CS2601 Linear and Convex Optimization
Homework 7

Due: 2021.11.20

For this assignment, you should submit a **single** pdf file as well as your source code (.py or .ipynb files). The pdf file should include all necessary figures, the outputs of your Python code, and your answers to the questions. Do NOT submit your figures in separate files. Your answers in any of the .py or .ipynb files will NOT be graded.

In this problem, you will implement gradient descent with backtracking line search. First complete the function `gd_armijo` in `gd.py`.

1. Consider the following optimization problem in (9.20) of Boyd and Vandenberghe,

$$\min_{x_1, x_2} f(x_1, x_2) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1} \tag{1}$$

(a). What's the optimal solution $\boldsymbol{x}^*$ and the optimal value $f(\boldsymbol{x}^*)$?

(b). Solve (1) numerically using your implementation of gradient descent with backtracking line search. Use $\alpha = 0.1$ and $\beta = 0.7$ as in Boyd and Vandenberghe. Use the initial points $\boldsymbol{x}_0 = (-1.5, 1)^T$. Report the solution, the number of iterations in the outer loop and the total number of iterations in the inner loop. Plot the trajectory of $\boldsymbol{x}_k$, the error $f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*)$ and the step sizes t_k.

(c). Solve (1) using your implementation of gradient descent with constant step size in HW6. Use $\boldsymbol{x}_0 = (-1.5, 1)^T$. Report the solution and the number of iterations for step sizes 0.1 and 0.01.

(d). Redo part (b) using the initial point $\boldsymbol{x}_0 = (1.5, 1)$.

(e). Redo part (c) using $\boldsymbol{x}_0 = (1.5, 1)^T$ and step size 0.005. Report the solution and the number of iterations. What happens if you use the step sizes in part (c)?

2. Noisy gradient. Consider an m-strongly convex and L-smooth function f with a global minimum $\boldsymbol{x}^*$. Suppose we try to find $\boldsymbol{x}^*$ using gradient descent, but we can only get an approximation to the true gradient in each iteration. More precisely, the approximate gradient we get in the k-th iteration is $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k) + \boldsymbol{\varepsilon}_k$, where $\boldsymbol{\varepsilon}_k$ is some noise term with norm bounded by E, i.e. $\|\boldsymbol{\varepsilon}_k\| \le E$ for all k. Suppose we use constant step size $t \in (0, 1/L]$. Then the gradient step becomes,

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - t\boldsymbol{g}_k = -t(\nabla f(\boldsymbol{x}_k) + \boldsymbol{\varepsilon}_k).$$

(a). Let $\tilde{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_k - t\nabla f(\boldsymbol{x}_k)$, which is the iterate we would get in the k-th iteration if $\boldsymbol{\varepsilon}_k = \mathbf{0}$. Show

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \le \|\tilde{\boldsymbol{x}}_{k+1} - \boldsymbol{x}^*\| + tE.$$

(b). Show

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \le q\|\boldsymbol{x}_k - \boldsymbol{x}^*\| + tE,$$

where $q = \sqrt{1 - mt}$. Hint: Use the result in our analysis for the noiseless case to bound $\|\tilde{\boldsymbol{x}}_{k+1} - \boldsymbol{x}^*\|$.

(c). Show by induction that

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\| \le q^k\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|_2 + \frac{1 - q^k}{1 - q}tE$$

(d). Conclude

$$\limsup_{k\to\infty} \|\boldsymbol{x}_k - \boldsymbol{x}^*\| \le \frac{tE}{1 - q} \le \frac{2E}{m}$$

i.e. $\boldsymbol{x}_k$ "converges" to a ball of radius $\frac{2E}{m}$ centered at $\boldsymbol{x}^*$.