Problem Set #1: Nonlinear Programming

Analytic Solutions to NLPs

Exercise 1

Consider the constraints

$$x_1 \geq 0$$
$$x_2 \geq 0$$
$$x_2 - (x_1 - 1)^2 \geq 0$$

in $\mathbb{R}^2$. Sketch the feasible region. Furthermore, show that the point $x_1 = 1, x_2 = 0$ is feasible but not regular.

Solution:

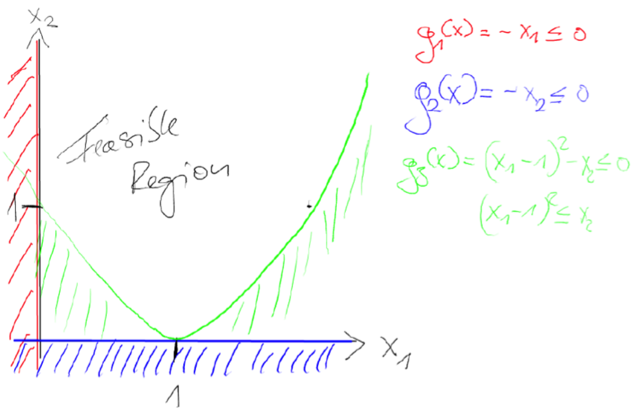

Figure 1: Sketch of the feasible region for Exercise 1.

The feasible region is sketched in Figure 1. The regularity of a point implies that the gradients of the active contraints at this point are linearly independent. At $\mathbf{x} = (1,0)^T$ the constraint $g_1(\mathbf{x}) = -x_1 \leq 0$ is inactive. The constraints $g_2(\mathbf{x}) = -x_2 \leq 0$ and $g_3(\mathbf{x}) = (x_1 - 1)^2 - x_2 \leq 0$ are active. Caculating the gradients of the active constraints yields

$$\nabla \mathbf{g}(\mathbf{x}) = \begin{pmatrix} 0 & -1 \\ 2(x_1 - 1) & -1 \end{pmatrix}.$$

We see that at $\mathbf{x} = (1,0)^T$ we have rank$(\nabla \mathbf{g}(\mathbf{x})) = 1$. Thus $\mathbf{x} = (1,0)^T$ is not a regular point.

Exercise 2

Find a solution to the problem

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^2} \quad & 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2 \\ s.t. \quad & x_1^2 + x_2^2 \leq 5 \\ & 3x_1 + x_2 \leq 6. \end{aligned}$$

Solution:
First, we caculate a stationary point of the unconstrained problem. This yields

$$\nabla f(\mathbf{x})^T = \begin{pmatrix} 4x_1 + 2x_2 - 10 \\ 2x_2 + 2x_1 - 10 \end{pmatrix} = \mathbf{0}.$$

Solving this equation we obtain a candidate point $\mathbf{x} = (0,5)^T$. This point however violates the first constraint $x_1^2 + x_2^2 \leq 5$.

Now, we suppose that at the minimum both constraints are active. If this is the case, there exists a vector $\nu^\star \in \mathbb{R}^2$ such that $\nu_1^\star \geq 0,\ \nu_2^\star \geq 0$

$$\begin{aligned} \begin{pmatrix} 4x_1 + 2x_2 - 10 \\ 2x_2 + 2x_1 - 10 \end{pmatrix}^T + (\nu^\star)^T \begin{pmatrix} 2x_1 & 2x_2 \\ 3 & 1 \end{pmatrix} &= (0,0) \\ x_1^2 + x_2^2 - 5 &= 0 \\ 3x_1 + x_2 - 6 &= 0 \end{aligned}$$

This system of equations has two solutions

$$\begin{pmatrix} x_1^\star \\ x_2^\star \\ \nu_1^\star \\ \nu_2^\star \end{pmatrix} = \begin{pmatrix} \frac{1}{10}(19 - \sqrt{14}) \\ \frac{3}{10}(2 + \sqrt{14} \\ \frac{1}{10}(16 - 7\sqrt{14} \\ \frac{1}{2}(-1 + \sqrt{14}) \end{pmatrix}, \quad \begin{pmatrix} x_1^\star \\ x_2^\star \\ \nu_1^\star \\ \nu_2^\star \end{pmatrix} = \begin{pmatrix} \frac{1}{10}(19 + \sqrt{14}) \\ \frac{3}{10}(2 - \sqrt{14} \\ \frac{1}{10}(16 + 7\sqrt{14} \\ \frac{1}{2}(-1 - \sqrt{14}) \end{pmatrix}. \tag{1}$$

