Problem 1.

We want to optimize where $\mathbf{x} \in \mathbb{R}^3$,

$f(\mathbf{x}) \triangleq 3x_1 + 5x_2 - 3x_3^2$

subjected to:

$g_1(\mathbf{x}) \triangleq 2x_1^2 - 37x_2 + 9x_3 = 18$

$g_2(\mathbf{x}) \triangleq 5x_1 + x_2 - 5x_3^2 = 24$

First of all we define symbolically the functions:

```
clear; clc;  % clean up
syms x1 x2 x3 lambda1 lambda2 real

x = [x1, x2, x3]; % variable vector xj

lambda = [lambda1, lambda2]; % multipliers vector lambdai

xlambda = [x, lambda];

% create functions symbolically f, g1, g2, y L

syms f(x1, x2, x3)

syms g1(x1, x2, x3)

syms g2(x1, x2, x3)

syms L(x1, x2, x3, lambda1, lambda2)

f(x1,x2,x3) = 3*x1 + 5*x2 - 3*x3^2;

g1(x1,x2,x3) = 2*x1^2 - 37*x2 + 9*x3;
g2(x1,x2,x3) = 5*x1 + x2 + 5*x3^2;

b = [18; 24]; % column vector which contains the constraint right side values

% Lagranian Function

L = f - lambda1*(g1-b(1)) - lambda2*(g2-b(2))
```

L(x1, x2, x3) = $3x_1 + 5x_2 + \lambda_1 \left(-2x_1^2 + 37x_2 - 9x_3 + 18\right) - \lambda_2 \left(5x_3^2 + 5x_1 + x_2 - 24\right) - 3x_3^2$

a) Regularity Analysis

We define the function gradL, which gives the gradient of L respected to x and λ. A feasible point x^* is said to be **regular** (i.e., it satisfies a constraint qualification) if the gradients of the active constraints are linearly independent.

```
gradL = simplify(gradient(L, xlambda))
```

gradL(x1, x2, x3) =

$$\begin{pmatrix} 3 - 4\lambda_1 x_1 - 5\lambda_2 \\ 37\lambda_1 - \lambda_2 + 5 \\ -9\lambda_1 - 6x_3 - 10\lambda_2 x_3 \\ -2x_1^2 + 37x_2 - 9x_3 + 18 \\ -5x_3^2 - 5x_1 - x_2 + 24 \end{pmatrix}$$

Equivalently, x^* is **not regular** if there exists some scalar $c \in \mathbb{R}$ such that:

$$\nabla g_2(x^*) = c\,\nabla g_1(x^*).$$

We define the respective gradients and solve the system.

```
gradg1 = gradient(g1, x)
```

gradg1(x1, x2, x3) =

$$\begin{pmatrix} 4x_1 \\ -37 \\ 9 \end{pmatrix}$$

```
gradg2 = gradient(g2, x)
```

gradg2(x1, x2, x3) =

$$\begin{pmatrix} 5 \\ 1 \\ 10x_3 \end{pmatrix}$$

So,

$$(4x_1, -37, 9) = c * (5, 1, 10x_3)$$

```
syms c1
xc = [x c1]
```

xc = $\begin{pmatrix} x_1 & x_2 & x_3 & c_1 \end{pmatrix}$

```
assume(xc, 'real')
xnr = solve(gradg2 == c1 * gradg1, g1 == b(1), g2 == b(2), xc)
```

```
xnr = struct with fields:
    x1: [0x1 sym]
    x2: [0x1 sym]
    x3: [0x1 sym]
    c1: [0x1 sym]
```

As the system is inconsistent, then, there is no feasible point where the constraint gradients are linearly dependent. These means that all feasible points are **regular.**

b) NC1-Lop

Suppose $\mathbf{x}^* = (x_1^*, x_2^*, x_3^*)$ is a regular local optimum. Then there must exist multipliers λ_1^* and λ_2^* such that the gradient of the Lagrangian $\nabla_x L(\mathbf{x}^*, \lambda^*)$ is zero.

Concretely, if

$$\nabla f(x) = (3, 5, \ -6x_3), \quad \nabla g_1(x) = (4x_1, \ -37, 9), \quad \nabla g_2(x) = (5, 1, 10x_3),$$

then the stationarity condition for $\mathbf{x}^*$ is:

$$\nabla f(x^*) - \lambda_1^* \nabla g_1(x^*) \ - \lambda_2^* \nabla g_2(x^*) \ = (0, 0, 0).$$

In coordinate form, this requirement yields three scalar equations:

$$3 - 4\,x_1^*\,\lambda_1^* - 5\,\lambda_2^* = 0$$

$$5 + 37\,\lambda_1^* - \lambda_2^* = 0$$

$$-6\,x_3^* - 9\,\lambda_1^* - 10\,x_3^*\,\lambda_2^* = 0$$

Hence, for $\mathbf{x}^*$ to be a local optimum, it must satisfy these three stationarity equations together with the original constraints.

