mu \to 0$,

$$\lim_{\mu \to 0} x_1(\mu) = 0, \quad \lim_{\mu \to 0} x_2(\mu) = 1.$$

We may verify that the limit (0,1) is indeed the minimum point of the problem by the graph below.

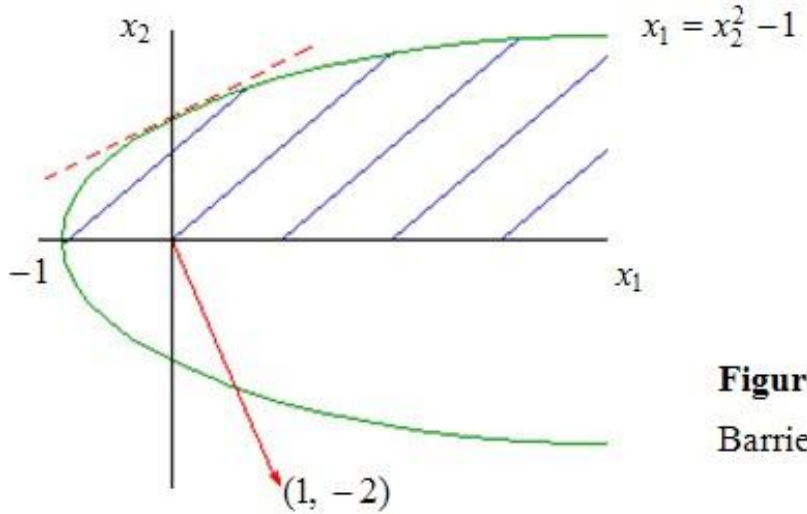


Figure (for Example 7.1)
Barrier Method

4 Features of Barrier Methods

(1) From the example, we see that when $\mu_k \to 0$, $x(\mu_k) \to x_*$ (the optimal solution). We shall prove this conclusion later.

(2) For different values of μ, the optimal solutions of problem (4) define a curve $x(\mu)$, called the barrier trajectory.

(3) If the logarithmic barrier function is used, the transformed equivalent problem is

$$\min_{x \in S^0} \beta(x, \mu) = f(x) - \mu \sum_{i=1}^{m} \log(g_i(x)).$$

The minimum point $x(\mu)$ satisfies

$$\nabla f(x) - \mu \sum_{i=1}^{m} \frac{\nabla g_i(x)}{g_i(x)} = 0.$$

If we define

$$\lambda_i(\mu) = \frac{\mu}{g_i(x)},$$

then $x = x(\mu)$ satisfies

$$\nabla f(x) - \sum_{i=1}^{m} \lambda_i(\mu) \nabla g_i(x) = 0.$$

So, all $x(\mu)$ and $\lambda(\mu)$ satisfy the following conditions:

$$g_i(x(\mu)) > 0, \quad i = 1, \ldots, m;$$

$$\nabla f(x(\mu)) - \sum_{i=1}^{m} \lambda_i(\mu) \nabla g_i(x(\mu)) = 0; \tag{5}$$

$$\lambda_i(\mu) g_i(x(\mu)) = \mu, \quad i = 1, \ldots, m; \tag{6}$$

$$\lambda_i(\mu) \geq 0, \quad i = 1, \ldots, m; \tag{7}$$

which resemble the first order necessary conditions for optimality, except that the RHS of (6) is μ, not 0.

When $\mu \to 0$, suppose $x(\mu) \to x_*$ and $\lambda(\mu) \to \lambda_*$ $= (\lambda_{*1}, \ldots, \lambda_{*m})^T$. Then from (5)-(7), we have

$$
\begin{aligned}
& g_i(x_*) \geq 0, \quad i = 1, \ldots, m; \\
& \nabla f(x_*) - \sum_{i=1}^{m} \lambda_{*i} \nabla g_i(x_*) = 0; \\
& \lambda_{*i} g_i(x_*) = 0, \quad i = 1, \ldots, m; \\
& \lambda_{*i} \geq 0, \quad i = 1, \ldots, m,
\end{aligned}
$$

i.e., $\lim_{\mu \to 0} \lambda(\mu)$ is the KKT multiplier of the problem (3).

