# Math Homework Week #7, Unconstrained Nonlinear Optimization

OSM Lab, Eun-Seok Lee

1. (9.3) "I refered textbook and Wikipedia."

   * Gradient descent
   (i) Basic Idea
   If objective function is differentiable, gradient of Df is the direction of greatest increase of f. and -Df is the direction of greatest decrease. It is also called as steepest descent". Basically, it is the method for finding local minimum. It can be applied to linear system and also nonlinear system. It can be viewd as kind of Euler's method.

   (ii) Which Situation
   For non-differentiable function, it should not be used. However, for differentiable multivariate function, it can be used.

   (iii) Strength
   It works in high dimension, and also in infinite dimension.

   (iv) Weakness
   For specific problem, it is very slow because it goes by zigzag way. For non-differentiable function, it can be wrongly defined.

   * Newton and Quasi-Newton Method
   (i) Basic Idea
   It is used by successive approximation. ($x_{n+1} = x_n - \frac{f(x_n}{f'(x_n)}$). Quasi-Newton can be used if Jacobian and Hessian is not available. Idea is very simple. by tangent line and lnitial guess, it iterates for several times until it converges. It is used for finding max or min of problems.

   (ii) Which Situation
   Sometimes, it diverges like the the function $f(x) = \|x\|^a, 0 < a < \frac{1}{2}$. In normal smooth function, it can be used very easily.

   (iii) Strength
   It is very easy to code. Also in normal differentiable polynomial, it is very efficient. Also, it works in high dimension.

   (iv) Weakness
   If the function is not differentiable, it could diverge. Also, with poor initial guess, sometimes it goes to wrong way. For certain cases, it goes very slowly.

* Conjugate Gradient method
(i) Basic Idea
They never compute n by n matrix but picking some impotant information and save calculation burden. It is very useful for quadratic optimization problems. Each step of the conjugate gradient method in this situation has both a temporal and spatial complexity of O(m), where m is the number of nonzero entries in Q.

(ii) Which Situation
When the dimension of matrix is very large to solve directly, it can be used. (For example, sparse matrix.) If we calculate correctly, it can save time.

(iii) Strength
As explained above, for specific problems, we can get the solution very fast compared to other methods.

(iv) Weakness
Concepts and implemetion of coding are not very easy compared to other methods. Also, sometimes it is not stable when we examine it by perturbation.

2. (9.6)
Please see the 9.6.py file.

3. (9.7)
Please see the 9.7.py file.

4. (9.10)
$Df(x_k) = Qx_k - b$
$D^2 f(x_k) = Q$

$x_{k+1} = x_k - (D^2 f(x_k))^{-1} Df(x_k)^T$
$= x_k - Q^{-1}(Qx_k - b)$
$= x_k - x_k + Q^{-1}b$
($\because Q$ is symmetric)$= Q^{-1}b$

From $Df(x_k) = 0$, ($x_k = Q^{-1}b$) is a minimizer in a reason that ($D^2 f(x_k) = Q$) is positive definite.

# References