Homework 3, due February 15

In this homework, we solve an unconstrained optimization problem by using the steepest descent method and the Newton type methods, all combined with line search.

Here we solve the minimum of the Rosenbrock function $f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$.

1. Compute the gradient $\nabla f(\mathbf{x})$ and the Hessian $\nabla^2 f(\mathbf{x})$ of the Rosenbrock function

$$f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Prove (by paper and pen) that $\mathbf{x}^* = (1,1)^T$ is a local minimum of this function.

Find this minimum by using the Matlab built-in optimization function *fminsearch*.

2. Write a Matlab function, *function [f, grad, gradTwo] = Rosen(**x**)*, which returns the Rosenbrock function value $f(\mathbf{x})$ to f, its gradient vector $\nabla f(\mathbf{x})$ to $grad$, and its Hessian matrix $\nabla^2 f(\mathbf{x})$ to $gradTwo$, at the given input point $\mathbf{x}$.

Verify that this function returns correct values.

3. Write the two line search functions *linesch.m* and *lszoom.m*, (see algorithms in the lecture note). **Pay attentions** to the inputs, outputs, and connections of these two functions to make the line search work. Take $c_1 = 10^{-4}$ and $c_2 = 0.9$, in the line search algorithm.

This line search functions will be called later to determine the step size at each iteration.

4. Write four functions, corresponding to the steepest descent method, the original Newton's method, the modified Newton's method with Hessian modification, and the BFGS algorithm, respectively, to solve the minimum of Rosenbrock function; see the lecture note for each algorithm.

At each iteration, given the current iterate $\mathbf{x}_k$ and the direction $\mathbf{p}_k$, use the line search functions in Question **3** to determine the step size α_k.

Stop the iteration when the norm of the gradient at the iterate is less than 10^{-6}. In the steepest descent method, allow as many as 20000 iterations to achieve convergence.

For the parameter δ in the Hessian modification approach, you can try, e.g., $\delta = 10^{-6}$.

Test each method by using different initial guesses. Do they always converge to the minimum of the Rosenbrock function? From your experiments, which algorithm is most robust?

5. Plot the contour lines of the Rosenbrock function by using Matlab function *contour* (first need to call the function *meshgrid* to generate a mesh; use Matlab help for more information).

For each method, take the initial guess as $\mathbf{x}_0 = (x_1, x_2) = (-2, -5)$, and plot the trajectory of the iterate sequence $\{\mathbf{x}_k\}$, on the contour plot of the Rosenbrock function.

Among the four studied methods, which trajectory is more oscillatory?