E0 230
Computational Methods in Optimization
Assignment 1

September 15, 2023

Instructions:

- This is an **individual assignment**, and **all work submitted must be your own**!
- Attempt all questions
- You have one week to submit your answers
- All code must be submitted.
- All long answers **must be submitted in a single PDF.**
- For algorithmic questions, a teams form with slots for the answers will be uploaded 24 hours before the submission deadline. If you are asked to provide a number, enter it into the teams form. You answer should be correct to 3 decimal places unless stated otherwise.
- Both your code and your PDF must be submitted in a single zip file, which should be called **student_name_cmo21assn1.zip**.
- Choose the files required for your setup from the concerned directory in the zip file
- For the numericals, you need to call the executables we have created from your script.
- If using python, you can choose use the following command: (Please copy the below carefully and do debugging checks to ensure you're reading the output correctly)
```
someVar = subprocess.run(["filename", "args"],
          stdout=subprocess.PIPE).stdout.decode("utf-8")
```
- In case using linux or mac ensure that you add chmod permissions for executing the file code.
- For mac use the following instructions:
  - Run `chmod 777 <executable name>`
  - You may need to do: `Preferences > Security and Privacy > General` and allow the executable (app) to run (`click "allow"`).
- For MATLAB, use the following instructions:
  - Run `[status,cmdout] = system("filename args")` as described in the problem statement.
  - Convert `cmdout` to floating point and use the output thereof.

1. (10 points) You are given an oracle for a greybox function $f(x) = x^\top Ax + b^\top x$, where $b = [1, 1, 1, 1, 1]^\top$, and $A \in \mathbb{R}^{5\times 5}$ is unknown. Your oracle takes x as an input, and returns $f(x)$ and $\nabla f(x)$.

   (a) (3 points) Using this oracle, estimate the maximum and minimum eigenvalues of the Hessian of $f(x)$; that is, find $\lambda_{\max}(\nabla^2 f(x))$ and $\lambda_{\min}(\nabla^2 f(x))$. Come up with a simple algorithm to find the largest and smallest eigenvalues of A. Note that your algorithm **should not** be of the form

   $$x_{t+1} = g(x_t)$$

   for some function $g(\cdot)$.

   For your solution, you can store at most 20 real numbers at any time. **If your method requires storing more than 20 floating point numbers, you will not receive credit for this problem**. Report your estimates for the optimal values of the maximum and minimum eigenvalues you obtained.

   Solution: Note: $A \neq A^\top$ - you cannot make that assumption based on the information given! We have $H = A + A^\top$. Then

   $$\nabla f(x) = g(x) = Hx + b \Rightarrow x^\top Hx = x^\top (g(x) - b).$$

   Alternatively, you can use the Taylor series expansion around 0:

   $$f(x) = f(0) + \nabla f(0) + \frac{1}{2} x^\top Hx \Rightarrow x^\top Hx = 2(f(x) - b^\top x)$$

   since $f(0) = 0$ and $\nabla f(0) = b$. Then we know that (for instance)

   $$\lambda_{\min} \leq \frac{x^\top Hx}{x^\top x} = \frac{x^\top (g(x) - b)}{x^\top x} \leq \lambda_{\max}.$$

   You can use a brute force approach (say by sampling a large number of vectors in $\mathbb{R}^5$ to estimate the eigenvalues. **If your solution requires you to reveal the Hessian, you will receive 0 credit.**

   (b) (3 points) Repeat the previous problem. However, this time, we will try to use derivative-based optimization to answer this question. Can you derive a suitable cost function $F(x)$ to solve this problem? If not, why? If you can find such a cost function, you can use the following iterative scheme:

   $$y_{t+1} = x_t - \frac{1}{\log(t+2)} \nabla F(x_t), \quad x_{t+1} = \frac{y_{t+1}}{\|y_{t+1}\|_2},$$

   with the same storage constraints as the previous part. Report the optimal values you obtained for the minimum and maximum eigenvalues of the Hessian of $f(x)$, and the number of iterations your algorithm took to reach a point wherein $\|\nabla F(x_T)\|_2 \leq 0.001$. How does your answer compare with the ones you obtained in the previous part?

