LECTURE NOTE II

Chapter 3

Function of Several Variables

• Unconstrained multivariable minimization problem:

$$\min_x f(x), \quad x \in R^N$$

where *x* is a vector of *design variables* of dimension *N*, and *f* is a scalar *objective function*.

- Gradient of *f*: $\nabla f = \left[\frac{\partial f}{\partial x_1} \; \frac{\partial f}{\partial x_2} \; \frac{\partial f}{\partial x_3} \cdots \frac{\partial f}{\partial x_N} \right]^T$

- Possible locations of local optima
  - points where the gradient of *f* is zero
  - boundary points only if the feasible region is defined
  - points where *f* is discontinuous
  - points where the gradient of *f* is discontinuous or does not exist
- Assumption for the development of optimality criteria

  f and its derivatives exist and are continuous everywhere

3.1 Optimality Criteria

- Optimality criteria are necessary to recognize the solution.
- Optimality criteria provide motivation for most of useful methods.
- Taylor series expansion of *f*

$$f(x) = f(\overline{x}) + \nabla f(\overline{x})^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(\overline{x}) \Delta x + O_3(\Delta x)$$

where $\overline{x}$ is the current expansion point,

$\Delta x = x - \overline{x}$ is the change in *x*,

$\nabla^2 f(\overline{x})$ is the *N*x*N* symmetric Hessian matrix at $\overline{x}$,

$O_3(\Delta x)$ is the error of 2nd-order expansion.

$$\nabla^2 f(\overline{x}) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right]$$

- In order for $\overline{x}$ to be local minimum

$$\Delta f = f(x) - f(\overline{x}) \geq 0 \;\text{ for }\; \|x - \overline{x}\| \leq \delta \;\; (\delta > 0)$$

- In order for $\overline{x}$ to be *strict* local minimum

$$\Delta f = f(x) - f(\overline{x}) > 0 \;\text{ for }\; \|x - \overline{x}\| \leq \delta \;\; (\delta > 0)$$

- Optimality criterion (strict)

$$\Delta f = f(x) - f(\bar{x}) \approx \nabla f(\bar{x})^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(\bar{x}) \Delta x > 0, \ \forall \|\Delta x\| < \delta$$

➔ $\nabla f(\bar{x}) = 0 \text{ and } \nabla^2 f(\bar{x}) > 0$ (positive definite)

- For $Q(z) = z^T A z$

A is *positive definite* if $Q(z) > 0, \ \forall z \neq 0$

A is *positive semidefinite* if $Q(z) \geq 0, \ \forall z$ and $\exists z \neq 0 \ \ni z^T A z = 0$

A is *negative definite* if $Q(z) < 0, \ \forall z \neq 0$

A is *negative semidefinite* if $Q(z) \leq 0, \ \forall z$ and $\exists z \neq 0 \ \ni z^T A z = 0$

A is *indefinite* if $Q(z) > 0$ for some z and $Q(z) < 0$ for other z

- Test for positive definite matrices
  1. If any one of diagonal elements is not positive, then *A* is not p.d.
  2. All the leading principal determinants must be positive.
  3. All eigenvalues of *A* are positive.
- Test for negative definite matrices
  1. If any one of diagonal elements is not negative, then *A* is not n.d.
  2. All the leading principal determinant must have alternate sign starting from $D_1<0$ ($D_2>0$, $D_3<0$, $D_4>0$, …).
  3. All eigenvalues of *A* are negative.
- Test for positive semidefinite matrices
  1. If any one of diagonal elements is nonnegative, then *A* is not p.s.d.
  2. All the principal determinants are nonnegative.
- Test for negative semidefinite matrices
  1. If any one of diagonal elements is nonpositive, then *A* is not n.s.d.
  2. All the *k*-th order principal determinants are nonpositive if *k* is odd, and nonnegative if *k* is even.

Remark 1: The *principal minor* of order *k* of *N*x*N* matrix *Q* is a submatrix of size *k*x*k* obtained by deleting any *n-k* rows and their corresponding columns from the matrix *Q*.

Remark 2: The *leading principal minor* of order *k* of *N*x*N* matrix *Q* is a submatrix of size *k*x*k* obtained by deleting the ***last*** *n-k* rows and their corresponding columns.

Remark 3: The determinant of a principal minor is called the *principal determinant*. For *N*x*N* matrix, there are $2^N - 1$ principal determinant in all.

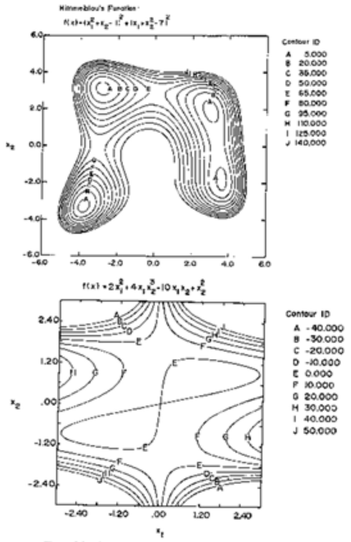


Figure 3.3. Two-variable nonlinear function of Example 3.1

- The stationary point $\bar{x}$ is a

  minimum if $\nabla^2 f(\bar{x})$ is positive definite,

  maximum if $\nabla^2 f(\bar{x})$ is negative definite,

  saddle point if $\nabla^2 f(\bar{x})$ is indefinite.

- **Theorem 3.1 Necessary condition for a local minimum**

  For x^* to be local minimum of $f(x)$, it is necessary that

  $$\nabla f(x^*) = 0 \text{ and } \nabla^2 f(x^*) \geq 0$$

- **Theorem 3.2 Sufficient condition for strict local minimum**

  If $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*) > 0$,

  then x^* to be strict or isolated local minimum of $f(x)$.