Conclusion. The minimum points $x(\mu)$ of the barrier method provide estimates

$$\lambda_i(\mu) = \frac{\mu}{g_i(x(\mu))}$$

for the KKT multipliers λ_* at the optimal solution of problem (3). When $\mu \to 0$, the estimate $\lambda(\mu)$ approaches the exact λ_*.

Example 7.2 (KKT Multiplier Estimates) Consider the problem

$$\begin{aligned} \min \quad & f(x) = x_1^2 + x_2^2 \\ s.t. \quad & g_1(x) = x_1 - 1 \geq 0 \\ & g_2(x) = x_2 + 1 \geq 0. \end{aligned}$$

It is easy to verify that the minimum point is $x_* = (1, 0)^T$.

KKT multipliers:

The second constraint is inactive $\Longrightarrow \lambda_{*2} = 0$. So,

$$\nabla f(x_*) - \lambda_{*1} \nabla g_1(x_*) = 0 \Longrightarrow \begin{bmatrix} 2 \\ 0 \end{bmatrix} = \lambda_{*1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Longrightarrow \lambda_{*1} = 2,$$

i.e.,

$$\lambda_* = (2, 0)^T.$$

Now suppose the problem is solved by the logarithmic barrier method.

$$\min_{x \in S^0} \beta(x, \mu) = x_1^2 + x_2^2 - \mu \log(x_1 - 1) - \mu \log(x_2 + 1).$$

$$\frac{\partial \beta}{\partial x_1} = 0 \Longrightarrow 2x_1 - \frac{\mu}{x_1 - 1} = 0 \Longrightarrow 2x_1^2 - 2x_1 - \mu = 0,$$

$$\frac{\partial \beta}{\partial x_2} = 0 \Longrightarrow 2x_2 - \frac{\mu}{x_2 + 1} = 0 \Longrightarrow 2x_2^2 + 2x_2 - \mu = 0,$$

yielding

$$x_1(\mu) = \frac{1 + \sqrt{1 + 2\mu}}{2}, \quad x_2(\mu) = \frac{-1 + \sqrt{1 + 2\mu}}{2}.$$

The KKT multiplier estimates are

$$\lambda_1(\mu) = \frac{\mu}{g_1(x(\mu))} = \frac{2\mu}{\sqrt{1+2\mu}-1} = \sqrt{1+2\mu}+1;$$

$$\lambda_2(\mu) = \frac{\mu}{g_2(x(\mu))} = \frac{2\mu}{\sqrt{1+2\mu}+1} = \sqrt{1+2\mu}-1.$$

When $\mu \to 0$,

$$x_1(\mu) \to 1 \quad x_2(\mu) \to 0;$$

$$\lambda_1(\mu) \to 2 \quad \lambda_2(\mu) \to 0.$$

We see that when $\mu \to 0$,

$$x(\mu) \to x_* \quad \text{and} \quad \lambda(\mu) \to \lambda_*.$$

7.3 Penalty Methods

7.3.1. Penalty Methods for Equality Constrained Problems

Consider the equality constrained problem

$$\begin{array}{ll} \min & f(x) \\ s.t. & g(x) = (g_1(x), \ldots, g_m(x))^T = 0. \end{array} \tag{8}$$

Penalty terms and penalty function:

(1) A continuous function ψ with the following property:

$$\begin{cases} \psi(x) = 0, & \text{if } x \text{ is feasible;} \\ \psi(x) > 0, & \text{otherwise,} \end{cases} \tag{9}$$

can play a role of penalty for constraint violation.

For example, we can use the **quadratic-loss function**:

$$\psi(x) = \frac{1}{2}\sum_{i=1}^{m} g_i^2(x) \quad (= \frac{1}{2} g(x)^T g(x)),$$

or more generally,

$$\psi(x) = \frac{1}{\gamma} \sum_{i=1}^{m} |g_i(x)|^\gamma \quad (\gamma > 1).$$

(2) We need to introduce a parameter ρ (> 0) to control the weight of the penalty. ρ is called the penalty parameter.

- When $\rho \to \infty$, $\rho\psi(x) \to \sigma(x)$.
- As ρ increases,
  $\Longrightarrow$ the penalty is increased
  $\Longrightarrow$ the iterates are forced to move towards the feasible region.

(3) We call

- $\rho\psi(x)$: penalty term;
- $\pi(x, \rho) = f(x) + \rho\psi(x)$: **penalty function**.