Observe that in both solutions one of the multipliers $\nu_{1,2}^\star$ is negative. This, however, violates the KKT conditions. Thus, at the optimal solution both constraints cannot be active simultaneously.

Now, assume that at the optimum the first constraint is inactive while the second one is active. This yields $\nu_1^\star = 0,\ \nu_2^\star \geq 0$ and the following equations

$$\begin{aligned} \begin{pmatrix} 4x_1 + 2x_2 - 10 \\ 2x_2 + 2x_1 - 10 \end{pmatrix}^T + \nu_2^\star \begin{pmatrix} 3 & 1 \end{pmatrix} &= (0,0) \\ 3x_1 + x_2 - 6 &= 0. \end{aligned}$$

This linear system of equations has the following solution

$$\begin{pmatrix} x_1^\star, & x_2^\star, & \nu_2^\star \end{pmatrix} = \begin{pmatrix} 0.4, & 4.8, & -0.4 \end{pmatrix}.$$

Again the multiplier $\nu_2^\star$ is negative. Thus we reject this solution.

Finally, we assume that at the optimum the first constraint is active while the second one is inactive. This yields $\nu_1^\star \geq 0,\ \nu_2^\star = 0$ and the following equations

$$\begin{aligned} \begin{pmatrix} 4x_1 + 2x_2 - 10 \\ 2x_2 + 2x_1 - 10 \end{pmatrix}^T + \nu_1^\star \begin{pmatrix} 2x_1 & 2x_2 \end{pmatrix} &= (0,0) \\ x_1^2 + x_2^2 - 5 &= 0. \end{aligned}$$

The solution to this set of equations is given by

$$\begin{pmatrix} x_1^\star, & x_2^\star, & \nu_1^\star \end{pmatrix} = \begin{pmatrix} 1, & 2, & 1 \end{pmatrix}.$$

We see that $\nu_1^\star > 0$ and also the second constraint is satisfied. Thus we have found the KKT point. And due to the convexity of the problem this is also the optimal solution.

Exercise 3

Find a solution to the problem

$$\begin{aligned}
\min_{\mathbf{x}\in\mathbb{R}^2} \quad & x_1 \\
s.t. \quad & x_2 = 0 \\
& \mathbf{x} \in \mathcal{X} := \{\mathbf{x} \mid x_1^2 \le x_2\}.
\end{aligned}$$

Solution:
The only feasible point is $\mathbf{x} = (0,0)^T$. Thus it is also the optimal solution. Note, however, that $\mathbf{x} = (0,0)^T$ is not a regular point.

Exercise 4

Consider the following NLP:

$$\begin{aligned}
\min_{\mathbf{x}\in\mathbb{R}^2} \quad & (x_1-3)^2 + (x_2-3)^2 \\
s.t. \quad & 4x_1^2 + 9x_2^2 \le 36 \\
& x_1^2 + 3x_2 = 3 \\
& \mathbf{x} \in \mathcal{X} := \{\mathbf{x} \mid x_1 \ge -1\}.
\end{aligned}$$

a) Sketch the feasible region and the contours of the objective function, then identify the optimum graphically.

b) Using this graphical information, determine the minimum precisely based on first-order necessary conditions. Are the second order conditions of optimality satisfied at this point?

c) Repeat by replacing $\min$ by $\max$.

