Numerical Optimization, Fall 2024
Homework 6

Due 23:59 (CST), Dec. 1, 2024

Problem 1: Projection Calculations [25pts]

Compute the projection of a point onto the following sets:

1. Projection onto the L_2 ball:

$$\min_x \|x - c\|_2^2 \quad \text{s.t.} \quad \|x\|_2 \le r. \tag{1}$$

$$x = \begin{cases} c, & \text{if } \|c\|_2 \le r, \\ r\frac{c}{\|c\|_2}, & \text{if } \|c\|_2 > r. \end{cases} \tag{2}$$

2. Projection onto the L_∞ ball:

$$\min_x \|x - c\|_2^2 \quad \text{s.t.} \quad \|x\|_\infty \le r. \tag{3}$$

$$x_i = \begin{cases} c_i, & \text{if } |c_i| \le r, \\ r \cdot \text{sign}(c_i), & \text{if } |c_i| > r. \end{cases} \tag{4}$$

3. Projection onto a hyperplane:

$$\min_x \|x - c\|_2^2 \quad \text{s.t.} \quad a^\top x = b. \tag{5}$$

$$x = c - \frac{a^\top c - b}{\|a\|_2^2} a. \tag{6}$$

4. Projection onto a half-space:

$$\min_x \|x - c\|_2^2 \quad \text{s.t.} \quad a^\top x \le b. \tag{7}$$

$$x = \begin{cases} c, & \text{if } a^\top c \le b, \\ c - \frac{a^\top c - b}{\|a\|_2^2} a, & \text{if } a^\top c > b. \end{cases} \tag{8}$$

5. Projection onto the intersection of hyperplanes (assume $A^\top A$ is invertible):

$$\min_x \|x - c\|_2^2 \quad \text{s.t.} \;\; Ax = b. \tag{9}$$

$$x = c - A^\top (AA^\top)^{-1}(Ac - b). \tag{10}$$

Problem 2: Frank-Wolfe Subproblem Calculations [15pts]

Solve the Frank-Wolfe subproblem for the following constraint sets:

1. L_1 ball:

$$\min_s \nabla f(x)^\top s \quad \text{s.t.} \;\; \|s\|_1 \le r. \tag{11}$$

The optimal solution occurs at one of the extreme points of the L_1 ball. Hence:

$$s = -r \cdot \text{sign}(\nabla f(x)) \cdot e_i, \tag{12}$$

where e_i is the unit vector corresponding to the largest absolute component of $\nabla f(x)$.

2. L_2 ball:

$$\min_s \nabla f(x)^\top s \quad \text{s.t.} \;\; \|s\|_2 \le r. \tag{13}$$

The solution is given by aligning $-s$ with the gradient direction:

$$s = -r \frac{\nabla f(x)}{\|\nabla f(x)\|_2}. \tag{14}$$

3. L_∞ ball:

$$\min_s \nabla f(x)^\top s \quad \text{s.t.} \;\; \|s\|_\infty \le r. \tag{15}$$

The optimal solution occurs by assigning each s_i to the corresponding gradient component:

$$s = -r \cdot \text{sign}(\nabla f(x)). \tag{16}$$

Problem 3: Write the KKT Conditions [30pts]

Write the Karush-Kuhn-Tucker (KKT) conditions for the following problems and calculate the stationary points:

1. Linear Programming (LP):

$$\begin{aligned} \min_{x} \quad & -2x_1 - 3x_2, \\ \text{s.t.} \quad & x_1 + x_2 \leq 4, \\ & x_1 - 2x_2 \leq 1, \\ & x_1, x_2 \geq 0. \end{aligned} \tag{17}$$

$$\mathcal{L}(x_1, x_2, \lambda_1, \lambda_2, \mu_1, \mu_2) = -2x_1 - 3x_2 + \lambda_1(x_1 + x_2 - 4) + \lambda_2(x_1 - 2x_2 - 1) - \mu_1 x_1 - \mu_2 x_2. \tag{18}$$

1. Stationarity:

$$\frac{\partial \mathcal{L}}{\partial x_1} = -2 + \lambda_1 + \lambda_2 - \mu_1 = 0, \tag{19}$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = -3 + \lambda_1 - 2\lambda_2 - \mu_2 = 0. \tag{20}$$

2. Primal feasibility:

$$x_1 + x_2 \leq 4, \quad x_1 - 2x_2 \leq 1, \quad x_1 \geq 0, \quad x_2 \geq 0. \tag{21}$$

3. Dual feasibility:

$$\lambda_1 \geq 0, \quad \lambda_2 \geq 0, \quad \mu_1 \geq 0, \quad \mu_2 \geq 0. \tag{22}$$

4. Complementary slackness:

$$\lambda_1(x_1 + x_2 - 4) = 0, \quad \lambda_2(x_1 - 2x_2 - 1) = 0, \tag{23}$$

$$\mu_1 x_1 = 0, \quad \mu_2 x_2 = 0. \tag{24}$$

By solving the system of equations, we can find the optimal values of $(x_1, x_2) = (0, 4)$.

