Lecture 7 Unconstrained nonlinear programming

Weinan E[1,2] and Tiejun Li[2]

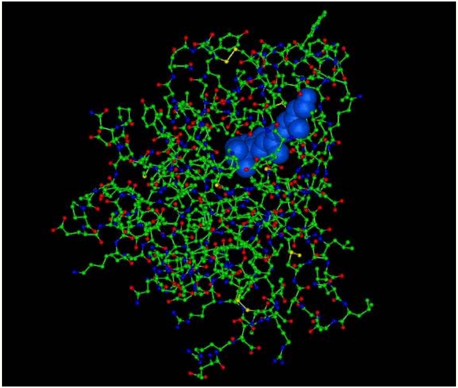
[1] Department of Mathematics,
Princeton University,
weinan@princeton.edu

[2] School of Mathematical Sciences,
Peking University,
tieli@pku.edu.cn
No.1 Science Building, 1575

Outline

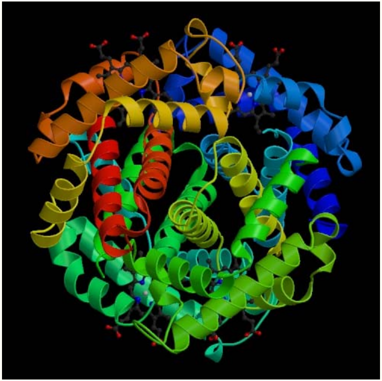
Energy minimization: virtual drug design

- Virtual drug design is to find a best position of a ligand (a small protein molecule) interacting with a large target protein molecule. It is equivalent to an energy minimization problem.



Energy minimization: protein folding

- Protein folding is to find the minimal energy state of a protein molecule from its sequence structure. It is an outstanding open problem for global optimization in the molecular mechanics.



Energy minimization: mathematical formulation

- Molecular force field

$$V_{\text{total}} = \sum_i \frac{k_{r_i}}{2}(r_i - r_{i0})^2 + \sum_i \frac{k_{\theta_i}}{2}(\theta_i - \theta_{i0})^2 + \sum_i \frac{V_{ni}}{2}(1 + \cos(n\phi_i - \gamma_i))$$
$$+ \sum_{ij} 4\epsilon\left(\left(\frac{\sigma_{ij}}{r_{ij}}\right)^6 - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12}\right) + \sum_{ij} \frac{q_i q_j}{\epsilon r_{ij}}$$

- Webpage for the explanation of the force field
- Energy minimization problem with respect to all the configuration of the atoms

$$\min V_{total}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$$

Nonlinear least squares

- Suppose that we have a series of experimental data (t_i, y_i), $i = 1, \ldots, m$. We wish to find parameter $\boldsymbol{x} \in \mathbb{R}^n$ such that the remainder

$$r_i(\boldsymbol{x}) = y_i - f(t_i, \boldsymbol{x}), \quad i = 1, \ldots, m$$

minimized.

- Mathematically, define error function

$$\phi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{r}(\boldsymbol{x})$$

where $\boldsymbol{r} = (r_1, \ldots, r_m)$ such that

$$\min_{\boldsymbol{x}} \phi(\boldsymbol{x}).$$

- Because the function f is nonlinear, it is called a nonlinear least square problem.

Optimal control problem

- Classical optimal control problem:

$$\min \int_0^T f(x,u)dt$$

such that the constraint

$$\frac{dx}{dt} = g(x,u), \quad x(0) = x_0, x(T) = x_T$$

is satisfied. Here $u(t)$ is the control function, $x(t)$ is the output.

- It is a nonlinear optimization in function space.

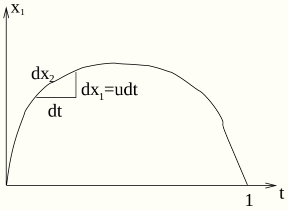
Optimal control problem

- Example: Isoparametric problem.

$$\max_u \int_0^1 x_1(t)dt$$

$$\frac{dx_1}{dt} = u, \quad \frac{dx_2}{dt} = \sqrt{1+u^2}.$$

$$x_1(0) = x_1(1) = 0, \quad x_2(0) = 0, x_2(1) = \frac{\pi}{3}$$



Outline

Application examples

Numerical methods

Iterations

- Iterative methods

  Object: construct sequence $\{\boldsymbol{x}_k\}_{k=1}^{\infty}$, such that $\boldsymbol{x}_k$ converge to a fixed vector $\boldsymbol{x}^*$, and $\boldsymbol{x}^*$ is the solution of the linear system.