  Remark 1: The reverse of Theorem 3.1 is not true. (e.g., $f(x)=x^3$ at $x=0$)

  Remark 2: The reverse of Theorem 3.2 is not true. (e.g., $f(x)=x^4$ at $x=0$)

3.2 Direct Search Methods

- Direct search methods use only function values.
- For the cases where ∇f is not available or may not exist.

• *Modified simplex search method* (Nelder and Mead)

- In n dimensions, a *regular simplex* is a polyhedron composed of n+1 equidistant points which form its vertices. (for 2-d equilateral triangle, for 3-d tetrahedron)
- Let $x_i = (x_{i1}, x_{i2}, \cdots, x_{in})$ $(i = 1, 2, \cdots, n+1)$ be the i-th vector point in R^n of the simples vertices on each step of the search.

  Define $f(x_h) = \max\{f(x_i); i = 1, \cdots, n+1\}$,

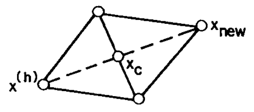
  $f(x_g) = \max\{f(x_{i \neq h}); i = 1, \cdots, n+1\}$ and

  $f(x_l) = \min\{f(x_i); i = 1, \cdots, n+1\}$.

  Select an initial simplex with termination criteria. (M=0)

  i) Decide x_h, x_g, x_l among (n+1) points in simplex vertices and let x_c be the centroid of all vertices excluding the worst point x_h.

  $$x_c = \frac{1}{n}\left\{\sum_{j=1}^{n+1} x_i - x_h\right\}$$



(a) Normal Reflection ($\theta = \alpha = 1$)

$f^{(l)} < f(x_{new}) < f^{(g)}$

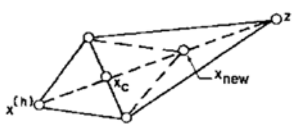
  ii) Calculate $f(x_h)$, $f(x_l)$, and $f(x_g)$. If x_l is same as previous one, then let M=M+1. If M>1.65n+0.05n^2, then M=0 and go to vi).

  iii) *Reflection*: $x_r = x_c + \alpha(x_c - x_h)$ (usually α =1)

  If $f(x_l) \leq f(x_r) \leq f(x_g)$, then set $x_h = x_r$ and go to i).

  iv) *Expansion*: If $f(x_r) < f(x_l)$, $x_e = x_c + \gamma(x_r - x_c)$.

  $(2.8 \leq \gamma \leq 3.0)$ If $f(x_e) \leq f(x_r)$, then set $x_h = x_e$ and go to i).



(b) Expansion ($\theta = \gamma > 1$)

$f(x_{new}) < f^{(l)}$

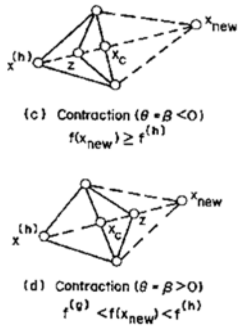
v) *Contraction*: If $f(x_r) \geq f(x_h)$, $x_t = x_c + \beta(x_h - x_c)$.

$(0.4 \leq \beta \leq 0.6)$ Else if $f(x_r) > f(x_g)$, $x_t = x_c - \beta(x_h - x_c)$.

Then set $x_h = x_t$ and go to i).

vi) If the simplex is small enough, then stop. Otherwise,

Reduction: $x_i = x_l + 0.5(x_i - x_l)$ for $i = 1, 2, \cdots, n+1$. And go to i).



Remark 1: The indices *h* and *l* have the one of value of *i*.

Remark 2: The termination criteria can be that the longest segment between points is small enough and the largest difference between function values is small enough.

Remark 3: If the contour of the objective function is severely distorted and elongated, the search can very inefficient and fail to converge.

• *Hooke-Jeeves Pattern Search*

- It consists of exploratory moves and pattern moves.

Select an initial guess $x^{(0)}$, increment vectors Δ_i for $i = 1, 2, \cdots, n$ and termination criteria. Start with *k*=1.

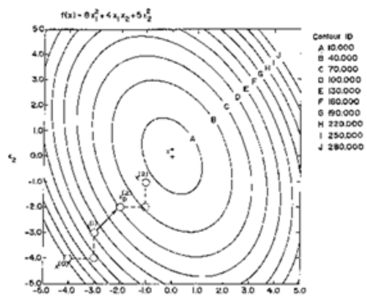


Figure 3.7. Pattern search iterations for Example 3.2.

i) *Exploratory search*:

A. Let *i*=1 and $x_b^{(k)} = x^{(k-1)}$.

B. Try $x_n^{(k)} = x_b^{(k)} + \Delta_i$. If $f(x_n^{(k)}) < f(x_b^{(k)})$, then $x_b^{(k)} = x_n^{(k)}$.

C. Else, try $x_n^{(k)} = x_b^{(k)} - \Delta_i$. If $f(x_n^{(k)}) < f(x_b^{(k)})$, then $x_b^{(k)} = x_n^{(k)}$.

D. Else, let $i = i + 1$ and go to B until $i > n$.

ii) If exploratory search fails ($x_b^{(k)} = x^{(k-1)}$)

A. If $\|\Delta_i\| < \varepsilon_i$ for $i = 1, 2, \cdots, n$, then $x^* = x^{(k-1)}$ and stop.

B. Else, $\Delta_i = 0.5\Delta_i$ for $i = 1, 2, \cdots, n$ and go to i).

iii) *Pattern search*:

A. Let $x_p^{(k+1)} = x_b^{(k)} + (x_b^{(k)} - x_b^{(k-1)})$

B. If $f(x_p^{(k+1)}) < f(x_b^{(k)})$, then $x^{(k)} = x_p^{(k)}$ and go to i).

C. Else, $x^{(k)} = x_b^{(k)}$ and go to i).

Remark 1: HJ method may be terminated prematurely in the presence of severe nonlinearity and will degenerate to a sequence of exploratory moves.