Solution:

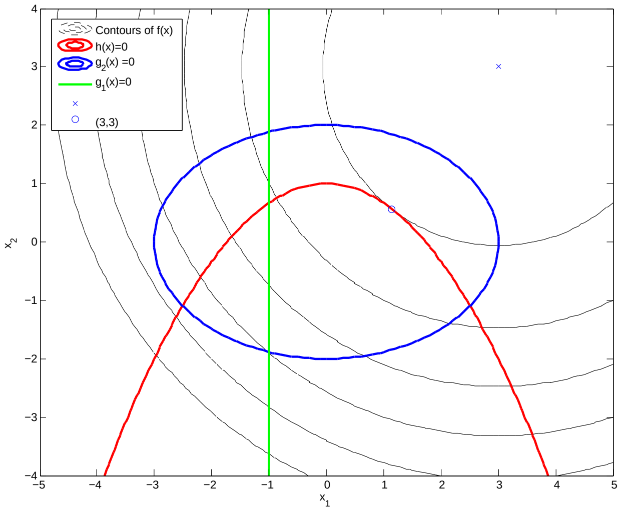

Figure 2: Sketch of the feasible region for Exercise 4.

a) Since the contour lines of the objective function are circles centered at $(3,3)^T$ the optimal solution is cearly the point inside the feasible region which is closest to $(3,3)^T$. The minimum also has to satisfy the equality constraint. Thus the optimal solution is easily found graphically at point marked by the small *o*.

b) At the optimum only the equality constraint is active. Thus we use the method of Lagrange multipliers.

$$L(\mathbf{x},\lambda) = f(\mathbf{x}) + \lambda h(\mathbf{x}).$$

The first-order necessary conditions yield

$$\nabla_x L^T = \begin{pmatrix} 2(x_1-3) \\ 2(x_2-3) \end{pmatrix} + \lambda \begin{pmatrix} 2x_1 \\ 3 \end{pmatrix} = 0$$
$$\nabla_\lambda L^T = x_1^2 + 3x_2 - 3 = 0.$$

Solving this numerically yields $(x^\star, \lambda^\star) = (1.14336,\ 0.5642,\ 1.6238)$. It is easily verfied that the two inequality constraints are inactive that this point.

The Hessian of the Lagrangian is

$$\nabla_{xx} L = \begin{pmatrix} 2+2\lambda & 0 \\ 0 & 2 \end{pmatrix}.$$

This Hessian is positive definite for all $\mathbf{x} \in \mathbb{R}^2$ as long as $\lambda > -1$. In particular, the Hessian of the Lagrangian is positive definite in the tangent subspace, and therefore the second-order sufficient conditions for a strict local minimum are satisfied.

Exercise 5*

Given a cardboard of area A to make a rectangular box. What is the maximum volume that can be attained?

a) Reformulate the problem as an unconstrained one, then find the candidate minima. *Hint: Eliminate x_3 and show that $x_1 = x_2$ must hold at an optimal solution.*

b) Solve the problem directly by using the method of Lagrange multipliers. *Hint: Show that $x_1 = x_2 = x_3$ must hold at an optimal solution.*

Solution:
The problem can be formulated as

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^3} \quad & -x_1x_2x_3 \\ \text{s.t.} \quad & 2(x_1x_2 + x_2x_3 + x_1x_3) - A = 0 \\ & x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0. \end{aligned}$$

Note that the equality constraintimplies that at the same time not more than one x_i can be equal to 0. Thus we solve the equality constraint for x_3 and obtain

$$x_3 = \frac{A - 2x_1x_2}{2(x_1+x_2)}. \tag{2}$$

Now we can eliminate x_3 from the problem and obtain

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^2} \quad & -x_1x_2\frac{A-2x_1x_2}{2(x_1+x_2)} \\ \text{s.t.} \quad & A - 2x_1x2 \geq 0 \\ & x_1 \geq 0, \quad x_2 \geq 0. \end{aligned}$$

Next, we evaluate the gradient of the objective function. We can use the Symbolic Toolbox of Matlab to help us with the computations.

```
x1 = sym('x1','real'); x2 = sym('x2','real'); A = sym('A','real');
x = [x1 x2]';
f = x1^2*x2^2/(x1+x1) - 0.5*A*x1*x2/(x1+x2);
df = simplify(jacobian(f,x))
```