- General iteration idea:

  If we want to solve equations

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0},$$

  and the equation $\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x})$ has the same solution as it, then construct

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k).$$

  If $\boldsymbol{x}_k \to \boldsymbol{x}^*$, then $\boldsymbol{x}^* = \boldsymbol{f}(\boldsymbol{x}^*)$, thus the root of $\boldsymbol{g}(\boldsymbol{x})$ is obtained.

Convergence order

- Suppose an iterating sequence $\lim \boldsymbol{x}_n = \boldsymbol{x}^*$, and

$$|\boldsymbol{x}_n - \boldsymbol{x}^*| \leq \epsilon_n$$

where ϵ_n is called error bound. If

$$\lim \frac{\epsilon_{n+1}}{\epsilon_n} = C,$$

when

1. $0 < C < 1$, $\boldsymbol{x}_n$ is called linear convergence;

$$q, q^2, q^3, \cdots, q^n, \cdots, \quad (q < 1)$$

2. $C = 1$, $\boldsymbol{x}_n$ is called sublinear convergence;

$$1, \frac{1}{2}, \frac{1}{3}, \cdots, \frac{1}{n}, \cdots$$

3. $C = 0$, $\boldsymbol{x}_n$ is called superlinear convergence;

$$1, \frac{1}{2!}, \frac{1}{3!}, \cdots, \frac{1}{n!}, \cdots$$

Convergence order

- If

$$\lim \frac{\epsilon_{n+1}}{\epsilon_n^p} = C, \quad C > 0,\ p > 1$$

then x_n is called p-th order convergence.

$$q, q^p, q^{p^2}, \cdots, q^{p^n}, \cdots$$

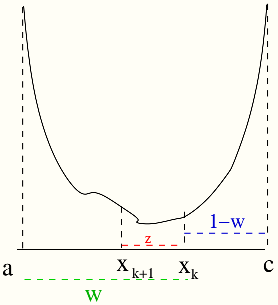
- Numerical examples for different convergence orders

Remark on p-th order convergence

- If $p = 1$, i.e. linear convergence, the number of significant digits is increasing linearly, such as $2, 3, 4, 5, \ldots$;
- If $p > 1$, the number of significant digits is increasing exponentially $(O(p^n))$. Suppose $p = 2$, then the number of significant digits is increased as $2, 4, 8, 16, \ldots$!! So a very accurate result will be obtained after $4 - 5$ iterations;

Golden section method

- Suppose there is a triplet (a, x_k, c) and $f(x_k) < f(a)$, $f(x_k) < f(c)$, we want to find x_{k+1} in (a, c) to perform a section. Suppose x_{k+1} is in (a, x_k).



- If $f(x_{k+1}) > f(x_k)$, then the new search interval is (x_{k+1}, c); If $f(x_{k+1}) < f(x_k)$, then the new search interval is (a, x_k).

Golden section method

- Define
$$w = \frac{x_k - a}{c - a}, \quad 1 - w = \frac{c - x_k}{c - a}$$
and
$$z = \frac{x_k - x_{k+1}}{c - a}.$$
If we want to minimize the worst case possibility (for two cases), we must make $w = z + (1 - w)$. $(w > \frac{1}{2})$
- Pay attention that w is also obtained from the previous stage of applying same strategy. This scale similarity implies
$$\frac{z}{w} = 1 - w$$
we have
$$w = \frac{\sqrt{5} - 1}{2} \approx 0.618$$
This is called Golden section method.

Golden section method

- Golden section method is a method to find the local minimum of a function f.
- Golden section method is a linear convergence method. The contraction coefficient is $C = 0.618$.
- Golden section method for Example

$$\min \varphi(x) = 0.5 - xe^{-x^2}$$

where $a = 0, c = 2$.