Remark 2: For the efficiency, the pattern search can be modified to perform a *line search* in the pattern search direction.

Remark 3: The Rosenblock's *rotating direction method* will rotate the exploratory search direction based on the previous moves using *Gram-Schmidt orthogonalization*.

- Let $\xi_1, \xi_2, \cdots, \xi_n$ be the initial search direction.

- Let α_i be the net distance moved in ξ_i direction. And

$$u_1 = \alpha_1\xi_1 + \alpha_2\xi_2 + \cdots + \alpha_n\xi_n$$
$$u_2 = \alpha_2\xi_2 + \cdots + \alpha_n\xi_n$$
$$\vdots$$
$$u_n = \alpha_n\xi_n$$

Then

$$\hat{\xi}_1 = u_1 / \|u_1\|$$
$$\bar{\xi}_j = w_j / \|w_j\| \quad \text{for } i = 2,3,\cdots,n \quad \text{where } w_j = u_j - \sum_{k=1}^{j-1} [(u_j)^T \bar{\xi}_k]\bar{\xi}_k$$

- Use $\bar{\xi}_1, \bar{\xi}_2, \cdots, \bar{\xi}_n$ as a new search direction for exploratory search.

Remark 4: More complicated methods can be derived. However, the next Powell's Conjugate Direction Method is better if a more sophisticated algorithm is to be used.

• *Powell's Conjugate Direction Method*

- Motivations
  - It is based on the model of a quadratic objective function.
  - If the objective function of *n* variables is quadratic and in the form of perfect square, then the optimum can be found after exactly *n* single variable searches.
  - Quadratic functions:

    $q(x) = a + b^T x + 0.5 x^T \mathbf{C} x$

    Similarity transform (Diagonalization): Find **T** with x=**T**z so that

    $Q(x) = x^T \mathbf{C} x = z^T \mathbf{T}^T \mathbf{C} \mathbf{T} z = z^T \mathbf{D} z$ (**D** is a diagonal matrix)

    cf) If **C** is diagonalizable, **T** is the eigenvector of **C**.
  - For optimization, **C** of objective function is not generally available.

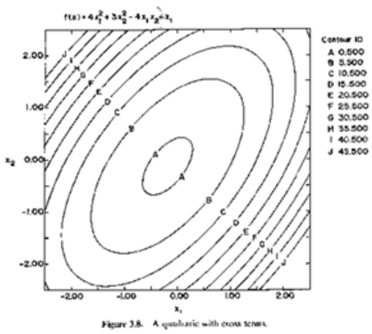


Figure 3.8. A quadratic with cross terms.

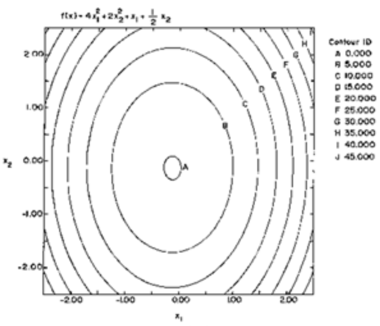


Figure 3.9. A quadratic without cross terms.

- *Conjugate directions*
  - Definition:

    Given an *nxn* symmetric matrix **C**, the direction s_1, s_2,, …, s_r ($r \le n$) are said to be **C** conjugate if the directions are linearly independent and

    $s_i^T \mathbf{C} s_j = 0$ for all $i \ne j$.

  Remark 1: If $s_i^T s_j = 0$ for all $i \ne j$ they are orthogonal.

Remark 2: If s_i is the i-th column of a matrix **T**, then $\mathbf{T}^T\mathbf{C}\mathbf{T}$ is a diagonal matrix.

- *Parallel subspace property*

For a 2D-quadratic function, pick a direction d and two initial points x_1 and x_2.

Let z_i be the minimum point of $\min_{\lambda} f(x_i + \lambda d)$.

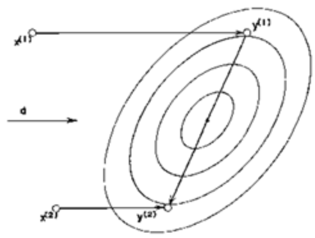


Figure 3.10. Conjugacy in two dimensions.

$$\frac{\partial f}{\partial \lambda} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial \lambda} = (b^T + x^T\mathbf{C})d\Big|_{x=z_1 \text{ or } z_2} = 0$$

$$(b^T + z_1^T\mathbf{C})d = (b^T + z_2^T\mathbf{C})d = 0 \Rightarrow (z_1 - z_2)^T\mathbf{C}d = 0$$

$\therefore$ $(z_1 - z_2)$ and d are conjugate directions.

- *Extended Parallel subspace property*

For a quadratic function, pick n-direction $s_i = e_i$ ($i = 1, 2, \cdots, n$) and a initial points x_0.

i) Perform a line search in s_n direction and let the result be x_1.

ii) Perform n line searches for $s_1, s_2, \ldots, s_n$ starting from a last line search result.
   Let the last point be z_1 after n line search.

iii) Then replace s_i with s_{i+1} (i=1,2,…n-1) and set $s_n = (z_1 - x_1)$.

iv) Repeat ii) and iii) (n-1) times. Then $s_1, s_2, \ldots, s_n$ are conjugate each other.

