Assignment 3

E0 230: Computational Methods of Optimization

November 2023

No copying allowed. **Thorough plagiarism check will be run.** Refer: CSA misconduct policy
You are given four questions, each of which require python programming.

SUBMISSION INSTRUCTIONS

1. You should submit **two files** (NOT a zip file) with the following naming convention.
   - ▷ `LastFiveDigitsOfSRNumber.pdf` → Answers to all the problems.
   - ▷ `CMO_2023_A3_LastFiveDigitsOfSRNumber.py` or `CMO_2023_A3_LastFiveDigitsOfSRNumber.m` → Code for the all the problems.

   For example, if the last five digits of your SR Number is 20000, then you should submit two files: `20000.pdf, CMO_2023_A3_20000.py or CMO_2023_A3_20000.m`.
2. **Any deviation from the above rule will incur serious penalty!**
3. All the code must be written in Python (3.10 or higher) OR MATLAB. All code written in Jupyter notebooks must be converted to `.py` format before submitting.
4. For the coding questions, you are asked to report some values, e.g., the number of iterations. These values should be reported in the `.pdf` file that you submit.
5. At the top of the `.pdf` file that you submit, write your name and SR Number.
6. You will get a bonus of 10% if your reports are typed neatly in LaTeX

ORACLE ACCESS INSTRUCTIONS

1. 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")`
2. In case using linux or mac ensure that you add chmod permissions for executing the file code.
3. 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"`).
4. 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 Quasi Newton

(5 marks)

Consider a twice differentiable function $f : \mathbb{R}^2 \to \mathbb{R}$. You have an oracle that provides $f(x)$ and $\nabla f(x)$ for an input x. We provide oracles for the function value and gradient for the function in question. You must implement quasi newton algorithms to minimize f and compare it with gradient descent.

1. Run a Quasi-Newton algorithm with Rank-1 updates to minimize f.
2. Run a Quasi-Newton algorithm with Rank-2 updates(BFGS) to minimize f.
3. Run a gradient descent to minimize f

Plot $||\nabla f(x)||$ for each of the algorithms mentioned in the previous parts and compare them. You must find the appropriate stopping criterion and step sizes. Your report must include both these choices and reasoning for choosing them. You must also include the aforementioned plots. **Note all observations in the report**. Can you make connections to something you have seen before?

> Instructions: Call the function through your script as:
> $ ./getGradient SRN x_1 x_2
>
> You will be returned the function value and gradient as follows:
> $f(x_1, x_2), [\nabla f(x)_1, \nabla f(x)_2]$
>
> For example:
> $ ./getGradient 10898 10 10
> $2.30, [-3.53, 0.63]$

2 Linear Programming and Duality

(15 marks)

1. (2 marks) Consider the set
$$S = \{x \in \mathbb{R}^n : \ Ax \leq b\}$$
where A and $b \in \mathbb{R}^m$, $m \neq n$ are constant. Prove that either $S \equiv \emptyset$ or there exists $y \in \mathbb{R}^m, \ y > 0$ elementwise, such that $A^\top y = 0, \ b^\top y < 0$.

2. (4 marks) Consider the set of inequalities
$$\begin{aligned} &x_2 \leq 1 \\ &-x_2 - \frac{1}{2}x_1 \leq \alpha \\ &\beta x_1 - 2x_2 \leq -1. \end{aligned}$$
For what values of α and β are the inequalities feasible? Justify your answer.

3. (4 marks) Let $C(\alpha, \beta) \subset \mathbb{R}^2$ be the set of $x \in \mathbb{R}^2$ that satisfies the inequalities given in the previous part. Assume $C(\alpha, \beta)$ is nonempty. Let $c = [-1, 1]$. For what values of α, β does the problem
$$\begin{aligned} &\min \ c^\top x \\ &\text{s.t.} \ \ x \in C(\alpha, \beta) \end{aligned}$$
have a solution? Justify your answer.

4. (5 marks) Let $\alpha = -1,\ \beta = 1$, and let $c = [-.25, 1]$.
   - (1 mark) Set up the problem in the standard LP form.
   - (1 marks) Solve this problem using your favourite standard linear programming solver (i.e. `Linprog` or `cvxopt`).
   - (3 marks) Solve this problem using the simplex method. For your the basis of your initial BFS, choose $(1, 2, 3)$, where 1 corresponds to x and 2 corresponds to y.

3 Support Vector machines

(10 marks)

Support vector machines (SVMs) are among the most widely used techniques for classifying data, and are very well studied. The SVM is a linear classifier; that is, we aim to find a function $y = f(x) = \text{sign}(w^T x + b)$. We are given data of the form $\mathcal{D} = \{x_i, y_i\}_N$, where the pair $(x_i, y_i) \in \mathbb{R}^n \times \{-1, 1\}$. To learn a support vector machine, we need to solve the following convex optimization problem.

$$\begin{aligned} w^*, b^* = \arg\min \frac{1}{2}\|w\|^2 \\ \text{s.t. } y_i(w^T x_i + b) \geq 1 \quad \text{for } i = 1, ..., N \end{aligned}$$

1. (3 marks) Write a program to solve the **dual** problem using `cvxopt` if you are using python or `quadprog` if you are using matlab.
2. (5 marks) Implement projected gradient descent to solve the dual formulation. Compare the results with the previous outputs. In your report, derive the projection oracle for the dual. Your report must include the details of your projected gradient descent algorithm including your choice of step size and stopping criterion.
3. (2 marks) Write the indices of the points corresponding to the active constraints **clearly** in your report.

You will find the data and labels in the `Files` tab on the teams page as `data.txt` and `labels.txt`.

4 Active Set method

(10 marks) Consider the following optimization problem

$$\begin{aligned} \arg\min_{x_1, x_2} (x_1 - 1)^2 + (x_2 - 2.5)^2 \\ \text{s.t. } x_1 - 2x_2 + 2 \geq 0 \ (1) \\ -x_1 - 2x_2 + 6 \geq 0 \ (2) \\ -x_1 + 2x_2 + 2 \geq 0 \ (3) \\ x_1 \geq 0 \ (4) \\ x_2 \geq 0 \ (5) \end{aligned}$$

Write a program to solve the above optimization problem using the active set algorithm. You are allowed to hard code the constraints for solving the equality problem. Run upto 10 iterations of active set starting with initial point $x_0 = (2, 0)$ and the following initial working sets

1. $\emptyset$
2. $\{3\}$
3. $\{5\}$
4. $\{3, 5\}$

Your report must include the working set at each iteration for each of these initial sets.