STOR566
Due date: Sep 04, 11:55PM
Keywords: optimization

Homework 1

Introduction to Deep Learning, Fall 2022

Submission Instruction:

- Please answer the questions using the jupyter notebook script.
- For submission, please include your code, code output and answers in the script and submit it on sakai.
- Please don't modify existing cells in the script. But you can add cells between the exercise statements.
- To make markdown, please switch the cell type to markdown (from code) - you can hit 'm' when you are in command mode - and use the markdown language. For a brief tutorial see: https://daringfireball.net/projects/markdown/syntax.

1 Problem 1. (10 points)

Prove whether the following functions are convex or not.

(a) (5 points) $f(x_1, x_2) = (x_1 x_2 - 1)^2$, where $x_1, x_2 \in \mathbb{R}$.

(b) (5 points) $f(\boldsymbol{w}_1, \boldsymbol{w}_2) = \|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_2^2$, where $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^2$.

2 Problem 2. (10 points)

Identify stationary points for $f(x) = 2x_1 + 12x_2 + x_1^2 - 3x_2^2$? Are they local minimum/maximum; global minimum/maximum or saddle points? Why?

3 Problem 3. (80 points)

Given training data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, each $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{+1, -1\}$, we try to solve the following logistic regression problem by gradient descent:

$$\min_{\boldsymbol{w} \in \mathbb{R}^d} \left\{ \frac{1}{n} \sum_{i=1}^n \log(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}) + \frac{1}{2}\|\boldsymbol{w}\|_2^2 \right\} := f(\boldsymbol{w}). \tag{1}$$

Test the algorithm using the "heart scale" dataset with $n = 270$ and $d = 13$: the matrix $\boldsymbol{X}$ is stored in the file "X_heart", and the vector $\boldsymbol{y}$ is stored in the file "y_heart". ("X_heart" contains n lines, each line stores a vector $\boldsymbol{x}_i$ with d real numbers. "y_heart" contains the y vector.)

(a) (5 points) Compute the gradient of $f(\boldsymbol{w})$ w.r.t. $\boldsymbol{w}$.

(b) (30 points) Implement the gradient descent algorithm with a fixed step size η. Find a small η_1 such that the algorithm converges. Increase the step size to η_2 so the algorithm cannot converge. Run 50 iterations and plot the iteration versus $\log(f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*))$ plot for η_1 and η_2. In practice it is impossible to get the exact optimal solution $\boldsymbol{x}^*$, so use the minimum value you computed as $f(\boldsymbol{x}^*)$ when you plot the figure. Report the $f(\boldsymbol{x}^*)$ value you used for generating the plots.

(c) (5 points) Write down the pseudo code of gradient descent with backtracking line search ($\sigma = 0.01$).

(d) (20 points) Implement the gradient descent algorithm with backtracking line search ($\sigma = 0.01$). Plot the same iteration versus $\log(f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*))$ plot.

(e) (20 points) Test your implementation (gradient descent with backtracking line search) on a larger dataset "epsilonsubset". Plot the same iteration vs error plot.