Assuming that no constraints are active, the first-order necessary condition reads

$$\nabla_x f^T = \left[x_2^2 - \frac{x_2^2 \left(x_2^2 + \dfrac{A}{2} \right)}{(x_1 + x_2)^2}, \quad x_1^2 - \frac{x_1^2 \left(x_1^2 + \dfrac{A}{2} \right)}{(x_1 + x_2)^2} \right] = \mathbf{0}$$

From where we obtain

$$(x_1 + x_2)^2 - (x_2^2 + \frac{A}{2}) = 0 \tag{3}$$

$$(x_1 + x_2)^2 - (x_1^2 + \frac{A}{2}) = 0 \tag{4}$$

Computing the difference between both equations we obtain

$$x_2^2 + \frac{A}{2} - x_1^2 - \frac{A}{2} = 0,$$

which proves that $x_1^\star = x_2^\star$. Thus, equation (3) can be rewritten as

$$4x_1^2 - \left(x_1^2 + \frac{A}{2} \right) = 0,$$

from where we have that $x_1^\star = \pm\sqrt{\frac{A}{6}}$. The negative solution has to be ruled out due to infeasibility. Therefore we have that $x_1^\star = x_2^\star = \sqrt{\frac{A}{6}}$. Computing $x_3^\star$ from (2) gives $x_3^\star = \sqrt{\frac{A}{6}}$. Hence, we conclude that the optimal solution is $x_1^\star = x_2^\star = x_3^\star = \sqrt{\frac{A}{6}}$.

Numerical Solutions to NLPs

Exercise 6: Computation of Function Derivatives

Consider the *Speelpenning's* function $f : \mathbb{R}^n \to \mathbb{R}$

$$f(\mathbf{x}) = \prod_{i=1}^{n} x_i.$$

The aim is to compute the gradient of f at a point $\mathbf{x}$. Consider $n = 10$ and $x = \pi + ones(n,1)$.

a) Write a Matlab-function to compute the gradient using forward, backward and central finite differences.

b) Write a Matlab-function to compute the gradient using the *imaginary trick* in Matlab.

c) Repeat the gradient computation via algorithmic differentiation using the CasADi package. The package can be obtained from `https://github.com/casadi/casadi/wiki`. The installations instructions are available on `https://github.com/casadi/casadi/wiki/InstallationInstructions`.

d) Finally, compare the results a) and b) in terms of accuracy with the ones from c).

Solution:
First, we code Speelpenning's function and its analytical gradient:

```
% Speelpenning's function
function f = speel(x)
    f = 1;
    for i=1:length(x)
        f = f * x(i);
    end
end

% analytic gradient of Speelpening's function
function g_s = gradSpeel(x)
    g_s = ones(1,length(x));
    for i = 1:length(x)
        for j = 1:length(x)
            if i ~= j
                g_s(i) = g_s(i)*x(j);
            end
        end
    end
end
```

Second, we implement finite differences in forward, backward and central mode, the imaginary trick and finally the algorithmic differentiation via CasADi:

```
% ex 6.a)
n = 10;
h = 1e-6;
x = pi + ones(n,1);

% compute gradients
g_forward   = grad(@speel, x, h, 'forward');
g_backward  = grad(@speel, x, h, 'backward');
g_central   = grad(@speel, x, h, 'central');

% ex. 6.b)
g_imag      = grad(@speel, x, h, 'imag');

```

```
% ex. 6.c)
g_cas        = grad(@speel, x, h, 'cas');


% computing gradients by different methods
function g = grad(f, x, h, method)
    n = length(x);
    g = zeros(n,1);
    if strcmp(method,'forward')
        for i=1:n
            ei       = zeros(n,1);
            ei(i)    = 1;
            g(i)     = (f(x+h*ei) - f(x))/h;
        end
    end
    if strcmp(method,'backward')
        for i=1:n
            ei       = zeros(n,1);
            ei(i)    = 1;
            g(i)     = (f(x) - f(x-h*ei))/h;
        end
    end
    if strcmp(method,'central')
        for i=1:n
            ei       = zeros(n,1);
            ei(i)    = 1;
            g(i)     = (f(x+0.5*h*ei) - f(x-0.5*h*ei))/h;
        end
    end
    if strcmp(method,'imag')
        for i=1:n
            ei       = zeros(n,1);
            ei(i)    = 1;
            g(i)     = imag((f(x+1i*h*ei) - f(x)))/h;
        end
    end
    if strcmp(method,'cas')
        import casadi.*
        xSym         = SX.sym('x',[n 1]);
        fCas         = f(xSym);
        fCasG        = gradient(fCas,xSym);
        fCasGfun     = Function('speelCasGfun',{xSym},{fCasG});

        g            = full(fCasGfun(x));
    end
end
```

