MTH603/AM8211
Fall 2023

Assignment Cover Page

Assignment Number: 2

Student Last Name: Walaa

Student First Name: Mariam

Student ID: 500-744-640

If you used any resources other than what is listed on the course outline, or if you collaborated with any of your colleagues, list below these resources or the names of your colleagues along with their id's:

You are expected to know what constitutes academic integrity, to avoid committing academic offences, and to take responsibility for your actions. Provide your signature below to indicate that you are aware of all policies of the course (described in the course outline), including copyright.

Signature: Mariam Walaa

Assignment 2

Mariam Walaa

October 4, 2023

Question 1

[Question 1] Let S_n be the set of all symmetric, $n \times n$, positive semi-definite matrices. Prove that S_n is convex.

Solution:

To prove that S_n is convex, we must show that for any two matrices A and B in S_n and for any λ in the closed interval $[0, 1]$, the convex combination $\lambda A + (1 - \lambda)B$ also belongs to S_n. In other words, we need to prove that $\lambda A + (1 - \lambda)B$ is symmetric and positive semi-definite for all such A, B, and λ.

Let A and B be two arbitrary symmetric, positive semi-definite matrices in S_n, which means that $A = A^T$ and $B = B^T$, and they satisfy $x^T Ax \geq 0$ and $x^T Bx \geq 0$ for all vectors $x \in \mathbb{R}^n$.

Now, consider the convex combination $\lambda A + (1 - \lambda)B$ for $\lambda \in [0, 1]$. We need to prove that this matrix is also symmetric and positive semi-definite.

1. **Symmetry of S_n:** We want to check that $\lambda A + (1 - \lambda)B = (\lambda A + (1 - \lambda)B)^T$.
We have:

$$\begin{aligned}(\lambda A + (1 - \lambda)B)^T &= \lambda A^T + (1 - \lambda)B^T \\ &= \lambda A + (1 - \lambda)B\end{aligned}$$

We use distributive property of transpose, and the fact that A and B are both symmetric. Thus, $\lambda A+(1-\lambda)B$ is symmetric.

2. **Positive Semi-definiteness:** Let x be an arbitrary vector in $\mathbb{R}^n$. We want to show that

$$x^T(\lambda A + (1 - \lambda)B)x \geq 0$$

.

Using the properties of matrix transpose, we have:

$$x^T(\lambda A + (1 - \lambda)B)x = \lambda x^T Ax + (1 - \lambda)x^T Bx$$

Since both A and B are positive semi-definite, we know that $x^T Ax \geq 0$ and $x^T Bx \geq 0$.
Therefore, $\lambda x^T Ax \geq 0$ and $(1 - \lambda)x^T Bx \geq 0$.
Adding these inequalities together, we get:

$$\lambda x^T Ax + (1 - \lambda)x^T Bx \geq 0$$

This shows that $\lambda A + (1 - \lambda)B$ is positive semi-definite.
Since we have shown that $\lambda A + (1 - \lambda)B$ is both symmetric and positive semi-definite for any $\lambda \in [0, 1]$ and for arbitrary A and B in S_n, we conclude that S_n is convex.
Therefore, the set of all symmetric, $n \times n$, positive semi-definite matrices S_n is indeed convex.

Question 2

[Question 4] Consider an arbitrary quadratic function $f : \mathbb{R}^n \to \mathbb{R}$ in standard form $f(x) = c^T x + \frac{1}{2}x^T Sx$, where S is not positive semi-definite (PSD). Prove that f is unbounded from below by showing that for all $m \in \mathbb{R}^n$, there exists $y \in \mathbb{R}^n$ such that $f(y) < m$.

Solution:

Consider the quadratic function $f(x) = c^T x + \frac{1}{2}x^T Sx$ where S is not positive semi-definite.

Given that S is not PSD, we know that there exists a negative eigenvalue t and a corresponding eigenvector u such that $Su = tu$. We will use this eigenvalue-eigenvector pair to show that for all m, there exists such a t that $f(tu) < m$, demonstrating that f is unbounded from below.