How does the penalty method work?

For an increasing sequence $\{\rho_k\}$ of positive values tending to ∞, solve the unconstrained minimization problems

$$\min_{x \in R^n} \pi(x, \rho_k)$$

to obtain $\{x^k\}$. x^k shall approach the optimal solution x_* of problem (8).

How to approximate Lagrange multiplier by penalty method?

Suppose that we choose the quadratic-loss function as the penalty term, and let $x(\rho)$ be the minimum point of

$$\pi(x, \rho) = f(x) + \frac{1}{2}\rho \sum_{i=1}^{m} g_i(x)^2.$$

Then $x(\rho)$ satisfies

$$\nabla_x \pi(x(\rho), \rho) = \nabla f(x(\rho)) + \rho \sum_{i=1}^{m} g_i(x(\rho)) \nabla g_i(x(\rho)) = 0.$$

Define

$$\lambda_i(\rho) = \rho g_i(x(\rho)).$$

Then

$$\nabla f(x(\rho)) + \sum_{i=1}^{m} \lambda_i(\rho) \nabla g_i(x(\rho)) = 0.$$

When $\rho \to \infty$, suppose $x(\rho) \to x_*$ and $\lambda(\rho) \to \lambda_*$, then we see that x_* and λ_* satisfy

$$\nabla f(x_*) + \sum_{i=1}^{m} \lambda_{*i} \nabla g_i(x_*) = 0.$$

Hence the limit λ_* is the Lagrange multiplier vector of problem (8).

This mean that vector $\lambda(\rho)$ **can be used to estimate the Lagrange multiplier vector** λ_*.

For large values of ρ, function $\pi(x, \rho)$ is difficult to be minimized. Therefore, we need to

- minimize a sequence of functions $\pi(x, \rho_k)$ with increasing ρ_k which tend to ∞, and
- use the minimizer x^k of the function $\pi(x, \rho_k)$ as the initial point in minimizing function $\pi(x, \rho_{k+1})$.

Example 7.3 (Penalty Method). Consider the problem

$$\begin{aligned} \min \quad & f(x) = -x_1 x_2 \\ s.t. \quad & g(x) = x_1 + 2x_2 - 4 = 0. \end{aligned}$$

Use quadratic-loss penalty function,

$$\min_{x \in R^2} \pi(x, \rho) = -x_1 x_2 + \frac{1}{2}\rho(x_1 + 2x_2 - 4)^2.$$

$$\frac{\partial \pi}{\partial x_1} = 0 \Longrightarrow -x_2 + \rho(x_1 + 2x_2 - 4) = 0;$$

$$\frac{\partial \pi}{\partial x_2} = 0 \Longrightarrow -x_1 + 2\rho(x_1 + 2x_2 - 4) = 0.$$

For $\rho > \frac{1}{4}$, the above equations have the solution

$$x_1(\rho) = \frac{8\rho}{4\rho - 1}, \quad x_2(\rho) = \frac{4\rho}{4\rho - 1}.$$

Hence

$$\begin{aligned} g(x(\rho)) &= x_1(\rho) + 2x_2(\rho) - 4 \\ &= \frac{16\rho}{4\rho - 1} - 4 = \frac{4}{4\rho - 1} \quad (> 0). \end{aligned}$$

Note that $x(\rho)$ is not a feasible point of the original constrained problem. Let

$$\lambda(\rho) = \rho \; g(x(\rho)) = \frac{4\rho}{4\rho - 1}.$$

When $\rho \to \infty$,

$$x_1(\rho) \to 2, \quad x_2(\rho) \to 1, \quad \lambda(\rho) \to 1.$$

It can be verified easily that:

$x_* = (2, 1)^T$ is indeed the minimum point, and
$\lambda_* = 1$ is indeed the Lagrange multiplier
to the constrained minimization problem.

