Assignment 6 of MATP482

Xinshi Wang 661975305

Problem 1

Consider the constrained quadratic minimization problem:

$$\underset{\mathbf{x}}{\text{minimize}}\, f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbf{b}^\top \mathbf{x}, \text{ s.t. } l_i \le x_i \le u_i, \forall i = 1, \dots, n, \qquad \text{(C-QuadMin)}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a symmetric positive definite matrix, $\mathbf{b} \in \mathbb{R}^n$, and $\mathbf{l} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^n$ are lower and upper bounds.

1. Let $\mathbf{A} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$, $l_1 = l_2 = 0$, and $u_1 = u_2 = 1$. Set the initial vector $\mathbf{x}^0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. Do by hand two cycles of the alternating minimization method.

   Set $X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, we have

$$\begin{aligned} &\frac{1}{2}x^T Ax - b^T x \\ =&\frac{1}{2}[2x_1 - x_2 - x_1 + 2x_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - 3x_2 \\ =&x_1^2 - x_1x_2 + x_2^2 - 3x_2 \end{aligned}$$

   Let $X^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ 1^{st} cycle:

   $X_1^{(1)} = \underset{x_1}{argmin} f(x_1, x_2^{(0)}) = \underset{x_1}{argmin} x_1^2 = 0$

   $X_2^{(1)} = \underset{x_2}{argmin} f(x_1^{(1)}, x_2) = \underset{x_2}{argmin} x_2^2 - 3x_2 = \frac{3}{2}$

   2^{nd} cycle:

   $X_1^{(2)} = \underset{x_1}{argmin} f(x_1, x_2^{(1)}) = \underset{x_1}{argmin} x_1^2 - \frac{3}{2}x_1 + \frac{9}{4} - \frac{9}{2} \implies 2x_1 - \frac{3}{2} = 0 \implies x_1^{(2)} = \frac{3}{4}$

   $X_2^{(2)} = \underset{x_2}{argmin} f(x_1^{(2)}, x_2) = \underset{x_2}{argmin} \frac{9}{16} - \frac{3}{4}x_2 + x_2^2 - 3x_2 \implies -\frac{3}{4} + 2x_2 - 3 = 0 \implies$

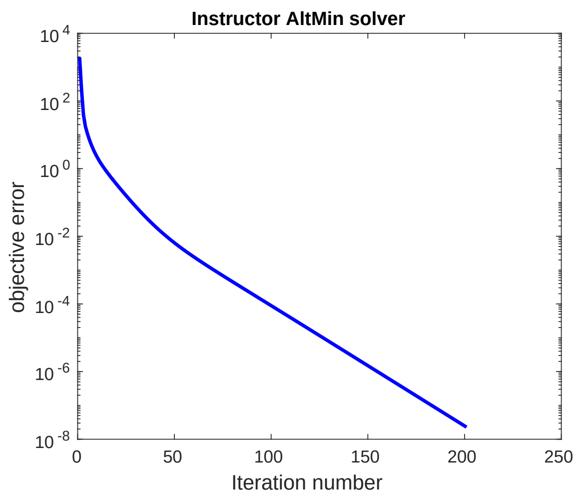
$x_2^{(2)} = \frac{15}{8}$

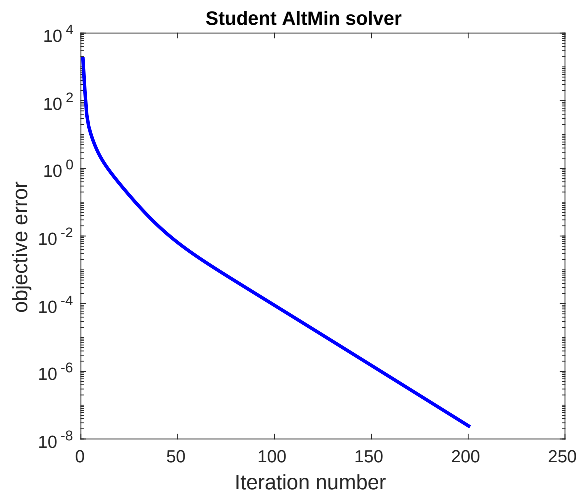
2. [**Bonus question**] Use the instructor's provided file `quadMin_AltMin.m` to write a Matlab function `quadMin_AltMin` of the alternating minimization method for solving (C-QuadMin). The input are $\mathbf{A}$, $\mathbf{b}$, $\mathbf{l}$, $\mathbf{u}$, initial vector $\mathbf{x}0$, and maximum iteration number `maxit`. Also test your function by running the provided test file `test_AltMin.m` and compare to the instructor's AltMin function and PG (projected gradient) function. Print your code and the results you get. What do you observe on the convergence speed of the AltMin and PG methods?