We choose t such that $tu < 0$. This choice is always possible since t is negative.
Consider the vector $x_0 = tu$. We can compute $f(x_0)$ as follows:

$$f(x_0) = c^T x_0 + \frac{1}{2}x_0^T Sx_0 = c^T(tu) + \frac{1}{2}(tu)^T S(tu)$$

Simplify further:

$$f(x_0) = t(c^T u) + \frac{1}{2}t^2(u^T Su)$$

Using the fact that $Su = tu$, we have $u^T Su = u^T(tu) = t(u^T u) = t\|u\|^2$.
Substituting this into the expression for $f(x_0)$:

$$f(x_0) = t(c^T u) + \frac{1}{2}t^2(u^T Su) = t(c^T u) + \frac{1}{2}t^3\|u\|^2$$

Now, we can see that $f(x_0)$ is a cubic polynomial in t with coefficients $c^T u$ and $\frac{1}{2}\|u\|^2$, and it can take any real value depending on the value of t since $c^T u$ and $\|u\|^2$ are constants.
To make $f(x_0)$ less than any given real number m, we can choose scalar t and vector u such that:

$$t(c^T u) + \frac{1}{2}t^3\|u\|^2 < m$$

This is always possible since $c^T u$ and $\|u\|^2$ are constants, and m is a fixed scalar.
This choice of t ensures that $f(x_0) < m$.
Therefore, we have shown that for all m, there exists a negative eigenvalue t and its corresponding eigenvector u such that $f(tu) < m$, demonstrating that f is unbounded from below.

Question 3

[Question 5(a)] Using MATLAB, product $c \in \mathbb{R}^4$ and symmetric $S \in \mathbb{R}^{nxn}$ whose entries are randomly selected integer values on the closed interval, $[-10, 10]$. Then solve the NLP $min_{x \in \mathbb{R}^4} f(x)$, with:

$(a) f(x) = c^T x + \frac{1}{2} x^T S x$

Solution:

```
n = 4;  % Dimension of vector c and size of symmetric matrix S
lower_bound = -10; % Range of random values for elements of c and S
upper_bound = 10;

% Generate random vector c in R^4
c = randi([lower_bound, upper_bound], n, 1);

% Generate random symmetric matrix S with integer values in specified range
S = randi([lower_bound, upper_bound], n, n);
S = triu(S) + triu(S, 1)';  % Ensures symmetry

% Define objective function
f = @(x) c'*x + 0.5*x'*S*x;

% Initial guess for the optimization
x0 = zeros(n, 1);

% Define bounds for the variables (optional)
lb = []; % No lower bounds
ub = []; % No upper bounds

% Solve the nonlinear program using fmincon
options = optimoptions('fmincon', 'Display', 'iter');
[x_min, f_min] = fmincon(f, x0, [], [], [], [], lb, ub, [], options);

% Display the result
disp('Optimal Solution:');
disp(x_min);
disp(['Minimum Value of f(x): ', num2str(f_min)]);
```

```
                                            First-order      Norm of
 Iter F-count            f(x)  Feasibility   optimality         step
    0       5    0.000000e+00    0.000e+00    7.000e+00
    1      12   -3.718750e+00    0.000e+00    9.750e+00    1.984e+00
    2      17   -1.779288e+02    0.000e+00    4.288e+01    1.543e+01
    3      22   -3.654515e+06    0.000e+00    4.523e+03    2.674e+03
    4      27   -1.771481e+12    0.000e+00    3.152e+06    1.871e+06
    5      32   -2.142039e+19    0.000e+00    1.096e+10    6.513e+09
    6      37   -6.472296e+27    0.000e+00    1.905e+14    1.132e+14

Problem appears unbounded.

fmincon stopped because the objective function value is less than
the value of the objective function limit and constraints
are satisfied to within the value of the constraint tolerance.
```

```
Optimal Solution:
   1.0e+13 *

   -0.5784
   -9.5917
    1.3267
   -5.8453

Minimum Value of f(x): -6.472295764367742e+27
```

The solver has terminated since the objective function value becomes extremely negative, and the increasing optimality measure indicates that the current solution is getting very close to the optimal solution. This means that the solver is having difficulty finding a bounded solution, and the problem is unbounded.