7.3.2. Penalty Methods for Inequality Constrained Problems

Penalty method is also available for inequality constrained problem

$$\begin{array}{ll} \min & f(x) \\ s.t. & g_i(x) \geq 0, \quad i = 1, \ldots, m. \end{array} \tag{10}$$

Again, any continuous function ψ with the following property:

$$\begin{cases} \psi(x) = 0, & \text{if } x \text{ is feasible;} \\ \psi(x) > 0, & \text{otherwise,} \end{cases} \tag{11}$$

can play a role of penalty for constraint violation. In particular, the **quadratic-loss function** in this case is

$$\psi(x) = \frac{1}{2} \sum_{i=1}^{m} [\min(g_i(x), 0)]^2.$$

Note that

- When x is feasible, all $g_i(x) \geq 0$
  $\Longrightarrow \psi(x) = 0 \quad \sim$ no penalty.
- Otherwise, at least for one constraint function, say $g_j(x)$, the constraint is violated: $g_j(x) < 0 \Longrightarrow \psi(x) \geq \frac{1}{2} g_j^2(x) > 0 \quad \sim$ a penalty is imposed.

The penalty method is the same as before: for a sequence of ρ_k ($\rho_k \nearrow \infty$), solve unconstrained optimization problems

$$\min_{x \in R^n} \pi(x, \rho_k) = f(x) + \rho_k \psi(x).$$

When we minimize this penalty function, we want to know: what is $\nabla \psi(x)$?

It can be verified that, for the above quadratic-loss function,

$$\nabla\psi(x) = \sum_{i=1}^{m} \min(g_i(x), 0) \cdot \nabla g_i(x) \tag{12}$$

(see the textbook for proof).

Finally, if a problem contains both equality and inequality constraints, say

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & h_j(x) = 0, \quad j = 1, \ldots, n, \qquad (13) \\ & g_i(x) \geq 0, \quad i = 1, \ldots, m. \qquad (14) \end{aligned}$$

then the penalty function can be

$$\pi(x, \rho) = f(x) + \frac{\rho}{2}\{\sum_{j=1}^{n} h_j^2(x) + \sum_{i=1}^{m} [\min(g_i(x), 0)]^2\}.$$

7.4 Convergence of the Methods

Here we consider only the barrier methods (penalty methods can be analyzed in a similar way, see the textbook) applied to the following problem:

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & g_i(x) \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{15}$$

Let S and S^0 denote, respectively, the feasible region and its interior, i.e.,

$$S = \{x | g_i(x) \geq 0, \ i = 1, \ldots, m\},$$

$$S^0 = \{x | g_i(x) > 0, \ i = 1, \ldots, m\}.$$

Assumptions:

1. f and g_i $(i = 1, \ldots, m)$ are continuous, ϕ is a continuous function on S^0, and $\phi(x) \to +\infty$ when x approaches the boundary of S.
2. S^0 is nonempty.
3. Any y on the boundary of S can be approached by a sequence $\{x_k\}$ in S^0: $x_k \to y,\ x_k \in S^0$.

Theorem 7.1 (Convergence of the Barrier Method)
Let

$$\beta(x,\mu) = f(x) + \mu\phi(x); \quad \mu_1 \geq \mu_2 \geq \cdots ; \quad \lim_{k\to\infty} \mu_k = 0,$$

and x_k be a global minimum point of problem $\min_{x\in S^0} \beta(x,\mu_k)$. Then for $k = 1,\ldots,$

(a) $f(x_{k+1}) \leq f(x_k)$;

(b) $\phi(x_{k+1}) \geq \phi(x_k)$;

(c) if a subsequence $\{x_k | k \in K\}$ converges to $\hat{x}$, then $\hat{x}$ must be a global solution to problem (15).

Proof

(a) For each k,

$$x_k \text{ is the solution of } \min_{x \in S^0} \beta(x, \mu_k). \tag{16}$$

So,

$$\begin{aligned} \beta(x_k, \mu_k) &\leq \beta(x_{k+1}, \mu_k) \\ \Longrightarrow f(x_k) + \mu_k \phi(x_k) &\leq f(x_{k+1}) + \mu_k \phi(x_{k+1}). \end{aligned} \tag{17}$$

$$\begin{aligned} \beta(x_{k+1}, \mu_{k+1}) &\leq \beta(x_k, \mu_{k+1}) \\ \Longrightarrow f(x_{k+1}) + \mu_{k+1} \phi(x_{k+1}) &\leq f(x_k) + \mu_{k+1} \phi(x_k). \end{aligned} \tag{18}$$

$$
\begin{aligned}
& \mu_{k+1} \times (17) + \mu_k \times (18) \\
\Longrightarrow \quad & \mu_{k+1} f(x_k) + \mu_k f(x_{k+1}) \leq \mu_{k+1} f(x_{k+1}) + \mu_k f(x_k) \\
\Longrightarrow \quad & (\mu_k - \mu_{k+1}) f(x_{k+1}) \leq (\mu_k - \mu_{k+1}) f(x_k) \\
\Longrightarrow \quad & f(x_{k+1}) \leq f(x_k) \ \ (\text{as } \mu_k - \mu_{k+1} > 0). \qquad (19)
\end{aligned}
$$