The comparison results can be obtained as follows:

```
% ex. 6.c)
g_anal       = gradSpeel(x)';
errors       = [max(abs(g_forward-g_anal)),max(abs(g_backward-g_anal)),...
                max(abs(g_central-g_anal)),max(abs(g_imag-g_anal)), ...
                max(abs(g_cas-g_anal))];

display('Computational errors (max-norm) for')
display('forward     backward     central     imaginary           CasADi' )
display(num2str(errors))
```

The results are:

```
Computational errors (1-norm) for
forward     backward    central     imaginary           CasADi
0.00024277  0.00024277  0.00068855  1.1642e-10          0
```

Exercise 7: `fmincon` Basics

We want to find (analytically and numerically) the maxima of

$$\max_{\mathbf{x}\in\mathbb{R}^2} \quad f(\mathbf{x})$$
$$s.t. \quad h(\mathbf{x}) = 0$$

where

$$f(\mathbf{x}) = e^{-x_1^2 - x_2^2}, \qquad h(\mathbf{x}) = x_2 + x_1^2 - 1 = 0.$$

a) Check first if the problem is concave. *Hint: Note that we can solve the equality constraint for x_2 and plug it into $f(\mathbf{x})$. Plot the resulting function in 3D using x_1 as the free parameter.*

b) Calculate the Lagrangian, state the first-order necessary conditions of optimality and find candidate solutions.

c) How could you exclude minima from the appearing solutions?

d) Solve the problem using `fmincon` from Matlab's Optimization Toolbox using as initial guesses $[0,0]$, $[-1,-1]$ and $[1,1]$. Try also $[100,1000]$ as initial guess.

Solution:
a) First, we plot the objective function:

```
x1 = [-3:.1:3];
x2 = x1;
F = zeros(length(x1), length(x1));
for i = 1:length(x1)
    for j = 1:length(x2)
    F(i,j) = exp(-x1(i)^2 -x2(j)^2);
    end
end
figure
mesh(x1,x2,F)
xlabel('x_1')
ylabel('x_2')
zlabel('F(x_1, x_2)')
```

Second, we plot the set of admissible solutions:

```
% set of addmissible solutions
x1 = -2:0.1:2;
x2 = 1 - x1.^2;
plot3(x1,x2,F(x1), 'Color', 'r', 'Linewidth', 2);
xlabel('x_1');ylabel('x_2');zlabel('F(x)');
grid on
end
%% define cost function as function of x1
function value=F(x1)
x2 = 1 - x1.^2;
value = exp(-x1.^2-x2.^2);
end
```

It is easy to see that the function is not concave.
b) State the first-order necessary conditions of optimality and find candidate solutions. *Plugging the equality constraint into the objective yields*

$$f(\mathbf{x}) = e^{-x_1^2 - x_2^2} = e^{-x_1^2 - (1-x_1^2)^2}.$$

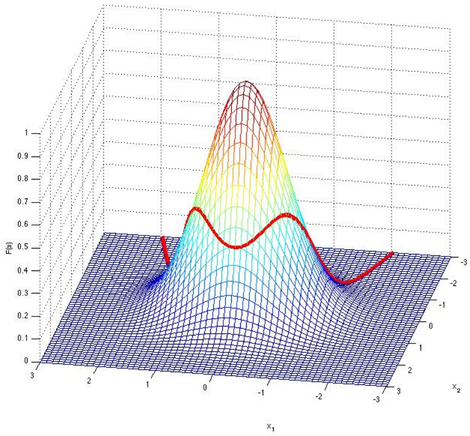

Figure 3: Objective function.