One dimensional Newton's method

- Suppose we want to minimize $\varphi(x)$

$$\min_x \varphi(x)$$

- Taylor expansion at current iteration point x_0

$$\varphi(x) = \varphi(x_0) + \varphi'(x_0)(x - x_0) + \frac{1}{2}\varphi''(x_0)(x - x_0)^2 + \cdots$$

- Local quadratic approximation

$$\varphi(x) \approx g(x) = \varphi(x_0) + \varphi'(x_0)(x - x_0) + \frac{1}{2}\varphi''(x_0)(x - x_0)^2$$

- Minimize $g(x)$ at $g'(x) = 0$, then

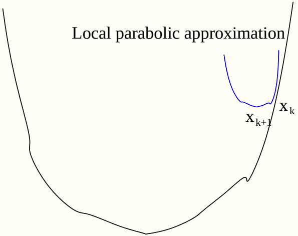
$$x_1 = x_0 - \frac{\varphi'(x_0)}{\varphi''(x_0)}$$

- Newton's method

$$x_{k+1} = x_k - \frac{\varphi'(x_k)}{\varphi''(x_k)}$$

One dimensional Newton's method

- Graphical explanation



- Example

$$\min \varphi(x) = 0.5 - xe^{-x^2}$$

where $x_0 = 0.5$.

One dimensional Newton's method

Theorem

If $\varphi''(x^*) \neq 0$, then Newton's method converges with second order if x^0 is close to x^* sufficiently.

Drawbacks of Newton's method:

1. one needs to compute the second order derivative which is a huge cost (especially for high dimensional case).
2. The initial state x_0 must be very close to x^*.

High dimensional Newton's method

- Suppose we want to minimize $f(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^n$

$$\min_{\boldsymbol{x}} f(\boldsymbol{x})$$

- Taylor expansion at current iteration point $\boldsymbol{x}_0$

$$f(\boldsymbol{x}) = f(\boldsymbol{x}_0) + \nabla f(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_0)^T \nabla^2 f(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0) + \cdots$$

- Local quadratic approximation

$$f(\boldsymbol{x}) \approx g(\boldsymbol{x}) = f(\boldsymbol{x}_0) + \nabla f(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_0)^T \boldsymbol{H}_f(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0)$$

  where $\boldsymbol{H}_f$ is the Hessian matrix defined as $(\boldsymbol{H}_f)_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$.

- Minimize $g(\boldsymbol{x})$ at $\nabla g(\boldsymbol{x}) = 0$, then

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 - \boldsymbol{H}_f(\boldsymbol{x}_0)^{-1} \cdot \nabla f(\boldsymbol{x}_0)$$

- Newton's method

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \boldsymbol{H}_f(\boldsymbol{x}_k)^{-1} \cdot \nabla f(\boldsymbol{x}_k)$$

High dimensional Newton's method

Example

$$\min f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$$

Initial state $\boldsymbol{x}_0 = (-1.2, 1)$.

Steepest decent method

- Basic idea: Find a series of decent directions $\boldsymbol{p}_k$ and corresponding stepsize α_k such that the iterations

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$$

and

$$f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{x}_k).$$

- The negative gradient direction $-\nabla f$ is the "steepest" decent direction, so choose

$$\boldsymbol{p}_k := -\nabla f(\boldsymbol{x}_k)$$

and choose α_k such that

$$\min_{\alpha} f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k)$$

Inexact line search

- To find α such that

$$\min_{\alpha} f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k)$$

is equivalent to perform a one dimensional minimization. But it is enough to find an approximate α by the following inexact line search method.

- Inexact line search is to make the following type of the decent criterion

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1}) \geq \epsilon_0$$

is satisfied.

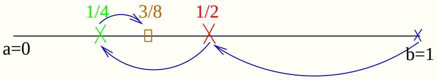
Inexact line search

- An example of inexact line search strategy by half increment (or decrement) method:

$$[a_0, b_0] = [0, +\infty), \quad \alpha_0 = 1; \qquad [a_1, b_1] = [0, 1], \quad \alpha_1 = \frac{1}{2}$$

$$[a_2, b_2] = [0, \frac{1}{2}], \quad \alpha_2 = \frac{1}{4}; \qquad [a_3, b_3] = [\frac{1}{4}, \frac{1}{2}], \quad \alpha_3 = \frac{3}{8}$$

$$\cdots\cdots \quad \cdots\cdots$$



Steepest decent method

Steepest decent method for example

$$\min f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$$

Initial state $\boldsymbol{x}_0 = (-1.2, 1)$.