(b) From (17) and (19)

$$
\begin{aligned}
\Longrightarrow \quad & \mu_k \phi(x_k) \leq \mu_k \phi(x_{k+1}) \\
\Longrightarrow \quad & \phi(x_k) \leq \phi(x_{k+1}).
\end{aligned}
$$

(c) Let $x_k \xrightarrow{K} \hat{x}$ (i.e., there is a subsequence of x_k which converges to $\hat{x}$). We need to prove that $\hat{x}$ must be an optimal solution of problem (15).

(i) First, since $x_k \in S^0$, $g_i(x_k) > 0$ for each k and i. Then, when $k \xrightarrow{K} \infty$,

$$g_i(\hat{x}) \geq 0, \ \forall i = 1, \ldots, m.$$

$\Longrightarrow \hat{x} \in S$, i.e. $\hat{x}$ is a feasible solution of problem (15)

(ii) Let x_* be a global minimum point of problem (15). We need to show that $f(\hat{x}) = f(x_*)$. By assumption (3) and the continuity of f, for **any** $\epsilon > 0$, there exists $x_\epsilon \in S^0$ such that

$$f(x_\epsilon) < f(x_*) + \epsilon.$$

Due to (16),

$$\beta(x_k, \mu_k) \leq \beta(x_\epsilon, \mu_k),$$

i.e.,

$$f(x_k) + \mu_k \phi(x_k) \leq f(x_\epsilon) + \mu_k \phi(x_\epsilon) < f(x_*) + \epsilon + \mu_k \phi(x_\epsilon). \quad (20)$$

We consider two cases:

Case A. $\hat{x} \in S^0$. Then $\phi(\hat{x})$ is a finite number, and

$$\phi(x_k) \xrightarrow{K} \phi(\hat{x}).$$

Let $k \xrightarrow{K} \infty$, from (20),

$$f(\hat{x}) + 0 \cdot \phi(\hat{x}) \leq f(x_*) + \epsilon + 0 \cdot \phi(x_\epsilon),$$

i.e.,

$$f(\hat{x}) \leq f(x_*) + \epsilon. \tag{21}$$

Case B. $\hat{x} \notin S^0$, i.e., $\hat{x}$ is on the boundary of S. Then $\phi(\hat{x}) = +\infty$. So, for large $k \in K$, $\phi(x_k) > 0$. We have from (20) that

$$f(x_k) \leq f(x_k) + \mu_k \phi(x_k) < f(x_*) + \epsilon + \mu_k \phi(x_\epsilon), \quad \text{for large } k \in K.$$

Let $k \xrightarrow{K} \infty$, from the above inequalities we have

$$f(\hat{x}) \leq f(x_*) + \epsilon. \tag{22}$$

(21) and (22) are the same result. As the two inequalities are true for any $\epsilon > 0$, they just mean that

$$\begin{aligned} f(\hat{x}) \leq f(x_*) \quad &\Longrightarrow \quad f(\hat{x}) = f(x_*) \\ &\Longrightarrow \quad \hat{x} \text{ is a global minimum point of problem (15).} \end{aligned}$$

So, the proof of part (c) is completed.

7.5 Augmented Lagrangian Method

In the penalty method, the exact solution cannot be found unless the parameter $\rho \to \infty$. But when ρ is very big, the penalty function becomes ill conditioned. Are there some methods that can obtain the optimal solution without requiring the parameter $\rho \to \infty$? Yes, exact penalty method and augmented Lagrangian method are two this kind of methods. Here we introduce only the augmented Lagrangian method.