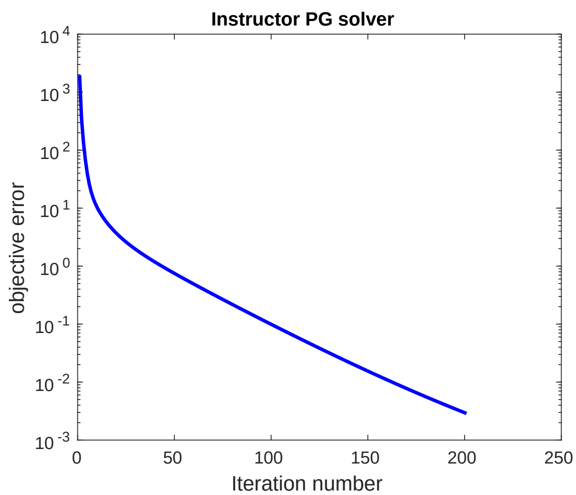
```
function [x, hist_obj] = quadMin_AltMin(A, b, x0, maxit, lb, ub)

% alternating minimization method for solving
% min_x 0.5*x'*A*x - b'*x
% s.t. lb <= x <= ub
x = x0;
% compute the gradient and maintain it
r = A * x - b;
hist_obj = .5 * (x' * (r - b));
n = length(b);
for iter = 1:maxit
    % update all coordinates cyclicly
    for i = 1:n
        % update x(i)
        old_x_i = x(i);
        x(i) = (b(i) - A(i, :) * x + A(i, i) * x(i)) / A(i, i);
        x(i) = max(lb(i), min(ub(i), x(i))); % Project x(i) onto the box constraint

        % update r vector in an efficient way
        delta_x_i = x(i) - old_x_i;
        r = r + A(:, i) * delta_x_i;
    end
    % save objective value after each cycle
    hist_obj = [hist_obj; .5 * (x' * (r - b))];
end
end
```







The objective value for Altmin Method is much lower than the PG method. The objective value for PG method drops quicker in the first place compared to Altmin method.

Problem 2

Consider the problem

$$\min_{x,y\in\mathbb{R}} f(x,y) = \frac{1}{2}x^2 - 2xy + 2y^2 + 4x - 10y, \text{ s.t. } x \le 2, y \le 2.$$

1. Does $(0,0)$ satisfy the optimality condition?

   If (0,0) is a minimizer, we have

   $< \nabla f(0,0), \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \end{bmatrix} >\ge 0, \forall \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} 2 \\ 2 \end{bmatrix}$

   $\nabla f(x,y) = \begin{bmatrix} x - 2y + 4 \\ -2x + 4y - 10 \end{bmatrix}$

   $\nabla f(0,0) = \begin{bmatrix} 4 \\ -10 \end{bmatrix}$

   $4x - 10y \ge 10, \forall \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} 2 \\ 2 \end{bmatrix}$

   We can have a counter example where $x = -5$ and $y = 1$ so $4x - 10y = -30 < 0$. Thus it is not a local minimizer.

2. Show that f is convex on $\mathbb{R}^2$. Find the global minimizer. $\nabla f(x,y) = \begin{bmatrix} x - 2y + 4 \\ -2x + 4y - 10 \end{bmatrix}$

   $\frac{\partial f}{\partial x^2} = 1, \frac{\partial f}{\partial xy} = \frac{\partial f}{\partial yx} = -2, \frac{\partial f}{\partial y^2} = 4.$

   $H = \begin{bmatrix} 1 & -2 \\ -2 & 4 \end{bmatrix}, det(H - \lambda I) = 0 \implies \lambda^2 - 5\lambda = 0 \implies \lambda = 0 \text{ or } \lambda = 5.$

   Therefore the Hessian matrix is positive semi-definite, and as a result the function is convex.

   If $\bar{x} = 2$ and $\bar{y} = 2$

   $$\begin{bmatrix} 2 & -6 \end{bmatrix} \begin{bmatrix} x - 2 \\ y - 2 \end{bmatrix} \ge 0$$
   $$2x - 4 - 6y + 12 \ge 0$$
   $$2x - 6y + 8 \ge 0$$

   When x = 1, y = 10, this equation does not hold.