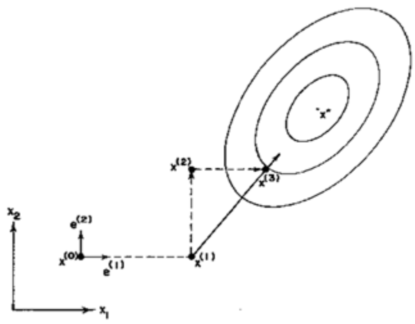


Figure 3.11. Conjugacy from a single point.

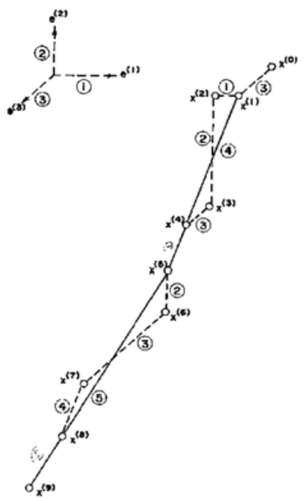


Figure 3.12. Construction of conjugate directions in three dimensions.

- Given **C**, find n-conjugate directions

A. Choose n linearly independent vectors, $u_1, u_2, \ldots, u_n$. Let $z_1 = u_1$.

$$z_j = u_j - \sum_{k=1}^{j-1}\left[\frac{u_j^T A z_k}{z_k^T A z_k}\right] z_k \quad \text{for } j = 2, 3, \cdots, n$$

B. Recursive method (from an arbitrary direction z_1)

$$z_2 = Az_1 - \left(\frac{z_1^T A^2 z_1}{z_1^T A z_1}\right) z_1$$

$$z_{j+1} = Az_j - \left(\frac{z_j^T A^2 z_j}{z_j^T A z_j} \right) z_j - \left(\frac{z_j^T A^2 z_j}{z_{j-1}^T A z_{j-1}} \right) z_{j-1} \text{ for } j = 2, 3, \cdots, n\text{-}1$$

cf) Select b so that $z_i^T A z_{i+1} = z_i^T A(Az_i + bz_i) = 0$

- *Powell's conjugate direction method*

Select initial guess x_0 and a set of n linearly independent directions ($s_i = e_i$).

i) Perform a line search in e_n direction and let the result be $x_0^{(1)}$ and $x^{(1)} = x_0^{(1)}$ (k=1).

ii) Starting at $x^{(k)}$, perform n line search in s_i direction from the previous point of line search result for $i = 1, 2, \cdots, n$. Let the point obtained from the each line search be $x_i^{(k)}$.

iii) Form a new conjugated direction, s_{n+1} using the extended parallel subspace property. $s_{n+1} = (x_n^{(k)} - x^{(k)}) / \left\| x_n^{(k)} - x^{(k)} \right\|$.

iv) If $\left\| s_{n+1} \right\| < \varepsilon$, then $x^* = x^{(k)}$ and stop.

v) Perform additional line search in s_{n+1} direction and let the result be $x_{n+1}^{(k)}$.

vi) Delete s_1 and replace s_i with s_{i+1} for $i = 1, 2, \cdots, n$. Then set $x^{(k+1)} = x_{n+1}^{(k)}$ and k=k+1 and go to ii).

Remark 1: If the objective function is quadratic, the optimum will be found after n^2 line searches.

Remark 2: Before step vi), needs a procedure to check the linear independence of the conjugate direction set.

A. Modification by Sargent

Suppose λ_k^* is obtained by $\min_{\lambda} f(x^{(k)} + \lambda_k s_{n+1})$. $\left(x^{(k+1)} = x^{(k)} + \lambda_k s_{n+1} \right)$

And let $f(x_{m-1}^{(k)}) - f(x_m^{(k)}) = \max_j \left[f(x_{j-1}^{(k)}) - f(x_j^{(k)}) \right]$

Check if $\left| \lambda_k^* \right| < \left[\dfrac{f(x^{(k)}) - f(x^{(k+1)})}{f(x_{m-1}^{(k)}) - f(x_m^{(k)})} \right]^{0.5}$

If yes, use old directions again. Else delete s_m and add s_{n+1}.

B. Modification by Zangwill

Let $D^{(k)} = \left[s_1 \ s_2 \ \cdots \ s_n \right]$ and $\left\| x_{m-1}^{(k)} - x_m^{(k)} \right\| = \max_j \left\| x_{j-1}^{(k)} - x_j^{(k)} \right\|$

Check if $\dfrac{\left\| x_{m-1}^{(k)} - x_m^{(k)} \right\|}{\left\| s_{n+1} \right\|} \det(D^{(k)}) \le \varepsilon$

If yes, use old directions again. Else delete s_m and add s_{n+1}.

Remark 3: This method will converge to a local minimum at *superlinear convergence rate*.

cf) Let $\lim_{k\to\infty}\frac{\left\|\varepsilon^{(k+1)}\right\|}{\left\|\varepsilon^{(k)}\right\|^{r}}=C$ where $\varepsilon^{(k)}=x^{(k)}-x^{*}$

If C<1, then it is convergent at r-order of convergence rate.

r=1 : *linear convergence* rate

r=2 : *quadratic convergence* rate

r=1 and C=0 : *superlinear convergence* rate

➔ Among unconstrained multidimensional direct search methods, the Powell's conjugate direction method is the most recommended method.

3.3 Gradient based Methods

- All techniques employ a similar iteration procedure:

$$x^{(k+1)}=x^{(k)}+\alpha^{(k)}s(x^{(k)})$$

where $\alpha^{(k)}$ is the step-length parameter found by a line search, and $s(x^{(k)})$ is the search direction.

- The $\alpha^{(k)}$ is decided by a line search in the search direction $s(x^{(k)})$.

i) Start from an initial guess $x^{(0)}$ (k=0).

ii) Decide the search direction $s(x^{(k)})$.

iii) Perform a line search in the search direction and get an improved point $x^{(k+1)}$.

iv) Check the termination criteria. If satisfied, then stop.

v) Else set k=k+1 and go to ii).

