HW2- Gradient Descent method and Newton method

Task 1 – Convex sets and functions

Q1

Show that if f_1 and f_2 are convex functions on a convex domain C, $g(x) = \max_{i=1,2} f_i(x)$ is a convex function.

Q2

Let $f(x)$ be a convex function defined over a convex domain C. Show that the level set $L = \{x \in C : f(x) \leq \alpha\}$ is convex.

Q3

Let $f(x)$ be a smooth and twice differentiable convex function. Show that $g(x) = f(Ax)$ is convex, were A is a matrix of appropriate size. Check positive semi-definiteness of Hessian.

Q4

Phrase and prove Jensen's inequality for the discrete case.

Q5

Using Jensen inequality, prove arithmetic geometric mean inequality

$$\frac{x_1 + x_2 + \cdots + x_n}{n} \geq \sqrt[n]{x_1 \cdot x_2 \cdots x_n}$$

Task 2 – Gradient Descent Analytical Convergence

Q6

Let $f(x = \frac{1}{2}x^T Qx - b^T x + c$ be the function to minimize, where $Q \succcurlyeq 0$ is symmetric.

A. We will define the condition number of positive definite matrix A as $\theta \triangleq \frac{\lambda_{max}}{\lambda_{min}}$. Write an upper bound on the convergence ratio β that we found in the tutorial, using $\theta(Q)$ - the condition number of Q.
B. Assume that the step size can be modified at any iteration. Find the optimal step size α_k^* .
hint: Formulate a 1D optimization problem and solve it analytically.

Q7

Let there be a strongly convex function $f(x)$.

Prove that if $\forall\, x \in\ Dom(f)$: $mI \succcurlyeq\ \nabla^2 f(x) \succcurlyeq MI$ then:

$$\frac{1}{2m}\left|\left|\nabla f(x)\right|\right|_2^2 \leq f(x) - f(x^*) \leq \frac{1}{2M}\left|\left|\nabla f(x)\right|\right|_2^2$$

Hint 1: Use Taylor's multivariate theorem:

$$\forall\, x, y \in \mathbb{R}^n\ \exists\, z \in [x, y]: \quad f(y) = f(x) + \nabla f(x)^T (y - x) + (y - x)^T \nabla^2 f(z)(y - x)$$

Hint 2: Find a boundary on the expression. Minimize both sides of the equation by comparing the gradient in regards to y to zero, and then substituating the result.

Task 3 – Gradient Descent and and Newton Method Implementation

In this task you will implement the gradient descend and the Newton methods, together with Armijo inexact line search for the step size calculation. Use the following parameters for all the exercises:

Initial step size: $\alpha_0 = s = 1$

$\sigma = 0.25$

$\beta = 0.5$

Stopping criterion: $\|\nabla f(x_k)\| < \varepsilon$ *with* $\varepsilon = 10^{-5}$

Remarks:

- For the Newton method, use LDL decomposition to find the Newton direction, i.e., to solve the following set of linear equations: $\nabla^2 f(x) d_{newton} = -\nabla f(x)$.
  Use forward/back substitution to solve this problem – do not use built in functions to invert the Hessian! To perform LDL decomposition, use the attached 'mcholmz' function.
- For all your calculations, use only analytic forms of the gradients/Hessians. Before the optimization process, test the correctness of your expressions with numerical differentiation.
- Submit all your analytical calculations and the graphs in a PDF file. Provide short description/explanation to all the graphs.
- Submit your code.

Q8

1. Given the Rosenbrock function:

$$f(\mathbf{x}) = f(x_1, x_2, \dots, x_N) = \sum_{i=1}^{N-1} [(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2]$$

   Derive the analytical expressions of its gradient and Hessian.

2. Use the gradient descend method to find the optimal point of the Rosenbrock function.
   - Starting point: $x_0 = (0, \dots, 0),\ N = 10.$
   - Plot the convergence curve: $f(x_k) - f^*$ as a function of the iteration number k, where f^* is the optimal value of the Rosenbrock function. Use logarithmic scale for the y-axis.

3. Use the gradient descend method to optimize the following quadratic function:

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathbf{T}}\mathbf{H}\mathbf{x}$$

   - Check two cases: well-conditioned and ill-conditioned. The Hessian matrices for both cases are in the attached MATLAB file (h.mat, H_well and H_ill variables). Those who use Python can open the file with the help of scipy library (scipy.io.loadmat).
   - The starting point is in the h.mat file (x_0 variable).
   - Plot the convergence curve: $f(x_k) - f^*$ as a function of the iteration number k, where f^* is the optimal value of the quadratic function. Use logarithmic scale for the y-axis.

Q9

Repeat the previous task (T1.2 and T1.3), but now use the Newton method. On the plots, mark the point where the convergence rate changes from linear to quadratic. Do all the graphs have this transition?