Exercises for Week 4

1 Quasi-Newton Conditions

1.1 (by hand)

Let $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$ and $\boldsymbol{p}_k$ be a descent direction. Show that the second strong Wolfe condition

$$|\nabla f(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k)^T \boldsymbol{p}_k| \leq c_2 |\nabla f(\boldsymbol{x}_k)^T \boldsymbol{p}_k|$$

implies the curvature condition

$$(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)^T (\nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k)) > 0.$$

1.2 (by hand)

Prove that the SR1 and DFP inverse Hessian update formula satisfy the secant equation.

2 BFGS method

2.1 (by hand)

Consider the quadratic minimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^2} f(\boldsymbol{x}) = x_1^2 - 2x_1 x_2 + 4x_2^2.$$

Set the starting point at $\boldsymbol{x}_0 = [4, -1]^T$.

1. Using a calculator (or a computer, as you prefer) to execute one iteration of the steepest descent method with exact line search. Show $\boldsymbol{x}_1$ and α_1.

2. Calculate the search direction for the first iteration of Newton's method. Does the minimizer $\boldsymbol{x}^* = \mathbf{0}$ on this search direction? Why?

3. Compute the search direction for the second iteration of the quasi-Newton method using the BFGS update with the initial inverse Hessian approximation $H_0 = I$. Note that the first iteration will just be the steepest descent, so take advantage of the work you did for the question. Does the search direction go through the minimizer? Why?

2.2 (in Matlab)

Consider the minimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^2} f(\boldsymbol{x}) = x_1^4 - 2x_1^2x_2 + x_1^2 + x_2^2 - 2x_1 + 5.$$

Set the starting point at $\boldsymbol{x}_0 = [1, 4]^T$.

1. Referring to the Matlab function `steepestdescent_line`, write a Matlab function `BFGSmethod_line` to implement BFGS method, which is given in the textbook page 140, with backtracking line search. We set initial inverse Hessian approximation $H_0 = I$.
2. Run 8 iterations on the above problem with the given starting point. Save $\boldsymbol{x}_k$ and $f(\boldsymbol{x}_k)$, and output them as a table.
3. Plot all $\boldsymbol{x}_k$ with $k = 0, \dots, 8$ in the contour plot of f.
4. Run 8 iterations with the same starting point of the steepest descent method (with backtracking line search) and Newton's method (with fixed step length 1). Output $\boldsymbol{x}_k$ and $f(\boldsymbol{x}_k)$ in the same table.
5. Comment on and explain the results of all three methods.

3 `fminunc` in Matlab's Optimization Toolbox

1. Read about `fminunc` using `doc fminunc`.
2. Read about optimization options using `doc optimoptions`.
3. See the options for `fminunc` by typing `options = optimoptions('fminunc')` in the command window.
4. Figure out how to supply the gradient, $\nabla f(\boldsymbol{x})$, to `fminunc`.
5. Use `fminunc` to solve the minimization problem in Question 2.2 with the starting point $\boldsymbol{x}_0 = [1, 4]^T$. In `fminunc` set the maximum number of iterations as 8, and display output at each iteration. Compare two algorithms, "trust-region" and "quasi-Newton", in `fminunc`, and which one gives better results?