- Gradient based methods require accurate values of first derivative of $f(x)$.
- Second-order methods use values of second derivative of $f(x)$ additionally.

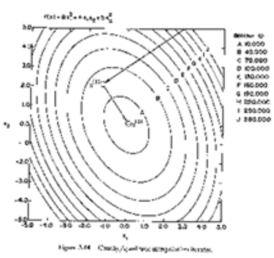
• *Steepest descent Method* (Cauchy's Method)

$$f(x)=f(\bar{x})+\nabla f(\bar{x})^{T}\Delta x+\cdots \text{ (higher-order terms ignored)}$$

$$f(\bar{x})-f(x)=-\nabla f(\bar{x})^{T}\Delta x$$

The *steepest descent direction*: Maximize the decent by choosing Δx

$$\Delta x^{*}=\arg\max_{\Delta x}\left(-\nabla f(\bar{x})^{T}\Delta x\right)=-\alpha\nabla f(\bar{x}) \quad (\alpha>0)$$



The search direction: $s(x^{(k)})=-\alpha^{(k)}\nabla f(x^{(k)})$

Termination criteria:

$$\left\|\nabla f(x^{(k)})\right\|<\varepsilon_{f} \text{ and/or } \left\|x^{(k+1)}-x^{(k)}\right\|/\left\|x^{(k)}\right\|<\varepsilon_{x}$$

Remark 1: This method shows slow improvement near optimum. ($\because \nabla f(x)\approx 0$)

Remark 2: This method possesses a *descent property*.

$$\nabla f(x^{(k)})^T s(x^{(k)}) < 0$$

- *Newton's Method* (Modified Newton's Method)

  $\nabla f(x) = \nabla f(\overline{x}) + \nabla^2 f(\overline{x})\Delta x + \cdots$ (higher-order terms ignored)

  The *optimality condition* for approximate derivative at $\overline{x}$:

  $$\nabla f(x) = \nabla f(\overline{x}) + \nabla^2 f(\overline{x})\Delta x = 0$$

  $$\therefore \Delta x = -\nabla^2 f(\overline{x})^{-1} \nabla f(\overline{x})$$

  The search direction: $s(x^{(k)}) = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$ (Newton's method)

  $s(x^{(k)}) = -\alpha^{(k)} \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$ (Modified Newton's method)

  Remark 1: In the modified Newton's method, the step-size parameter $\alpha^{(k)}$ is decided by a line search to ensure for the best improvement.

  Remark 2: The calculation of the inverse of Hessian matrix $\nabla^2 f(x^{(k)})$ imposes quite heavy computation when the dimension of the optimization variable is high.

  Remark 3: The family of Newton's methods exhibits quadratic convergence.

  $$\left\|\varepsilon^{(k+1)}\right\| \le C \left\|\varepsilon^{(k)}\right\|^2$$ (C is related to the condition of Hessian $\nabla^2 f(x^{(k)})$)

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

  Also, if the initial condition is chosen such that $\left\|\varepsilon^{(0)}\right\| < \frac{1}{C}$, the method will converge. It implies that the initial condition is chosen poorly, it may diverge.

  Remark 4: The family of Newton's methods does not possess the *descent property*.

  $\nabla f(x^{(k)})^T s(x^{(k)}) = -\nabla f(x^{(k)})^T \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)}) < 0$ only if the Hessian is *positive definite*.

- *Marquardt's Method* (Marquardt's compromise)
  - This method combines steepest descent and Newton's methods.
  - The steepest descent method has good reduction in f when $x^{(k)}$ is far from x^*.
  - Newton's method possesses quadratic convergence near x^*.
  - The search direction: $s(x^{(k)}) = -[\mathbf{H}^{(k)} + \lambda^{(k)}\mathbf{I}]^{-1} \nabla f(x^{(k)})$
  - Start with large $\lambda^{(0)}$, say 10^4 (steepest descent direction) and decrease to zero.

    If $f(x^{(k+1)}) < f(x^{(k)})$, then set $\lambda^{(k+1)} = 0.5\lambda^{(k)}$.

Else set $\lambda^{(k+1)} = 2\lambda^{(k)}$.

Remark 1: This is quite useful for the problems with objective function form of

$$f(x) = f_1^2(x) + f_2^2(x) + \cdots + f_m^2(x) \quad \text{(Levenberg-Marquardt method)}$$

Remark 2: *Goldstein and Price Algorithm*

Let δ $(\delta < 0.5)$ and γ be positive numbers.

i) Start from $x^{(0)}$ with $k=1$. Let $\phi(x^{(0)}) = \nabla f(x^{(0)})$.

ii) Check if $\left\|\nabla f(x^{(k)})\right\| < \varepsilon$. If yes, then stop.

iii) Calculate $g(x^{(k)}, \theta_k) = \dfrac{f(x^{(k)}) - f(x^{(k)} - \theta_k \phi(x^{(k)}))}{\theta_k \nabla f(x^{(k)})^T \phi(x^{(k)})}$

If $g(x^{(k)}, 1) < \delta$, select θ_k such that $\delta < g(x^{(k)}, \theta_k) < 1 - \delta$.
Else, $\theta_k = 1$.

iv) Let $\mathbf{Q} = [Q_1 \; Q_2 \; \cdots \; Q_n]$ (approximation of the Hessian)

where $Q_i = \dfrac{\nabla f(x^{(k)} + \gamma \left\|\nabla f(x^{(k-1)})\right\| e_i) - \nabla f(x^{(k)})}{\gamma \left\|\nabla f(x^{(k-1)})\right\|}$

If $\mathbf{Q}(x^{(k)})$ is singular or $\nabla f(x^{(k)})^T Q(x^{(k)})^{-1} \nabla f(x^{(k)}) \le 0$,
then $\phi(x^{(k)}) = \nabla f(x^{(k)})$. Else $\phi(x^{(k)}) = Q(x^{(k)})^{-1} \nabla f(x^{(k)})$.

v) Set $x^{(k+1)} = x^{(k)} - \theta_k \phi(x^{(k)})$ and $k=k+1$. Then go to ii).