The **fmincon** function envokes constrained non-linear optimization theory for multivariable functions. The objective function envokes the quadratic form derived in linear algebra. The results from the solver envoke the concepts of bounded and unbounded functions from calculus.

Question 3

[Question 5(b)] Using MATLAB, produce $c \in \mathbb{R}^4$ and symmetric $S \in \mathbb{R}^{nxn}$ whose entries are randomly selected integer values on the closed interval, $[-10, 10]$. Then solve the non-linear program

$$min_{x \in \mathbb{R}^4} f(x)$$

with:

$(b) f(x) = c^T x + \frac{1}{2} x^T (S + 50 I_4) x$ where I_4 is the $4x4$ identity matrix.

Solution:

```
n = 4;  % Dimension of vector c and size of symmetric matrix S
lower_bound = -10; % Range of random values for elements of c and S
upper_bound = 10;

% Generate random vector c in R^4
c = randi([lower_bound, upper_bound], n, 1);

% Generate random symmetric matrix S with integer values in the specified range
S = randi([lower_bound, upper_bound], n, n);
S = triu(S) + triu(S, 1)';  % Ensures symmetry

% Define the 4x4 identity matrix
I_4 = eye(4);

% Define objective function
f = @(x) c' * x + 0.5 * x' * (S + 50 * I_4) * x;

% Initial guess for the optimization
x0 = zeros(n, 1);

% Define bounds for the variables (optional)
lb = []; % No lower bounds
ub = []; % No upper bounds

% Solve the nonlinear program using fmincon
options = optimoptions('fmincon', 'Display', 'iter');
[x_min, f_min] = fmincon(f, x0, [], [], [], [], lb, ub, [], options);

% Display the result
disp('Optimal Solution:');
disp(x_min);
disp(['Minimum Value of f(x): ', num2str(f_min)]);
```

```
                                            First-order      Norm of
 Iter F-count            f(x)  Feasibility   optimality         step
    0       5    0.000000e+00    0.000e+00    7.000e+00
    1      16   -2.727915e-01    0.000e+00    7.664e+00    5.000e-01
    2      24   -4.293013e-01    0.000e+00    6.791e+00    1.250e-01
    3      32   -5.054887e-01    0.000e+00    7.237e+00    1.257e-01
    4      38   -5.657532e-01    0.000e+00    5.232e+00    4.670e-01
    5      49   -1.421070e+00    0.000e+00    1.635e+00    2.335e-01
```

```
    6      63   -1.468190e+00    0.000e+00    4.710e-01    5.108e-02
    7      68   -1.470901e+00    0.000e+00    4.237e-02    1.084e-02
    8      76   -1.470932e+00    0.000e+00    1.302e-02    1.598e-03
    9      81   -1.470934e+00    0.000e+00    1.335e-05    3.395e-04
   10      86   -1.470934e+00    0.000e+00    8.941e-08    4.670e-07

Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

Optimal Solution:
    0.1990
   -0.1567
    0.1614
    0.0372

Minimum Value of f(x): -1.4709
```

The solver finds a local minimum satisfying the constraints at $(0.1990, -0.1567, 0.1614, 0.0372)$ with an objective function value of $f(x) = -1.4709$. It can be seen that the value of the objective function remains about the same after the eighth iteration.

The **fmincon** function envokes constrained non-linear optimization theory for multivariable functions. The objective function envokes the quadratic form derived in linear algebra. The results from the solver envoke the concepts of bounded and unbounded functions from calculus.