   If $\bar{x} < 2$ and $\bar{y} = 2$

   $$\begin{bmatrix} \bar{x} & -2\bar{x} - 2 \end{bmatrix} \begin{bmatrix} x - \bar{x} \\ y - 2 \end{bmatrix} \ge 0$$
   $$\bar{x}(x - \bar{x}) + (y - 2)(-2\bar{x} - 2) \ge 0$$

When $\bar{x} = 0$ and $\bar{y} = 2$, this equation must hold.
If $\bar{x} = 2, \bar{y} < 2$

$$\begin{aligned}
\begin{bmatrix} -2\bar{y} + 6 & 4\bar{y} - 14 \end{bmatrix} \begin{bmatrix} x - 2 \\ y - \bar{y} \end{bmatrix} &\geq 0 \\
(x-2)(-2\bar{y}+6) + (4\bar{y}-14)(y-\bar{y}) &\geq 0 \\
2x - 6y + 8 &\geq 0
\end{aligned}$$

When $\bar{x} = 2$ and $\bar{y} = 5$, this equation does not hold.
If $\bar{x} < 2, \bar{y} < 2$

$$\begin{aligned}
\begin{bmatrix} \bar{x} - 2\bar{y} + 4 & -2x + 4\bar{y} - 10 \end{bmatrix} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} &\geq 0 \\
(\bar{x} - 2\bar{y} + 4)(x - \bar{x}) + (-2\bar{x} + 4\bar{y} - 10)(y - \bar{y}) &\geq 0
\end{aligned}$$

When $\bar{x} = 2$ and $\bar{y} = 5$, this equation does not hold.

Therefore the minimizer is Point (0,2).

Problem 3

Consider the problem

$$\min_{x,y\in\mathbb{R}} f(x,y) = -x^2 + 2xy + y^2 - x - 2y$$
$$\text{s.t. } x \geq 0, y \geq 0, x + y \leq 4.$$

In the lecture, we showed that $(1/4, 3/4)$ and $(4, 0)$ are two KKT points. Find all other KKT points if there is any.

1. At Point (0,0)
   $u_3 = 0$

   $$\begin{cases} -2x + 2y - 1 - u_1 = 0 \\ 2x + 2y - 2 - u_2 = 0 \end{cases}$$
   $$\begin{cases} u_1 = -1 \\ u_2 = -2 \end{cases}$$
   Invalid solution.

2. At Y axis($x = 0, y \neq 0$)
   $u_2 = 0, u_3 = 0$ when $y \neq 4$
   if $y \neq 4$
   $$\begin{cases} 2y - 1 - u_1 = 0 \\ 2y - 2 = 0 \end{cases}$$
   $$\begin{cases} y = 1 \\ u_1 = 1 \end{cases}$$
   Point (0,1) is a Valid solution.
   if $y = 4$
   $$\begin{cases} 8 - 1 - u_1 + u_3 = 0 \\ 8 - 2 + u_3 = 0 \end{cases}$$
   $$\begin{cases} u_3 = -6 \\ u_1 = 13 \end{cases}$$
   Invalid solution.

3. At X axis($x \neq 0, y = 0$)
   $u_1 = 0, u_3 = 0$ when $x \neq 4$
   if $x \neq 4$
   $$\begin{cases} -2x - 1 = 0 \implies x = -\frac{1}{2} \\ 2x - 2 - u_2 = 0 \end{cases}$$

Invalid Solution

if $x = 4$

$$\begin{cases} -8 - 1 - u_1 + u_3 = 0 \\ 8 - 2 - u_2 + u_3 = 0 \end{cases}$$

$$\begin{cases} u_3 = 9 \\ u_1 = 3 \end{cases}$$

Valid Solution, duplicate.

4. At the hypotenuse

   $u_1 = 0, u_2 = 0$

   if $y \neq 4$

   $$\begin{cases} -2x + 2y - 1 + u_3 = 0 \\ 2x + 2y - 2 + u_3 = 0 \end{cases}$$

   Since x+y = 4, we have $2 \times 4 - 2 + u_3 = 0 \implies u_3 = -6$ Invalid Solution

Therefore the valid solution is $(0, 1)$

Problem 4

Consider the nonnegative quadratic program:

$$\underset{\mathbf{x}\in X}{\text{minimize}}\, f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\mathbf{x} - \mathbf{c}^\top \mathbf{x}, \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b} \tag{1}$$

where $\mathbf{Q}$ is a symmetric and positive semidefinite matrix, and $X = \{\mathbf{x} \in \mathbb{R}^n : x_i \geq 0, i = 1, \ldots, n\}$.