   Solution: Choose

   $$F(x) = \frac{x^\top Hx}{x^\top x} \quad \nabla F(x) = \frac{2((x^\top x)Hx - (x^\top Hx)x)}{(x^\top x)^2}$$

   and directly implement the algorithm provided. Use the expressions derived above for getting $x^\top Hx$

   (c) (4 points) We now wish to design an iterative algorithm for minimizing $f(x)$. That is, we wish to design an algorithm of the form

   $$x_{t+1} = x_t - \alpha_t d_t.$$

In particular, at each iteration, we will update only one coordinate - that is, $d_t \in \{e_1, \cdots, e_5\}$, where e_i are the coordinate vectors. Thus, at each iteration, our algorithm needs to solve

$$\alpha_t, i_t = \arg\min_{\alpha,i}(f(x_t - \alpha e_i) - f(x_t)).$$

Design an algorithm around these constraints. Starting this algorithm at

$$x_0 = [0,0,0,0,0]^\top,$$

solve

$$x^* = \arg\min f(x).$$

Does this problem have a global minimum? What is x^*? What is the function value at x^*? On this problem, how many iterations does it take for this algorithm to reach stationarity, starting at the given value fo x_0? If you change x_0 to $x_0' \neq x^*$, does the number of iterations change?

Solution: From the Taylor series we get

$$(f(x_t - \alpha e_i) - f(x_t)) = -\alpha e_i^\top \nabla f(x_t) + \frac{\alpha^2}{2} e_i^\top H e_i = -\alpha \nabla_i f(x_t) + \frac{\alpha^2 H_{ii}}{2}$$

Minimizing over α, we get

$$\alpha_t = \frac{\nabla_i(x_t)}{e_i^\top H_{ii} e_i} = \frac{e_i^\top g(x_t)}{e_i^\top (g(e_i) - b)}.$$

Then,

$$f(x_t - \alpha_t e_i) - f(x_t) = -\frac{\nabla_i(x_t)^2}{2H_{ii}} = -\frac{(e_i^\top g(x_t))^2}{2e_i^\top (g(e_i) - b)}$$

Therefore

$$i_t = \arg\max_i \; -\frac{(e_i^\top g(x_t))^2}{2e_i^\top (g(e_i) - b)}$$

The algorithm should converge in 5 steps provided for no i, $(x_0)_i = x_i^*$. If k elements of the initial point are equivalent with x^*, then the algorithm converges in $5 - k$ steps.
Note: you can also use $e_i^\top H e_i = 2(f(e_i) - 1)$.

2. (5 points) Consider the polynomial

$$p(x,y) = x^4y^2 + x^2y^4 - 9x^2y^2.$$

Find $f^* = \inf_{x,y} p(x,y)$. Is $x = [0,0]^\top$ a stationary point, and is it the global minimum? If not, can you identify a global minimum for this function? Is it unique (that is, if x^* is the unique global minimum, then $f(x^*) > f(x)$ for all $x \neq x^*$)?

Solution: Employ the AM-GM inequality

$$\frac{x^4y^2 + x^2y^4 + 27}{3} \geq 3x^2y^2$$

from which we see that the $p^* = -27$. Then, we find the gradient

$$\nabla p(x) = \begin{bmatrix} 4x^3y^2 + 2xy^4 - 18xy^2 \\ 2yx^4 + 4x^2y^3 - 18x^2y \end{bmatrix} = 2xy \begin{bmatrix} 2x^2y + y^3 - 9y \\ x^3 + 2xy^2 - 9x \end{bmatrix}$$

Clearly, $(0,0)$ is a stationary point, but $f(0,0) = 0 > -27$, and is thus not a global minimum. Next, set $|x| = |y| = |t|$, and we have $\nabla p(x) = [3|t|^3 - 9|t|, 3|t|^3 - 9|t|]^\top = [|t|(3|t|^2 - 9), |t|(3|t|^2 -$

9)]$^\top$. Solving for $|t|$, we get $|t| = \sqrt{3}$. Thus, the five stationary points we get are $(0,0)$ $(\sqrt{3}, \sqrt{3})$, $(-\sqrt{3}, \sqrt{3})$, $(\sqrt{3}, -\sqrt{3})$, $(-\sqrt{3}, -\sqrt{3})$. Clearly, $x^* = (\pm\sqrt{3}, \pm\sqrt{3})$ is a minimum as $f(x^*) = -27$ for all four points.