• *Conjugate Gradient Method*

- *Quadratically convergent method*: The optimum of a n-D quadratic function can be found in approximately n steps using exact arithmetic.
- This method generates conjugate directions using gradient information.
- For a quadratic function, consider two distinct points, $x^{(0)}$ and $x^{(1)}$.

Let $g(x^{(0)}) = \nabla f(x^{(0)}) = \mathbf{C}x^{(0)} + b$ and

$g(x^{(1)}) = \nabla f(x^{(1)}) = \mathbf{C}x^{(1)} + b$.

$$\Delta g(x) = g(x^{(1)}) - g(x^{(0)}) = \mathbf{C}(x^{(1)} - x^{(0)}) = \mathbf{C}\Delta x$$

(Property of quadratic function: expression for a change in gradient)

- Iterative update equation: $x^{(k+1)} = x^{(k)} + \alpha^{(k)} s(x^{(k)})$

$$\frac{\partial f(x^{(k+1)})}{\partial \alpha^{(k)}} = b^T s^{(k)} + s^{(k)T} \mathbf{C}(x^{(k)} + \alpha^{(k)} s^{(k)})$$

$$= s^{(k)T}(b + \mathbf{C}x^{(k)}) + s^{(k)T} \mathbf{C} \alpha^{(k)} s^{(k)} = 0$$

$$\therefore \alpha^{(k)} = -\frac{s^{(k)^T} \nabla f(x^{(k)})}{s^{(k)^T} \mathbf{C} s^{(k)}} \quad \text{and} \quad \nabla f(x^{(k+1)})^T s^{(k)} = 0 \quad \text{(optimality of line search)}$$

- Search direction: $s^{(k)} = -g^{(k)} + \sum_{i=0}^{k-1} \gamma^{(i)} s^{(i)}$ with $s^{(0)} = -g^{(0)}$

- In order that the $s^{(k)}$ is **C**-conjugate to all previous search direction

i) Choose $\gamma^{(0)}$ such that $s^{(1)^T} \mathbf{C} s^{(0)} = 0$

where $s^{(1)} = -g^{(1)} + \gamma^{(0)} s^{(0)} = -g^{(1)} - \gamma^{(0)} g^{(0)}$

$$\Rightarrow [g^{(1)} + \gamma^{(0)} g^{(0)}]^T C[\Delta x / \alpha^{(0)}] = 0 \quad (\because \Delta x = \alpha^{(0)} s^{(0)})$$

$$\Rightarrow [g^{(1)} + \gamma^{(0)} g^{(0)}]^T \Delta g = 0 \quad \text{(property of quadratic function)}$$

$$\therefore \gamma^{(0)} = -\frac{\Delta g^T g^{(1)}}{\Delta g^T g^{(0)}} = \frac{(g^{(1)} - g^{(0)})^T g^{(1)}}{(g^{(0)} - g^{(1)})^T g^{(0)}} = \frac{g^{(1)^T} g^{(1)}}{g^{(0)^T} g^{(0)}} = \frac{\left\| g^{(1)} \right\|^2}{\left\| g^{(0)} \right\|^2}$$

0

ii) Choose $\gamma^{(0)}$ and $\gamma^{(1)}$ such that $s^{(2)^T} \mathbf{C} s^{(1)} = 0$ and $s^{(2)^T} \mathbf{C} s^{(0)} = 0$.

where $s^{(2)} = -g^{(2)} - \gamma^{(0)} g^{(0)} - \gamma^{(1)} (g^{(1)} + \gamma^{(0)} g^{(0)})$

$$\Rightarrow \gamma^{(0)} = 0 \quad \text{and} \quad \therefore \gamma^{(1)} = \frac{\left\| g^{(2)} \right\|^2}{\left\| g^{(1)} \right\|^2}$$

ii) In general, $s^{(k)} = -g^{(1)} + \gamma^{(k)} s^{(k-1)}$

$$s^{(k)} = -g^{(k)} + \left[\frac{\left\| g^{(k)} \right\|^2}{\left\| g^{(k-1)} \right\|^2} \right] s^{(k-1)} \quad \textit{(Fletcher and Reeves Method)}$$

Remark 1: Variations of conjugate gradient method

i) Miele and Cantrell (Memory gradient method)

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

where $\gamma^{(1)}$ is sought directly at each iteration such that $s^{(k)^T} \mathbf{C} s^{(k-1)} = 0$.

cf) Use when the objective and gradient evaluations are very inexpensive.

ii) Daniel

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

iii) Sorenson and Wolfe

$$s^{(k)} = -\nabla f(x^{(k)}) + \frac{\Delta g(x^{(k)})^T g(x^{(k)})}{\Delta g(x^{(k)})^T s^{(k-1)}} s^{(k-1)}$$

iv) Polak and Ribiere

$$s^{(k)} = -\nabla f(x^{(k)}) + \frac{\Delta g(x^{(k)})^T g(x^{(k)})}{\left\| g(x^{(k-1)}) \right\|^2} s^{(k-1)}$$

Remark 2: These methods are doomed to a linear rate of convergence in the absence of periodic restarts to avoid the dependency of the directions.