The first-derivative with respect to x_1 reads

$$\frac{\partial f}{\partial x_1} = \left(-4x_1^3 + 2x_1\right) e^{-x_1^2-(1-x_1^2)^2}.$$

The candidate optimal solutions have $x_1 \in \{0, \pm\frac{1}{2}\sqrt{2}\}$.

c) *By evaluating the the second order conditions.*

d)

```
function P1EX7_d
clc;
xtest=[0,0;...
       1,1;...
       -1,-1; ...
       100, 1000];

%loop over initial guesses
for i=1:length(xtest)
  xopt=fmincon(@cost,... %cost function
              xtest(i,:),... %initial guess
         [],...    %A from Au<=B
         [],...    %B
         [],...    %Ae from Ae u=Be
         [],...    %Be
         [],...    %lower bound for u (u_i>=LB)
         [],...    %upper bound for u (u_i<=UB)
         @constr)  %function defining the nonlinear constraints
end
%% define objective
function objective=cost(x)
objective=-exp(-x(1)^2-x(2)^2);
end
%% define constraints
function [ineqconstr,eqconstr]=constr(x)
eqconstr=x(2)+x(1)^2-1;
ineqconstr=[];
end
```

`fmincon` *gives the following solutions:* $(0, 1.0), (-0.707, 0.5), (0.707, 0.5), (14.9, -221.0)$. *From the above considerations and the solver output message we can infer that the first and the last candidates are numerical artefacts. Only the second and the third solution correspond to true local maxima.*

Exercise 8*: Interpretation of Lagrange Multipliers

Consider the following minimization problem

$$\begin{aligned}\min_{\mathbf{x}\in\mathbb{R}^3} \quad & x_1^2 + x_1x_2 + 2x_2^2 - 6x_1 - 2x_2 - 12x_3 \\ s.t. \quad & g_1(\mathbf{x}) = 2x_1^2 + x_2^2 \le 15 \\ & g_2(\mathbf{x}) = x_1 - 2x_2 - x_3 \ge -3 \\ & x_1, x_2, x_3 \ge 0.\end{aligned}$$

a) Find an optimal solution to this problem using `fmincon` from Matlab's Optimization Toolbox. *Hints:*

   – Set the solution tolerance, the function tolerance and the constraint tolerance in `fmincon` to 10^{-7};

   – Use the inital guess $\mathbf{x}_{guess} = (1,\ 1,\ 1)^T$;

   – Make sure that the constraint $g_2 : \mathbb{R}^3 \to \mathbb{R}$ is treated as a linear constraint.

   – First solve the problem by finite-difference gradient approximation for objective and constraints. Second, resolve the problem by providing explicit expressions for the gradients of objective and constraints (ask `fmincon` to check the gradients before running the optimization);

   – Make sure that the solver terminated succesfully in each case.

b) Repeat the optimization with a different initial guess, for instance $\mathbf{x}_{guess} = (14,\ 0,\ 0)^T$. Does `fmincon` converge to the same solution? Could this be expected?

c) Get the values of the Lagrange multipliers at $\mathbf{x}^\star$, as well as the gradients of objective function and constraints. Check whether the optimal solution is a regular point and that it satisfies the KKT conditions.

d) Consider the pertubed problem

$$\begin{aligned}\min_{\mathbf{x}\in\mathbb{R}^3} \quad & x_1^2 + x_1x_2 + 2x_2^2 - 6x_1 - 2x_2 - 12x_3 \\ s.t. \quad & g_1(\mathbf{x}) = 2x_1^2 + x_2^2 \le \theta \\ & g_2(\mathbf{x}) = x_1 - 2x_2 - x_3 \ge -3 \\ & x_1, x_2, x_3 \ge 0\end{aligned}$$

   where θ is the pertubation parameter in $g_1 : \mathbb{R}^3 \to \mathbb{R}$. Solve the pertubed problem for N equally spaced values of $\theta \in [0, 30]$. Use a resolution of 0.5 for θ. Lets denote the optimal solution for some value of θ as $\xi^\star(\theta)$ and the Lagrange multiplier associated with $g_1 : \mathbb{R}^3 \to \mathbb{R}$ as $\omega^\star(\theta)$.