3. (5 points) Consider the function

$$f(x) = e^{x^\top Ax} \frac{e^{-x^\top (B+C)x}}{1 + e^{-x^\top (C-B)x}}.$$

Suppose $\lambda_{\min}(B) = 1$, $\lambda_{\max}(B) = 4$. Can you find the range of values of $\lambda_{\min}(A)$, $\lambda_{\min}(C)$, $\lambda_{\max}(A)$, and $\lambda_{\max}(C)$ such that $f(x)$ is a coercive function?

Solution: We can rewrite the function as

$$f(x) = \frac{e^{x^\top Ax}}{e^{x^\top (B+C)x} + e^{x^\top (B+C)x - x^\top (C-B)x}} = \frac{e^{x^\top Ax}}{e^{x^\top (B+C)x} + e^{2x^\top Bx}} \geq \frac{e^{x^\top Ax}}{2 \max \left(e^{x^\top (B+C)x}, e^{2x^\top Bx}\right)}.$$

For $f(x)$ to be coercive, we need $\lambda_{\min}(A) > \max\{8, \lambda_{\max}(C+B)$, $\lambda_{\max}(A) = \infty$, and there are no bounds on the eigenvalues of C.

4. (5 points) You are each given 100 pairs of data points (x_i, y_i), where $x_i \in \mathbb{R}^5$ and $y_i \in \mathbb{R}$. We know that the data is generated by the equation

$$y_i = w^\top x_i + b.$$

Using the provided data, find w that minimizes the least squares error between y_i and $w^\top x_i + b$. Furthermore, for the general case where $x \in \mathbb{R}^n$ and we are given m data points, what is the closed form solution to this problem? Is this solution unique?

Suppose the number of linearly independent data points is less than n - how would you solve this problem, and is the solution unique?

Enter the value of w obtained with precision upto 2 decimal places, and your long form solutions along with any derivations in the PDF.

Solution:

Define $Y \in \mathbb{R}^n$, where $Y_i = y_i$, and $X \in \mathbb{R}^{m \times n}$ where the ith column of X is x_i. Then, $f(w) = \sum_i (y_i - w^T x_i)^2 = \|Y - Xw\|^2 = w^T X^T X w + Y^T Y - 2Y^T Xw$. The Hessian of $f(w)$ is $X^T X$, which is positive definite if there are m linearly independent data points (in which case, a unique minimum exists), or PSD if there are $r < m$ LI datapoints (in which there may be infinitely many equally good minima). To solve this problem, we set $\nabla f(w^*) = 0 \Rightarrow 2X^T X w^* = 2X^T Y \Rightarrow w^* = (X^T X)^{-1} X^T Y$, or, if $\text{rank}(X) = r < m$, we can use the Psuedoinverse $w^* = (X^T X)^\dagger X^T Y$, where $A^\dagger = V\hat{\Sigma}U^T$, where $\hat{\Sigma}_i i = \frac{1}{\Sigma_i i}$ if $\Sigma_i i \neq 0$.

5. (5 points) You are each given a black box function which returns $f(x)$ and $\nabla f(x)$. Use the following iteration to try and find a minimum:

$$x_{k+1} = x_k - \eta \nabla f(x_k).$$

Starting at $x_0 = [0, 0]$, how many iterations will it take until you reach an ε-approximate point if (a) $\varepsilon = 0.01$, (b) $\varepsilon = 0.001$, for $\eta = 0.8$, $\eta = 0.5$, $\eta = 0.33$, and $\eta = 0.1$? Plot the trajectories of your algorithm on the $x - y$ plane for each stepsize η. Based on your answers can you say anything about the function?

Solution: You need to check $\|\nabla f(x)\| < \varepsilon$. Any other choice would be incorrect, as you do not know whether the function is convex/strongly convex, and thus do not know whether your algorithm will converge to a minimum or a stationary point. You should see slower convergence for $\eta = 0.5,\ 0.1$, faster convergence for $\eta = 0.33$, and the algorithm diverges for $\eta = 0.8$ You cannot say anything about the function based on the convergence of these iterates (thought the function does have a unique global minimum, which is also the single stationary point).