CENG 216 – NUMERICAL COMPUTATION
Homework 3

April 26, 2017

Due Date: May 10, 2017

Exercise 1 General Unconstrained Optimization

a. **(Nocedal-Wright)** The Rosenbrock function is given as

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

Compute the gradient $\nabla f(\tilde{\mathbf{x}})$ and the Hessian

$$\nabla^2 f(\tilde{\mathbf{x}}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}$$

and show that $\tilde{\mathbf{x}}_* = (1, 1)$ is a local minimizer of this function (You need to show that the gradient is zero and the Hessian is positive definite at this point).

b. **(Nocedal-Wright)** Given the vector $\tilde{\mathbf{a}} \in \mathbb{R}^n$ and the symmetric matrix $A \in \mathbb{R}^{n \times n}$, compute the gradients and Hessian matrices of $f_1(\tilde{\mathbf{x}}) = \tilde{\mathbf{a}}^\top \tilde{\mathbf{x}}$ and $f_2(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^\top A \tilde{\mathbf{x}}$

c. **(Nocedal-Wright)** Consider the function $f(x_1, x_2) = (x_1 + x_2^2)^2$. At the point $\tilde{\mathbf{x}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, we can define a search direction $\tilde{\mathbf{p}} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$. Find all the minimizers of f in the direction $\tilde{\mathbf{p}}$

$$\min_{\alpha > 0} f(\tilde{\mathbf{x}} + \alpha \tilde{\mathbf{p}}).$$

Hint: Substitute $x_1 = 1 + \alpha \cdot (-1), x_2 = 1 + \alpha \cdot 1$ for the function f. Differentiate w.r.t α and set the result to zero to find the critical points along the search direction, evaluate the function at $\alpha = 0$ and the critical points to prove that these are minimizers.

Exercise 2 Gradient Descent

Write a C++ program that implements Gradient Descent for a function $f : \mathbb{R}^2 \to \mathbb{R}$. Use this program to minimize the Rosenbrock function (defined above) with initial step size $\alpha_0 = 10^{-3}$ starting from the point

$$\tilde{\mathbf{x}}_0 = \begin{bmatrix} 1.2 \\ 1.2 \end{bmatrix}.$$

At every 100 steps k, print $\tilde{\mathbf{x}}_k$, $f(\tilde{\mathbf{x}}_k)$, and $\nabla f(\tilde{\mathbf{x}}_k)$. Terminate if any one of these criteria is satisfied

- The function value decreases less than ϵ_1 between two consecutive iterations.
- The norm of the gradient vector is less than ϵ_2.
- The number of iterations reaches `iter_max`.

Experiment with different values of ϵ_1, ϵ_2, and `iter_max`.

Minimize the same function starting from the more difficult point

$$\tilde{\mathbf{x}}_0 = \begin{bmatrix} -1.2 \\ 1.0 \end{bmatrix}.$$

Adjust the parameters ϵ_1, ϵ_2, and `iter_max` accordingly.

When we run the submitted homework, we should see the iteration for both runs and a final message for each run indicating where the iterations stopped and because of which of the above conditions.

BONUS (10 pts): Optionally, implement the same program in Python using the `numpy` for numerical computations.

NOTE: There will be a small bonus for those submitting a homework prepared in LaTeX.