CS257 Linear and Convex Optimization

Homework 9

Due: November 16, 2020

November 23, 2020

For this assignment, you should submit a report in pdf format as well as your source code (.py files). The report should include all necessary figures, the outputs of your Python code, and your answers to the questions. Do NOT write your answers in any of the .py files.

1. Consider the optimization problem $\min_x f(x)$, where $f : \mathbb{R} \to \mathbb{R}$ is given by $f(x) = (x-a)^4$, and $a \in \mathbb{R}$ is a constant.

(a). Find an explicit expression for the Newton step.

(b). Let x_k be the sequence of iterates generated by Newton's method. Let $y_k = |x_k - a|$ be the error between the k-th iterate and the optimal solution. Show that $y_{k+1} = \frac{2}{3} y_k$

(c). Conclude $|x_k - a|$ decays to zero exponentially, i.e. x_k converges exponentially to a.

2. Consider $\min_x f(x)$ for $f(x) = x - \log x$, where log denotes the natural logarithm. This is an unconstrained problem, but the domain of f is $\operatorname{dom} f = (0, +\infty)$.

First implement the function `newton_damped` in `algo9.py`, then complete the code in `hw9.py`.

(a). Find the exact optimal solution and optimal value analytically.

(b). Find an explicit expression for the Newton step.

(c). Run the Newton step you find in (b) in for 5 iterations, using the initial points $x_0 = 0.5$, 1.5 and 2.5. Show the values of x_k for each iteration. What do you observe?

(d). Let $y_k = x^* - x_k$, where x^* is the optimal solution you find in (a). Show $y_{k+1} = y_k^2$ and hence $y_k = y_0^{2^k}$. Conclude that $x_k \to x^*$ if and only if $|x_0 - x^*| < 1$. Is this consistent with what you see in (c)?

(e). Implement and run the **damped Newton's** method using the same initial points $x_0 = 0.5$, 1.5 and 2.5, with $\alpha = 0.1$ and $\beta = 0.7$. Note that you should make sure the point is in the domain before using armijo condition (Hint: you can add the domain check to the WHILE condition for the inner loop in `algo9.py` or let function f(x) in `hw9.py` return positive infinity if the point is not in the domain). Show the values of x_k, plot the error $f(x_k) - f(x^*)$ and the step sizes for each iteration.