2. Quadratic Programming (QP):

$$\begin{aligned} \min_{x} \quad & \frac{1}{2}\left(x_1^2 + 2x_1x_2 + 2x_2^2\right) - 4x_1 - 6x_2, \\ \text{s.t.} \quad & x_1 + x_2 \leq 5, \\ & x_1, x_2 \geq 0. \end{aligned} \tag{25}$$

$$\mathcal{L}(x_1, x_2, \lambda, \mu_1, \mu_2) = \frac{1}{2}(x_1^2 + 2x_1x_2 + 2x_2^2) - 4x_1 - 6x_2 + \lambda(x_1 + x_2 - 5) - \mu_1 x_1 - \mu_2 x_2. \tag{26}$$

1. Stationarity:

$$\frac{\partial \mathcal{L}}{\partial x_1} = x_1 + x_2 - 4 + \lambda - \mu_1 = 0, \tag{27}$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = x_1 + 2x_2 - 6 + \lambda - \mu_2 = 0. \tag{28}$$

2. Primal feasibility:

$$x_1 + x_2 \leq 5, \quad x_1 \geq 0, \quad x_2 \geq 0. \tag{29}$$

3. Dual feasibility:

$$\lambda \geq 0, \quad \mu_1 \geq 0, \quad \mu_2 \geq 0. \tag{30}$$

4. Complementary slackness:

$$\lambda(x_1 + x_2 - 5) = 0, \quad \mu_1 x_1 = 0, \quad \mu_2 x_2 = 0. \tag{31}$$

By solving the system of equations, we can find the optimal values of $(x_1, x_2) = (2, 2)$.

3. Nonlinear Problem (NLP):

$$\begin{aligned} \min_{x} \ & (x_1 - 1)^2 + (x_2 - 2)^2 + x_1 x_2, \\ \text{s.t.} \ & x_1^2 + x_2^2 \leq 4, \\ & x_1 - x_2 = 1. \end{aligned} \tag{32}$$

$$\mathcal{L}(x_1, x_2, \lambda, \mu) = (x_1 - 1)^2 + (x_2 - 2)^2 + x_1 x_2 + \lambda(x_1^2 + x_2^2 - 4) + \mu(x_1 - x_2 - 1). \tag{33}$$

1. Stationarity:

$$\frac{\partial \mathcal{L}}{\partial x_1} = 2(x_1 - 1) + x_2 + 2\lambda x_1 + \mu = 0, \tag{34}$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = 2(x_2 - 2) + x_1 + 2\lambda x_2 - \mu = 0. \tag{35}$$

2. Primal feasibility:

$$x_1^2 + x_2^2 \leq 4, \quad x_1 - x_2 = 1. \tag{36}$$

3. Dual feasibility:

$$\lambda \geq 0, \quad \mu \text{ free}. \tag{37}$$

4. Complementary slackness:

$$\lambda(x_1^2 + x_2^2 - 4) = 0. \tag{38}$$

By solving the system of equations, we can find the optimal values of $(x_1, x_2) = (\frac{3}{2}, \frac{1}{2})$.

Problem 4: Algorithm Selection and Implementation [30pts]

1. Consider the problem:

$$\min_x \|Ax - b\|_2^2 \quad \text{s.t.} \ \|x\|_2 \le r. \tag{39}$$

Analyze whether Gradient Descent (GD) or Frank-Wolfe (FW) is more suitable for this problem. Provide a detailed explanation.

GD is more suitable because the projection onto the L_2 ball is straightforward by normalizing the solution if its norm exceeds r, while FW may be slower due to the iterative linear approximations.

2. Consider the problem:

$$\min_x \|Ax - b\|_2^2 \quad \text{s.t.} \ \|x\|_1 \le r. \tag{40}$$

Analyze whether Gradient Descent (GD) or Frank-Wolfe (FW) is more suitable for this problem. Provide a detailed explanation.

FW is more suitable because it handles the sparse structure naturally by choosing the coordinate with the largest gradient magnitude and updating it, while GD needs expensive projections.

3. Write a program to randomly generate A and b, compute the solutions under both constraints (L_2 and L_1 balls), and compare the performance and results of GD and FW.

```
clc;clear;
m = 100;
n = 50;
A = randn(m, n);
b = randn(m, 1);
r = 1;
maxiter = 10000;
learning_rate = 0.001;
eps1 = 1e-6;
eps2 = 1e-3;

x_gd_l2 = gradient_descent_l2(A, b, r, n, maxiter, learning_rate, eps1);
x_fw_l1 = frank_wolfe_l1(A, b, r, n, maxiter, eps2);

figure;
plot(x_gd_l2, 'r--', 'LineWidth', 2);
hold on;
plot(x_fw_l1, 'k-', 'LineWidth', 2);
legend('GD (L2 ball)', 'FW (L1 ball)');
title('Comparison of GD and FW on L2 and L1 Balls');
```

```
xlabel('Index');
ylabel('Value');
grid on;


function x = gradient_descent_l2(A, b, r, n, maxiter, lr, eps)
    x = zeros(n, 1);
    iter = 0;
    while iter < maxiter
        iter = iter + 1;
        cur_x = x;
        grad = 2 * A' * (A * x - b);
        x = x - lr * grad;
        if norm(x) > r
            x = r * x / norm(x);
        end
        d = norm(x - cur_x)^2;
        if d < eps
            break
        end
    end
    if iter == maxiter
        error("GD_l2 fails to converge!.\n");
    end
end


function x = frank_wolfe_l1(A, b, r, n, maxiter, eps)
    x = zeros(n, 1);
    iter = 0;
    while iter < maxiter
        iter = iter + 1;
        grad = 2 * A' * (A * x - b);
        s = zeros(n, 1);
        [~, idx] = max(abs(grad));
        s(idx) = - r * sign(grad(idx));
        d = (grad'*(s - x))^2;
        gamma = 2 / (iter + 2);
        x = (1 - gamma) * x + gamma * s;
        if d < eps
            break
        end
    end
    if iter == maxiter
        error("FW_l1 fails to converge!.\n");
    end
```

```
    end
```

Submission Requirements

- Submit a PDF file with detailed derivations and explanations.
- Include the program code in the PDF (e.g., Python or MATLAB).