Use the instructor's provided file `penalty_qp.m` to write a Matlab function `penalty_qp` with input $\mathbf{Q}, \mathbf{c}, \mathbf{A}$, $\mathbf{b}$, initial vector $\mathbf{x}0$, stopping tolerance `tol`, initial penalty parameter μ_0, and the final μ_1. Also test your function by running the provided test file `test_penalty_qp.m` and compare to the instructor's function. Print your code and the results you get.

```
function [x, hist_obj, hist_res] = penalty_qp(Q,c,A,b,tol,mu0,mu1,x0)
% quadratic penalty method for the quadratic programming
% min_x 0.5*x'*Q*x - c'*x
% s.t.  x >= 0, A*x == b

mu = mu0;
x = x0;

% compute the residual for the constraint A*x == b
r = A*x - b;

res = norm(r);
grad_err = 1;
hist_res = res;
hist_obj = 0.5*x'*Q*x - c'*x;

while (res > tol || grad_err > tol) && mu < mu1
    % use constant stepsize
    alpha = 1/norm(Q + mu*A'*A);
    % compute the gradient
    grad = Q*x - c + mu*A'*(A*x - b);

    % compute violation of optimality condition
    grad_err = 0;
    for i = 1:20
        if x(i) == 0
```

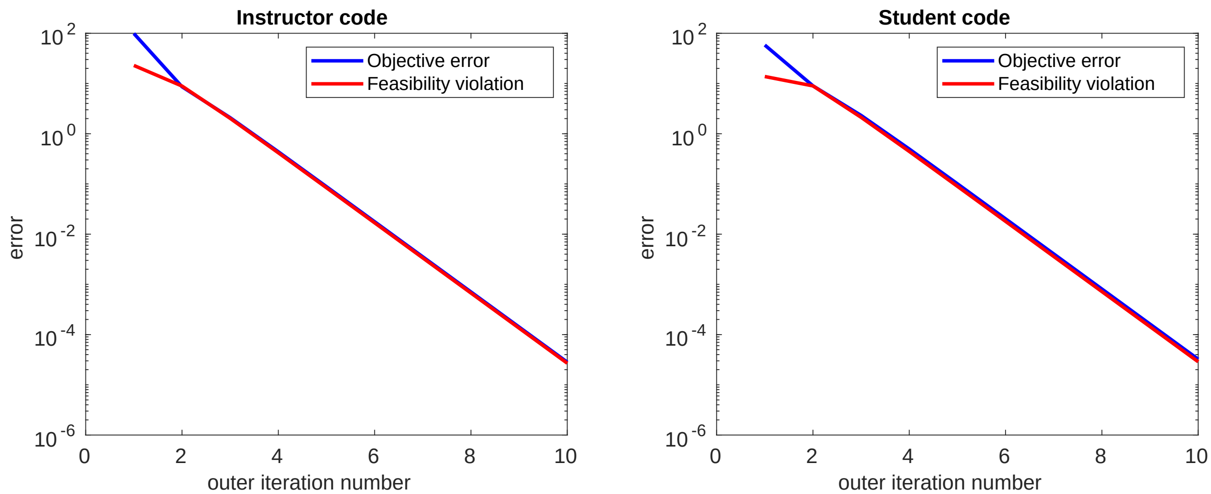
```
            grad_err = grad_err + max(0,-grad(i));
        else
            grad_err = grad_err + abs(grad(i));
        end
    end
    while grad_err > tol
        % update x
        x = x - alpha*grad;
        x = max(x, 0);
        % compute the gradient
        grad = Q*x - c + mu*A'*(A*x - b);
        % compute violation of optimality condition
        grad_err = 0;
        for i = 1:20
            if x(i) == 0
                grad_err = grad_err + max(0,-grad(i));
            else
                grad_err = grad_err + abs(grad(i));
            end
        end
    end
    % compute the residual
    r = A*x - b;
    res = norm(r);
    obj = 0.5*x'*Q*x - c'*x;

    % save res and obj
    hist_res = [hist_res; res];
    hist_obj = [hist_obj; obj];

    % increase the penalty parameter
    mu = 5*mu;
end
end
```

Student solver: Total running time of student code is 16.8549

Instructor solver: Total running time of instructor code is 14.6748



Instructor code
Objective error
Feasibility violation
error
outer iteration number
Student code
Objective error
Feasibility violation
error
outer iteration number