Dumped Newton's method

- If the initial value of Newton's method is not near the minimum point, a strategy is to apply dumped Newton's method.
- Choose the decent direction as the Newton's direction

$$\boldsymbol{p}_k := -\boldsymbol{H}_f^{-1}(\boldsymbol{x}_k)\nabla f(\boldsymbol{x}_k)$$

and perform the inexact line search for

$$\min_{\alpha} f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k)$$

Conjugate gradient method

Recalling conjugate gradient method for quadratic function

$$\varphi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} - b^T\boldsymbol{x}$$

1. Initial step: $\boldsymbol{x}_0, \boldsymbol{p}_0 = \boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$
2. Suppose we have $\boldsymbol{x}_k, \boldsymbol{r}_k, \boldsymbol{p}_k$, the CGM step
   2.1 Search the optimal α_k along $\boldsymbol{p}_k$;
   $$\alpha_k = \frac{(\boldsymbol{r}_k)^T\boldsymbol{p}_k}{(\boldsymbol{p}_k)^T\boldsymbol{A}\boldsymbol{p}_k}$$
   2.2 Update $\boldsymbol{x}_k$ and gradient direction $\boldsymbol{r}_k$;
   $$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k\boldsymbol{p}_k, \quad \boldsymbol{r}_{k+1} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_{k+1}$$
   2.3 According to the calculation before to form new search direction $\boldsymbol{p}_{k+1}$
   $$\beta_k = -\frac{(\boldsymbol{r}_{k+1})^T\boldsymbol{A}\boldsymbol{p}_k}{(\boldsymbol{p}_k)^T\boldsymbol{A}\boldsymbol{p}_k}, \quad \boldsymbol{p}_{k+1} = \boldsymbol{r}_{k+1} + \beta_k\boldsymbol{p}_k$$

Conjugate gradient method

- Local quadratic approximation of general nonlinear optimization

$$f(\boldsymbol{x}) \approx f(\boldsymbol{x}_0) + \nabla f(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_0)^T \boldsymbol{H}_f(\boldsymbol{x}_0)(\boldsymbol{x} - \boldsymbol{x}_0)$$

  where $\boldsymbol{H}_f(\boldsymbol{x}_0)$ is the Hessian of f at $\boldsymbol{x}_0$.
- Apply conjugate gradient method to the quadratic function above successively.
- The computation of β_k needs the formation of Hessian matrix $\boldsymbol{H}_f(\boldsymbol{x}_0)$ which is a formidable task!
- Equivalent transformation in the quadratic case

$$\beta_k = -\frac{(\boldsymbol{r}_{k+1})^T \boldsymbol{A}\boldsymbol{p}_k}{(\boldsymbol{p}_k)^T \boldsymbol{A}\boldsymbol{p}_k} = \frac{\|\nabla\varphi(\boldsymbol{x}_{k+1})\|^2}{\|\nabla\varphi(\boldsymbol{x}_k)\|^2}$$

  This formula does NOT need the computation of Hessian matrix.

Conjugate gradient method for nonlinear optimization

Formally generalize CGM to nonlinear optimization

1. Given initial $\boldsymbol{x}_0$ and $\epsilon > 0$;
2. Compute $\boldsymbol{g}_0 = \nabla f(\boldsymbol{x}_0)$ and $\boldsymbol{p}_0 = -\boldsymbol{g}_0,\ k = 0$;
3. Compute λ_k from

$$\min_{\lambda} f(\boldsymbol{x}_k + \lambda \boldsymbol{p}_k)$$

   and

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \lambda_k \boldsymbol{p}_k, \quad \boldsymbol{g}_{k+1} = \nabla f(\boldsymbol{x}_{k+1})$$

4. If $\|\boldsymbol{g}_{k+1}\| \leq \epsilon$, the iteration is over. Otherwise compute

$$\mu_{k+1} = \frac{\|\boldsymbol{g}_{k+1}\|^2}{\|\boldsymbol{g}_k\|^2}$$

$$\boldsymbol{p}_{k+1} = -\boldsymbol{g}_{k+1} + \mu_{k+1} \boldsymbol{p}_k$$

   Set $k = k + 1$, iterate until convergence.

Conjugate gradient method

In realistic computations, because there is only n conjugate gradient directions for n dimensional problem, it often restarts from current point after n iterations.

Conjugate gradient method

CGM for example

$$\min f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$$

Initial state $\boldsymbol{x}_0 = (-1.2, 1)$.

Variable metric method

- A general form of iterations

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \lambda_k \boldsymbol{H}_k \nabla f(\boldsymbol{x}_k)$$

  1. If $\boldsymbol{H}_k = \boldsymbol{I}$, it is steepest decent method;
  2. If $\boldsymbol{H}_k = [\nabla^2 f(\boldsymbol{x}_k)]^{-1}$, it is dumped Newton's method.