Consider the problem

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & h_i(x) = 0, \quad i = 1, \ldots, m \\ & x \in R^n. \end{aligned} \tag{23}$$

Let x^* be an optimal solution with associated multiplier λ^*. Obviously x^* is also an optimal solution to the problem

$$\begin{array}{ll} \min & L(x,\lambda) = f(x) + \lambda^T h(x) \\ \text{s.t.} & h(x) = 0 \end{array} \tag{24}$$

where $h(x) = (h_1(x), \ldots, h_m(x))^T$, because when $h(x) = 0$, the function $L(x,\lambda) = f(x)$. We now use the penalty function method to solve problem (24), that is, we consider the unconstrained optimization problem

$$\min_x \mathcal{A}(x,\lambda,\rho) = f(x) + \lambda^T h(x) + \frac{1}{2}\rho h(x)^T h(x).$$

We call function $\mathcal{A}(x, \lambda, \rho)$ **an augmented Lagrangian function** because it adds a penalty term to the Lagrangian function. The method to find x^* by using penalty method on the augmented Lagrangian function is called **augmented Lagrangian method**.

Algorithm (Augmented Lagrangian Method)

Step 0. Give an initial guess (x^0, λ^0), and choose an initial penalty parameter value $\rho_0 > 0$. Set $k = 0$.

Step 1. Optimal Test: if $\nabla L(x^k, \lambda^k) = 0$, then output $(x^*, \lambda^*) = (x^k, \lambda^k)$ and stop.

Step 2. Solve the unconstrained optimization problem

$$\min_x \mathcal{A}(x, \lambda^k, \rho_k) = f(x) + (\lambda^k)^T h(x) + \frac{1}{2}\rho_k h(x)^T h(x)$$

using any of unconstrained optimization methods, and let the optimal solution be x^{k+1}.

Step 3. Update λ^k by formula

$$\lambda^{k+1} = \lambda^k + \rho_k h(x^{k+1}) \tag{25}$$

and choose $\rho_{k+1} \geq \rho_k$.

Step 4. Set $k := k + 1$ and return to Step 1.

Note that in Step 1, $\nabla L = (\nabla_x L, \nabla_\lambda L)$ as we explained in Chapter 6, where

$$\nabla_x L(x, \lambda) = \nabla f(x) + \nabla h(x)\lambda,$$

and

$$\nabla_\lambda L(x, \lambda) = h(x).$$

The reason to use formula (25) to update the multiplier vector λ is as follows.

If x^{k+1} minimizes $\mathcal{A}(x, \lambda^k, \rho_k)$, then

$$\nabla_x \mathcal{A}(x^{k+1}, \lambda^k, \rho_k) = 0,$$

or

$$\nabla f(x^{k+1}) + \nabla h(x^{k+1})\lambda^k + \rho_k \nabla h(x^{k+1}) h(x^{k+1}) = 0.$$

This can be rearranged as

$$\nabla f(x^{k+1}) + \nabla h(x^{k+1})[\lambda^k + \rho_k h(x^{k+1})] = 0.$$

If we use formula (25) to obtain λ^{k+1}, then

$$\nabla_x L(x^{k+1}, \lambda^{k+1}) = \nabla f(x^{k+1}) + \nabla h(x^{k+1})\lambda^{k+1} = 0,$$

that is, **the first order necessary conditions for optimality are always partially satisfied.**

The optimality test in Step 1 in fact only need to check if

$$\nabla_\lambda L(x^{k+1}, \lambda^{k+1}) = h(x^{k+1}) = 0?$$

i.e., if x^{k+1} is a feasible point? If yes, then we can finish computation.

Note that in the algorithm, **the penalty parameter ρ is unnecessary to go to ∞.**

Function $\mathcal{A}$ is more complicated than the penalty function $f + \frac{1}{2}\rho h^T h$. Then **what is the advantage if we solve problem (24) instead of problem (23)?** The main reason is that we can prove, under certain conditions, that for the augmented Lagrangian method, there exists a constant $M > 0$ such that

$$\|\lambda^{k+1} - \lambda^*\| \leq \frac{M}{\rho_k}\|\lambda^k - \lambda^*\|, \tag{26}$$

$$\|x^{k+1} - x^*\| \leq \frac{M}{\rho_k}\|\lambda^k - \lambda^*\|. \tag{27}$$

Remember that here x^* is the optimal solution to the original constrained minimization problem (23), and λ^* is the associated Lagrange multiplier. Therefore, if $\rho_k > M$, for example let all $\rho_k = \hat{\rho}$ and the constant $\hat{\rho} > M$, then from (26) we know that $\{\lambda^k\}$ converges to λ^* linearly. Then from (27) we see that $x_k \to x^*$.