→ Set $s^{(k)} = -g(x^{(k)})$ whenever $\left| g(x^{(k)})^T g(x^{(k-1)}) \right| \ge 0.2 \left\| g(x^{(k)}) \right\|^2$ or every n iterations.

Remark 3: The Polak and Ribiere method is more efficient for general functions and less sensitive to inexact line search than the Fletcher and Reeves.

- *Quasi-Newton Method*
  - Mimic the Newton's method using only first-order information
  - Form of search direction: $s(x^{(k)}) = -\mathbf{A}^{(k)} \nabla f(x^{(k)})$

    where **A** is an *nxn* matrix call the *metric*.
  - *Variable metric methods* employ search direction of this form.
  - *Quasi-Newton method* is a variable metric method with the quadratic property.

    $\Delta x = \mathbf{C}^{-1} \Delta g$

  - Recursive form for estimation of the inverse of Hessian

    $\mathbf{A}^{(k+1)} = \mathbf{A}^{(k)} + \mathbf{A}_c^{(k)}$ ($\mathbf{A}_c^{(k)}$ is a correction to the current metric)

  - If $\mathbf{A}^{(k)}$ approaches to $\mathbf{H}^{-1} = \nabla^2 f(x^*)^{-1}$, on additional line search will produce the minimum if the function is quadratic.
  - Assume $\mathbf{H}^{-1} = \beta \mathbf{A}^{(k)}$. Then $\Delta x^{(k)} = \beta \mathbf{A}^{(k)} \Delta g^{(k)} \approx \beta \mathbf{A}^{(k+1)} \Delta g^{(k)}$

    $\Rightarrow \mathbf{A}_c^{(k)} \Delta g^{(k)} = \Delta x^{(k)} / \beta - \mathbf{A}^{(k)} \Delta g^{(k)}$

    Family of solutions

    $$\Rightarrow \mathbf{A}_c^{(k)} = \frac{1}{\beta}\left(\frac{\Delta x^{(k)} y^T}{y^T \Delta g^{(k)}} \right) - \frac{\mathbf{A}^{(k)} \Delta g^{(k)} z^T}{z^T \Delta g^{(k)}}$$ (y and z are arbitrary vectors)

  - *DFP method* (Davidon-Fletcher-Powell)

    Let $\beta = 1$, $y = \Delta x^{(k)}$ and $z = \mathbf{A}^{(k)} \Delta g^{(k)}$.

    $$\Rightarrow \mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \left(\frac{\Delta x^{(k-1)} \Delta x^{(k-1)^T}}{\Delta x^{(k-1)^T} \Delta g^{(k-1)}} \right) - \frac{\mathbf{A}^{(k-1)} \Delta g^{(k-1)} \Delta g^{(k-1)^T} \mathbf{A}^{(k-1)}}{\Delta g^{(k-1)^T} \mathbf{A}^{(k-1)} \Delta g^{(k-1)}}$$

    - If $\mathbf{A}^{(0)}$ is any symmetric positive definite, then $\mathbf{A}^{(k)}$ will be so in the absence of round-off error. ($\mathbf{A}^{(0)} = \mathbf{I}$ is a convenient choice.)

$$z^T \mathbf{A}^{(k)} z = z^T \mathbf{A}^{(k-1)} z + \left(\frac{z^T \Delta x^{(k-1)} \Delta x^{(k-1)^T} z}{\Delta x^{(k-1)^T} \Delta g^{(k-1)}} \right) - \frac{z^T \mathbf{A}^{(k-1)} \Delta g^{(k-1)} \Delta g^{(k-1)^T} \mathbf{A}^{(k-1)} z}{\Delta g^{(k-1)^T} \mathbf{A}^{(k-1)} \Delta g^{(k-1)}}$$

$$= a^T a - \frac{(a^T b)^2}{b^T b} + \frac{(z^T \Delta x^{(k-1)})^2}{\Delta x^{(k-1)^T} \Delta g^{(k-1)}}$$ where $a = A^{(k-1)^{1/2}} z$, $b = A^{(k-1)^{1/2}} \Delta g^{(k-1)}$

i) $\Delta x^{(k-1)^T} \Delta g^{(k-1)} = \Delta x^{(k-1)^T} g^{(k)} - \Delta x^{(k-1)^T} g^{(k-1)} = -\Delta x^{(k-1)^T} g^{(k-1)}$

$\therefore \Delta x^{(k-1)^T} \Delta g^{(k-1)} = -(-\alpha^{(k-1)} g^{(k-1)^T} \mathbf{A}^{(k-1)} g^{(k-1)}) > 0$

ii) $\left(a^T a\right)\left(b^T b\right) - \left(a^T b\right)^2 \geq 0$ (Schwarz inequality)

iii) If a and b are proportional (z and $\Delta g^{(k-1)}$ are too),

$\left(a^T a\right)\left(b^T b\right) - \left(a^T b\right)^2 = 0$.

but $\Delta x^{(k-1)^T} z = c\Delta x^{(k-1)^T} \Delta g^{(k-1)} = -c\alpha^{(k-1)} g^{(k-1)^T} \mathbf{A}^{(k-1)} g^{(k-1)} \neq 0$

$\Rightarrow z^T \mathbf{A} z > 0$

- This method has the descent property.