- In order to keep the fast convergence of Newton's method, we hope to approximate $[\nabla^2 f(\boldsymbol{x}_k)]^{-1}$ as $\boldsymbol{H}_k$ with reduced computational efforts as

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \boldsymbol{C}_k,$$

where $\boldsymbol{C}_k$ is a correction matrix which is easily computed.

Variable metric method

- First consider quadratic function

$$f(\boldsymbol{x}) = a + \boldsymbol{b}^T\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^T\boldsymbol{G}\boldsymbol{x}$$

we have

$$\nabla f(\boldsymbol{x}) = \boldsymbol{b} + \boldsymbol{G}\boldsymbol{x}$$

- Define $\boldsymbol{g}(\boldsymbol{x}) = \nabla f(\boldsymbol{x}),\ \boldsymbol{g}_k = \boldsymbol{g}(\boldsymbol{x}_k)$, then

$$\boldsymbol{g}_{k+1} - \boldsymbol{g}_k = \boldsymbol{G}(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k).$$

Define

$$\Delta\boldsymbol{x}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k, \quad \Delta\boldsymbol{g}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k$$

we have

$$\boldsymbol{G}\Delta\boldsymbol{x}_k = \Delta\boldsymbol{g}_k.$$

Variable metric method

- For general nonlinear function

$$f(\boldsymbol{x}) \approx f(\boldsymbol{x}_{k+1}) + \nabla f(\boldsymbol{x}_{k+1}) \cdot (\boldsymbol{x} - \boldsymbol{x}_{k+1}) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_{k+1})^T \boldsymbol{H}_f(\boldsymbol{x}_{k+1})(\boldsymbol{x} - \boldsymbol{x}_{k+1}).$$

  Similar procedure as above we have

$$[\boldsymbol{H}_f(\boldsymbol{x}_{k+1})]^{-1} \Delta \boldsymbol{g}_k = \Delta \boldsymbol{x}_k.$$

- As $\boldsymbol{H}_{k+1}$ is a approximation of $[\boldsymbol{H}_f(\boldsymbol{x}_{k+1})]^{-1}$, it must satisfy

$$\boldsymbol{H}_{k+1} \Delta \boldsymbol{g}_k = \Delta \boldsymbol{x}_k.$$

DFP method

- Davidon-Fletcher-Powell method:
  Choose $\boldsymbol{C}_k$ as rank-2 correction matrix

$$\boldsymbol{C}_k = \alpha_k \boldsymbol{u}\boldsymbol{u}^T + \beta_k \boldsymbol{v}\boldsymbol{v}^T$$

where $\alpha_k, \beta_k, \boldsymbol{u}, \boldsymbol{v}$ are undetermined variables.

- From $\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \boldsymbol{C}_k$ and $\boldsymbol{H}_{k+1}\Delta\boldsymbol{g}_k = \Delta\boldsymbol{x}_k$ we have

$$\alpha_k \boldsymbol{u}(\boldsymbol{u}^T\Delta\boldsymbol{g}_k) + \beta_k \boldsymbol{v}(\boldsymbol{v}^T\Delta\boldsymbol{g}_k) = \Delta\boldsymbol{x}_k - \boldsymbol{H}_k\Delta\boldsymbol{g}_k$$

- Take $\boldsymbol{u} = \boldsymbol{H}_k\Delta\boldsymbol{g}_k, \quad \boldsymbol{v} = \Delta\boldsymbol{x}_k$ and

$$\alpha_k = -\frac{1}{\boldsymbol{u}^T\Delta\boldsymbol{g}_k}, \quad \beta_k = \frac{1}{\boldsymbol{v}^T\Delta\boldsymbol{g}_k}$$

We obtain the famous DFP method

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k - \frac{\boldsymbol{H}_k\Delta\boldsymbol{g}_k\Delta\boldsymbol{g}_k^T\boldsymbol{H}_k}{\Delta\boldsymbol{g}_k^T\boldsymbol{H}_k\Delta\boldsymbol{g}_k} + \frac{\Delta\boldsymbol{x}_k\Delta\boldsymbol{x}_k^T}{\Delta\boldsymbol{x}_k^T\Delta\boldsymbol{g}_k}$$

Remark on DFP method

- If $f(\boldsymbol{x})$ is quadratic and $\boldsymbol{H}_0 = \boldsymbol{I}$, then the result will converge in n steps theoretically;
- If $f(\boldsymbol{x})$ is strictly convex, the DFP method is convergent globally.
- If $\boldsymbol{H}_k$ is SPD and $\boldsymbol{g}_k \neq 0$, then $\boldsymbol{H}_{k+1}$ is SPD also.