Therefore, when we use the augmented Lagrangian method, we may obtain the solution x^* and its corresponding multiplier vector λ^* without requiring the penalty parameter ρ_k to increase to infinity. This is the main advantage of the method. But here we only assure the existence of constant M, not its exact value. Hence we do not know what value of ρ_k is already large enough for the convergence.

We now give an example.

Example 7.6 (Augmented Lagrangian Method) Consider the problem

$$\begin{aligned} \textit{minimize} \quad & f(x_1, x_2) = \frac{1}{2}x_1^2 + \frac{1}{6}x_2^2 \\ \textit{subject to} \quad & h(x_1, x_2) = x_1 + x_2 - 1 = 0. \end{aligned}$$

Form the augmented Lagrange function

$$\mathcal{A}(x, \lambda_k, \rho) = \frac{1}{2}x_1^2 + \frac{1}{6}x_2^2 + \lambda_k(x_1 + x_2 - 1) + \frac{1}{2}\rho(x_1 + x_2 - 1)^2.$$

Let the minimum solution to $\mathcal{A}(x, \lambda_k, \rho)$ be $x^{k+1} = (x_1^{k+1}, x_2^{k+1})$. To obtain this minimum solution, we solve the equations $\nabla_x \mathcal{A} = 0$, i.e.,

$$\begin{cases} x_1 + \lambda_k + \rho(x_1 + x_2 - 1) = 0, \\ \frac{1}{3}x_2 + \lambda_k + \rho(x_1 + x_2 - 1) = 0. \end{cases}$$

From the above equations, it is easy to see that $x_2 = 3x_1$. Substituting it into the first equation, we obtain that

$$x_1^{k+1} = \frac{\rho - \lambda_k}{1 + 4\rho}, \tag{28}$$

$$x_2^{k+1} = \frac{3(\rho - \lambda_k)}{1 + 4\rho}. \tag{29}$$

The updated multiplier is

$$\begin{aligned} \lambda_{k+1} &= \lambda_k + \rho h(x^{k+1}) \\ &= \lambda_k + \rho(x_1^{k+1} + x_2^{k+1} - 1) \\ &= \lambda_k + \rho(\frac{4(\rho - \lambda_k)}{1 + 4\rho} - 1) \\ &= \frac{\lambda_k - \rho}{1 + 4\rho}. \end{aligned} \tag{30}$$

We now show that the sequence $\{\lambda_k\}$ has a limit $-\frac{1}{4}$.

It is easy to see that for any $\rho > 0$, the sequence $\{\lambda_k\}$ is decreasing because

$$\lambda_{k+1} < \frac{\lambda_k}{1 + 4\rho} < \lambda_k.$$

Also, if we choose the initial $\lambda_0 \geq -\frac{1}{4}$, then all $\lambda_k \geq -\frac{1}{4}$ because when $\lambda_k \geq -\frac{1}{4}$, by (30),

$$\lambda_{k+1} \geq \frac{-\frac{1}{4} - \rho}{1 + 4\rho} = -\frac{1}{4}.$$

So, the sequence $\{\lambda_k\}$ has a limit λ^*. Taking limits on both sides of (30), we obtain

$$\lambda^* = \frac{\lambda^* - \rho}{1 + 4\rho},$$

from which we see that

$$\lambda^* = -\frac{1}{4}.$$

Now taking limits in (28) and (29), we obtain

$$x_1^{k+1} \to \frac{\rho + \frac{1}{4}}{1 + 4\rho} = \frac{1}{4} = x_1^*;$$

$$x_2^{k+1} = 3x_1^{k+1} \to \frac{3}{4} = x_2^*.$$

It is easy to verify that $x^* = (\frac{1}{4}, \frac{3}{4})$ is indeed the optimal solution of the problem with a multiplier $\lambda^* = -\frac{1}{4}$. Note that in order to obtain the optimal solution, the parameter ρ can be **any positive number, and we do not require ρ to approach ∞**. This is the main advantage of the augmented Lagrangian method against the penalty method.