$\Delta f = \nabla f(x^{(k)})^T \Delta x = -\alpha^{(k)} \nabla f(x^{(k)})^T \mathbf{A}^{(k)} \nabla f(x^{(k)}) < 0$ for $\alpha^{(k)} > 0$

- Variations

- *McCormick* (Pearson No.2)

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \frac{(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)} \Delta g^{(k-1)}) \Delta x^{(k-1)^T}}{\Delta x^{(k-1)^T} \Delta g^{(k-1)}}$$

- *Pearson* (Pearson No.3)

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \frac{(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)} \Delta g^{(k-1)}) \Delta g^{(k-1)^T} \mathbf{A}^{(k-1)}}{\Delta g^{(k-1)^T} \mathbf{A}^{(k-1)} \Delta g^{(k-1)}}$$

- *Broydon 1965 method* (not symmetric)

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \frac{(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)} \Delta g^{(k-1)}) \Delta x^{(k-1)^T} \mathbf{A}^{(k-1)}}{\Delta x^{(k-1)^T} \mathbf{A}^{(k-1)} \Delta g^{(k-1)}}$$

- *Broydon symmetric Rank-one method* (1967)

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \frac{(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)}\Delta g^{(k-1)})(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)}\Delta g^{(k-1)})^T}{(\Delta x^{(k-1)} - \mathbf{A}^{(k-1)}\Delta g^{(k-1)})^T \Delta g^{(k-1)}}$$

- *Zoutendijk*

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} - \frac{\mathbf{A}^{(k-1)}\Delta g^{(k-1)}\Delta g^{(k-1)^T}\mathbf{A}^{(k-1)}}{\Delta g^{(k-1)^T}\mathbf{A}^{(k-1)}\Delta g^{(k-1)}}$$

- *BFS method* (Broydon-Fletcher-Shanno, rank-two method)

$$\boldsymbol{A}^{(k)} = \left[\boldsymbol{I} - \frac{\Delta x^{(k-1)}\Delta g^{(k-1)^T}}{\Delta x^{(k-1)^T}\Delta g^{(k-1)}}\right]\boldsymbol{A}^{(k-1)}\left[\boldsymbol{I} - \frac{\Delta x^{(k-1)}\Delta g^{(k-1)^T}}{\Delta x^{(k-1)^T}\Delta g^{(k-1)}}\right]^T + \frac{\Delta x^{(k-1)}\Delta x^{(k-1)^T}}{\Delta x^{(k-1)^T}\Delta g^{(k-1)}}$$

- *Invariant DFP* (Oren, 1974)

$$\mathbf{A}^{(k)} = \frac{\Delta x^{(k-1)}\Delta g^{(k-1)^T}}{\Delta g^{(k-1)^T}\mathbf{A}^{(k-1)}\Delta g^{(k-1)}}\left[\mathbf{A}^{(k-1)} - \frac{\mathbf{A}^{(k-1)}\Delta g^{(k-1)}\Delta g^{(k-1)^T}\mathbf{A}^{(k-1)}}{\Delta g^{(k-1)^T}\mathbf{A}^{(k-1)}\Delta g^{(k-1)}}\right] + \frac{\Delta x^{(k-1)}\Delta x^{(k-1)^T}}{\Delta x^{(k-1)^T}\Delta g^{(k-1)}}$$

- *Hwang* (Unification of many variations)

$$\mathbf{A}^{(k)} = \mathbf{A}^{(k-1)} + \left[\Delta x^{(k-1)} \quad \mathbf{A}^{(k-1)}\Delta g^{(k-1)}\right]\mathbf{B}^{(k-1)}\left[\Delta x^{(k-1)} \quad \mathbf{A}^{(k-1)}\Delta g^{(k-1)}\right]^T$$

where **B** is 2x2 and $\mathbf{B}^{(k-1)}\left[\Delta x^{(k-1)} \quad \mathbf{A}^{(k-1)}\Delta g^{(k-1)}\right]^T \Delta g^{(k-1)} = [\omega \ \ -1]^T$.

Remark: If $\omega = 1$ and $B^{(k)} = diag(1/\Delta x^{(k)^T}\Delta g^{(k)}, -1/\Delta g^{(k)^T}\mathbf{A}^{(k)}\Delta g^{(k)})$, this method will be same as DFP method.

Remark 1: As these methods iterate, $\mathbf{A}^{(k)}$ tends to become ill-conditioned or nearly singular. Thus, they require restart. ($\mathbf{A}^{(k)} = \mathbf{I}$: loss of 2nd-order information)
cf) *Condition number*= ratio of max. and min. magnitudes of eigenvalues of **A**.
Ill-conditioned: if **A** has large condition number

Remark 2: The size of $\mathbf{A}^{(k)}$ is quite big if *n* is large. (computation and storage)

Remark 3: BFS method is widely used and known that it has *decreased need for restart* and it is *less dependent on exact line search*.

Remark 4: The line search is the *most time-consuming phase* of these methods.

Remark 5: If the gradient is not explicitly available, the numerical gradient can be obtained using, for example, *forward and central difference approximations*. If the changes in *x* and/or *f* between iterations are small, the central difference approximation is better at the cost of more computation.

3.4 Comparison of Methods

- Test functions
  - Rosenblock's function: $f(x)=100(x_2-x_1^2)^2+(1-x_1)^2$
  - Fenton and Eason's Function: $f(x)=\frac{1}{10}\left\{12+x_1^2+\frac{1+x_2^2}{x_1^2}+\frac{x_1^2x_2^2+100}{\left(x_1x_2\right)^4}\right\}$
  - Wood's function: $$\begin{aligned} f(x)&=100(x_2-x_1^2)^2+(1-x_1)^2+90(x_4-x_3^2)^2+(1-x_3)^2\\ &+10.1\left[(x_2-1)^2+(x_4-1)^2\right]+19.8(x_2-1)(x_4-1)\end{aligned}$$

- Test results
  - Himmelblau (1972): BFS, DFP and Powell's direct search methods are superior.
  - Sargent and Sebastian (1971): BFS among BFS, DFP and FR methods
  - Shanno and Phua (1980): BFS
  - Reklaitis (1983): FR among Cauchy, FR, DFP, and BFS methods