DFP method

DFP method for example

$$\min f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$$

Initial state $\boldsymbol{x}_0 = (-1.2, 1)$.

BFGS method

- The most popular variable metric method is BFGS (Broyden-Fletcher-Goldfarb-Shanno) method shown as below

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k - \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}_k \Delta \boldsymbol{g}_k^T \boldsymbol{H}_k}{\Delta \boldsymbol{g}_k^T \boldsymbol{H}_k \Delta \boldsymbol{g}_k} + \frac{\Delta \boldsymbol{x}_k \Delta \boldsymbol{x}_k^T}{\Delta \boldsymbol{x}_k^T \Delta \boldsymbol{g}_k} + (\Delta \boldsymbol{g}_k^T \boldsymbol{H}_k \Delta \boldsymbol{g}_k) \boldsymbol{v}_k \boldsymbol{v}_k^T$$

where

$$\boldsymbol{v}_k = \frac{\Delta \boldsymbol{x}_k}{\Delta \boldsymbol{x}_k^T \Delta \boldsymbol{g}_k} - \frac{\boldsymbol{H}_k \Delta \boldsymbol{g}_k}{\Delta \boldsymbol{g}_k^T \boldsymbol{H}_k \Delta \boldsymbol{g}_k}$$

- BFGS is more stable than DFP method;
- BFGS is also a rank-2 correction method for $\boldsymbol{H}_k$.

Nonlinear least squares

- Mathematically, nonlinear least squares is to minimize

$$\phi(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{r}(\boldsymbol{x})^T\boldsymbol{r}(\boldsymbol{x})$$

- We have

$$\nabla\phi(\boldsymbol{x}) = \boldsymbol{J}^T(\boldsymbol{x})\boldsymbol{r}(\boldsymbol{x}), \quad \boldsymbol{H}_\phi(\boldsymbol{x}) = \boldsymbol{J}^T(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x}) + \sum_{i=1}^{m} r_i(\boldsymbol{x})\boldsymbol{H}_{r_i}(\boldsymbol{x})$$

  where $\boldsymbol{J}(\boldsymbol{x})$ is the Jacobian matrix of $\boldsymbol{r}(\boldsymbol{x})$.

- Direct Newton's method for increment $\boldsymbol{s}_k$ in nonlinear least squares

$$\boldsymbol{H}_\phi(\boldsymbol{x}_k)\boldsymbol{s}_k = -\nabla\phi(\boldsymbol{x}_k)$$

Gauss-Newton method

- If make the assumption that the residual $r_i(\boldsymbol{x})$ is very small, we will drop the term $\sum_{i=1}^{m} r_i(\boldsymbol{x})\boldsymbol{H}_{r_i}(\boldsymbol{x})$ in Newton's method and we obtain Gauss-Newton method

$$(\boldsymbol{J}^T(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k))\boldsymbol{s}_k = -\nabla\phi(\boldsymbol{x}_k)$$

- Gauss-Newton method is equivalent to solve a sequence of linear least squares problems to approximate the nonlinear least squares.

Levenberg-Marquardt method

- If the Jacobian $\boldsymbol{J}(x)$ is ill-conditioned, one may take the Levenberg-Marquardt method as

$$(\boldsymbol{J}^T(\boldsymbol{x}_k)\boldsymbol{J}(\boldsymbol{x}_k) + \mu_k \boldsymbol{I})\boldsymbol{s}_k = -\nabla\phi(\boldsymbol{x}_k)$$

  where μ_k is a nonnegative parameter chosen by some strategy.
- L-M method may be viewed as a regularization method for Gauss-Newton method.

Homework assignment

Newton's method and BFGS method for example

$$\min f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (x_1 - 1)^2$$

Initial state $\boldsymbol{x}_0 = (-1.2, 1)$.

References

1. 唐焕文，秦学志，实用最优化方法，大连理工大学出版社，第三版，2004。
2. J.F. Bonnans et al., Numerical optimization: Theoretical and practical aspects, Universitext, Springer, Berlin, 2003.