   d1) Plot the objective $f(\xi^\star(\theta))$ versus θ and estimate the slope at $\theta = 15$. What does the corresponding value $\frac{\partial f(\xi^\star(\theta))}{\partial \theta}$ represent?

   d2) Plot $\omega^\star(\theta)$ versus θ. Comment this plot and, in particular, explain the behavior at $\theta = 0$. What is the slope of $f(\xi^\star(\theta))$ at $\theta = 0$?

a) Find an optimal solution to this problem using `fmincon` from Matlab's Optimization Toolbox. *Hints:*

   – Make sure that the SQP algorithm with Quasi-Newton update and line-search is the selected solver in `fmincon`;

   – Set the solution tolerance, the function tolerance and the constraint tolerance in `fmincon` to 10^{-7};

   – Use the inital guess $\mathbf{x}_{guess} = (1,\ 1,\ 1)^T$;

   – Make sure that the constraint $g_2 : \mathbb{R}^3 \to \mathbb{R}$ is treated as a linear constraint.

```
function P1EX8_a
clc
options = optimoptions('fmincon');
options.TolFun =  1E-7;
options.TolCon =  1E-7;
options.TolX    =  1E-7;
options.Algorithm  = 'sqp'
options.GradConstr = 'off'
options.GradObj = 'off';

x_guess = [1 1 1];

%linear constraints
A_ineq = -[1, -2, -1];
b_ineq = 3;
A_eq = [];
b_eq = [];
lb = [0, 0, 0]';
ub = [];

[xopt,fval,exitflag,output,lambda,grad,hessian] =fmincon(@objective,...
                                                    x_guess,...
                                                    A_ineq,...
                                                    b_ineq,...
                                                    A_eq,...
                                                    b_eq,...
                                                    lb,...
                                                    ub,...
                                                    @constr, options);
```

– First solve the problem by finite-difference gradient approximation for objective and constraints. Second, resolve the problem by providing explicit expressions for the gradients of objective and constraints (ask `fmincon` to check the gradients before running the optimization);

```
%% supplying gradient information ot the solver
options.GradConstr = 'on';
options.GradObj = 'on';
options.DerivativeCheck = 'on';

[xopt,fval,exitflag,output,lambda,grad,hessian] =fmincon(@objective,...
                                                    x_guess,...
                                                    A_ineq,...
                                                    b_ineq,...
                                                    A_eq,...
                                                    b_eq,...
                                                    lb,...
                                                    ub,...
                                                    @constr, options);
```

– Make sure that the solver terminated succesfully in each case.

b) Repeat the optimization with a different initial guess, for instance $\mathbf{x}_{guess} = (14,\ 0,\ 0)^T$. Does `fmincon` converge to the same solution? Could this be expected?

The problem is strictly convex. Thus this behavior can be expected.

c) Get the values of the Lagrange multipliers at $\mathbf{x}^\star$, as well as the gradients of objective function and constraints. Check whether the optimal solution is a regular point and that it satisfies the KKT conditions.

The values of the Lagrange multipliers are

$$\lambda_1 = 12, \quad \lambda_2 = 24.7386, \quad \lambda_3 = 1.1432$$

for the linear inequality constraint, the lower bound on x_2 and, respectively, the nonlinear inequality constraint.

```
%% checking regularity
[gopt,hopt, grad_gopt, grad_hopt]=constr(xopt);
[fopt, grad_fopt] = objective(xopt);

Jacobian = [grad_gopt, A_ineq', [0;-1;0]];
rankJac = rank(Jacobian);
display(['The rank of the Jacobian of the active constraints is:  ', ...
 num2str(rankJac)]);
```

d) Consider the pertubed problem

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^3} \quad & x_1^2 + x_1x_2 + 2x_2^2 - 6x_1 - 2x_2 - 12x_3 \\ s.t. \quad & g_1(\mathbf{x}) = 2x_1^2 + x_2^2 \leq {\color{red}\theta} \\ & g_2(\mathbf{x}) = x_1 - 2x_2 - x_3 \geq -3 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

where θ is the pertubation parameter in $g_1 : \mathbb{R}^3 \to \mathbb{R}$. Solve the pertubed problem for N equally spaced values of $\theta \in [0, 30]$. Use a resolution of 0.5 for θ. Lets denote the optimal solution for some value of θ as $\xi^\star(\theta)$ and the Lagrange multiplier associated with $g_1 : \mathbb{R}^3 \to \mathbb{R}$ as $\omega^\star(\theta)$.

d1) Plot the objective $f(\xi^\star(\theta))$ versus θ and estimate the slope at $\theta = 15$. What does the corresponding value $\frac{\partial f(\xi^\star(\theta))}{\partial \theta}$ represent?

d2) Plot $\omega^\star(\theta)$ versus θ. Comment this plot and, in particular, explain the behavior at $\theta = 0$. What is the slope of $f(\xi^\star(\theta))$ at $\theta = 0$?

The plots are shown in Figure 4. The slope of $f(\xi^\star(\theta))$ can be approximated via finite differences

$$\left.\frac{\partial f(\xi^\star(\theta))}{\partial \theta}\right|_{\theta=15} \approx \frac{f(\xi^\star(15+\delta\theta)) - f(\xi^\star(15-\delta\theta))}{2\delta\theta} = -1.1434$$

The slope of $f(\xi^\star(\theta))$ at $\theta = 15$ is the sensitvity of the optimal value with respect to a change in θ. It corresponds to the negative of the corresponding Lagrange muliplier

$$\left.\frac{\partial f(\xi^\star(\theta))}{\partial \theta}\right|_{\theta=15} = -\lambda_3 = -1.1432.$$

At $\theta = 0$ the constraint g_1 requires that $x_1 = x_2 = 0$. Thus, the constraint g_1 and the lower bounds on x_1 and x_2 are active. In addition, the constraint g_2 and the positivity constraint enforce that $x_3 \in [0, 3]$. Since the objective is inversly proportional to x_3 the optimal solution is $(0, 0, 3)$. Hence, four constraints are active in this case and thus the constraints are linearly dependent. In other words, $(0, 0, 3)$ is not a regular point and the KKT-conditions do not hold for $\theta = 0$. The non-existence of a Lagrange multiplier at this point can also be observed in Figure 4 since $\lim_{\theta\to+0} \omega(\theta) = +\infty$. Observe that the slope of $f(\xi^\star(\theta))$ also goes to infinity for $\theta \to +0$.

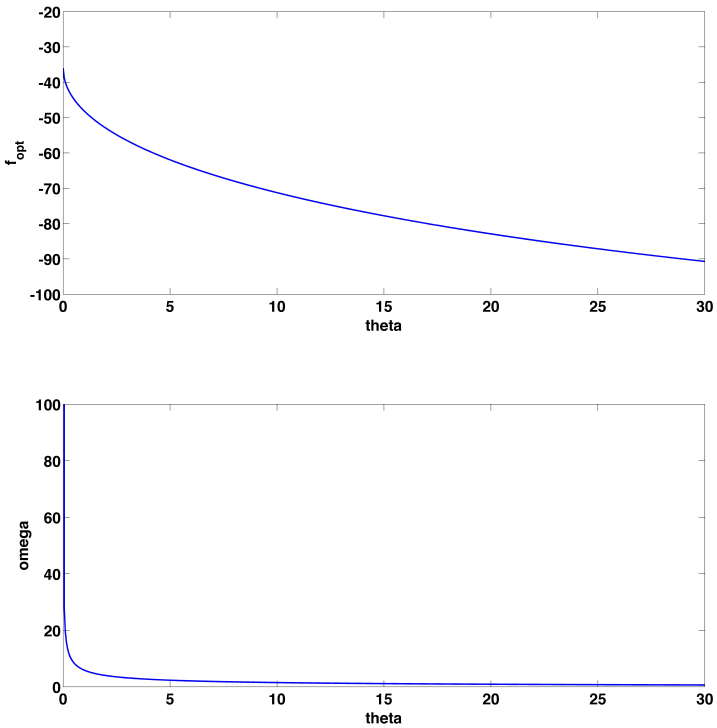

Figure 4: Minimum of the objective function for different values of θ and the corresponding Lagrange multiplier for the active inequality constraint.