We obtain the stationary points for the Lagranian L according to the multipliers.

```
assume(xlambda, 'real')
ptosEstac = solve(gradL == 0, xlambda)
```

```
ptosEstac = struct with fields:
           x1: [4x1 sym]
           x2: [4x1 sym]
           x3: [4x1 sym]
      lambda1: [4x1 sym]
      lambda2: [4x1 sym]
```

```
pe = double([ptosEstac.x1 ptosEstac.x2 ptosEstac.x3 ptosEstac.lambda1
ptosEstac.lambda2])
```

```
pe = 4×5
   -9.6690    5.4567    3.6576   -0.1504   -0.5630
  -10.1460    4.1641   -3.7568   -0.1523   -0.6365
    4.6563    0.7002    0.0607   -0.1080    1.0025
  -97.1627  509.8128   -0.0101    0.1080    8.9970
```

"pe" contains four points with their corresponding multipliers which are feasible and stationary. Thus, they are local optima candidates.

We are going to save all of them as matrices named $\mathbf{x_i}$ and $\boldsymbol{\lambda_i}$ $\quad \forall i \in \{A, B, C, D\}$.

```
xA = double([ptosEstac.x1(1) ptosEstac.x2(1) ptosEstac.x3(1)]);
xB = double([ptosEstac.x1(2) ptosEstac.x2(2) ptosEstac.x3(2)]);
xC = double([ptosEstac.x1(3) ptosEstac.x2(3) ptosEstac.x3(3)]);
xD = double([ptosEstac.x1(4) ptosEstac.x2(4) ptosEstac.x3(4)]);
lambdaA = double([ptosEstac.lambda1(1) ptosEstac.lambda2(1)]);
lambdaB = double([ptosEstac.lambda1(2) ptosEstac.lambda2(2)]);
lambdaC = double([ptosEstac.lambda1(3) ptosEstac.lambda2(3)]);
lambdaD = double([ptosEstac.lambda1(4) ptosEstac.lambda2(4)]);
```

c) 2nd-order Conditions

In order to determine if the points from question b) are optima (local or global), first, we compute the Hessian matrix:

$$\mathbf{Hess}(L(\mathbf{x}^*, \lambda^*))$$

Then, substituting λ, we check the eigenvalues in order to classify if the matrix is definite or not. We obtain the four different Hessians referred to the four different candidate points.

```
HessxL = hessian(L, x)
```

HessxL(x1, x2, x3) =

$$\begin{pmatrix} -4\lambda_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -10\lambda_2 - 6 \end{pmatrix}$$

```
HA = subs(HessxL, lambda, lambdaA)
```

HA(x1, x2, x3) =

$$\begin{pmatrix} \frac{677122728403867}{1125899906842624} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{833084811658503}{2251799813685248} \end{pmatrix}$$

```
double(eig(HA))
```

```
ans = 3×1
   -0.3700
         0
    0.6014
```

As the eigenvalues take both negative and positive values, the matrix is indefinite. We will check later this point.

```
HB = double(subs(HessxL, lambda, lambdaB))
```

```
HB = 3×3
    0.6094         0         0
         0         0         0
         0         0    0.3650
```

In this case, the matrix is PSD, which means that xB is a global minimum. We can assure that is global because as HB is PSD, it means that $L(\mathbf{x}, \lambda^*))$ is convex in $\mathbb{R}^3$.

```
HC = double(subs(HessxL, lambda, lambdaC))
```

```
HC = 3×3
    0.4322         0         0
         0         0         0
         0         0  -16.0246
```

As in the case of point xA, the matrix related to point xC is indefinite. We will analyze it later.

```
HD = double(subs(HessxL, lambda, lambdaD))
```

```
HD = 3×3
   -0.4321         0         0
         0         0         0
         0         0  -95.9704
```

Point xD is just the other way around of xB. The matrix is NSD, so xD is a global maximum and we can assure it's globality because an NSD matrix indicates that $L(\mathbf{x}, \lambda^*))$ is concave in $\mathbb{R}^3$.

We must say that this approach to verify optimal points just taking into account only the Hessian deffinitness comes from considering the second necessary condition:

If $\mathbf{x}^*$ is a regular local minimum satisfying part (b) and is associated with the multiplier λ^*, then for every tangent direction d we must have $d^{\top}\mathrm{Hess}\big(L(x^*, \lambda^*)\big)\, d \;\geq 0$

We define d and we multiply them by the restriction gradients evaluated in our solutions.

```
syms d1 d2 d3 real
d = [d1 d2 d3].';


subs(gradg1.' * d, x, xA)
```

```
ans(x1, x2, x3) =
```

$$9\,d_3 - 37\,d_2 - \frac{2721593429150539\,d_1}{70368744177664}$$

```
subs(gradg2.' * d, x, xA)
```

```
ans(x1, x2, x3) =
```

$$5\,d_1 + d_2 + \frac{41180391572731175\,d_3}{1125899906842624}$$

Now we can print $d^\top H_A d$ and conclude depending on the sign.

```
assume(9*d3 - 37*d2 - (sym("2721593429150539")*d1)/sym("70368744177664") ==
0 & 5*d1 + d2 + (sym("41180391572731175")*d3)/sym("1125899906842624") == 0)

simplify(d.' * HA * d)
```

```
ans(x1, x2, x3) =
```

$$\frac{1581670238076957565936625444852850693611100480865\,d_3^2}{3055832018316570970635999409065760717645214541}$$

As the sign is higher or equal to 0, CN2-minL are fullfilled for xA.

Now, to check CS2-minL, for every d tangent different from 0, the sign of the last expression has to be extrictly positive.

```
solve(9*d3 - 37*d2 - (sym("2721593429150539")*d1)/sym("70368744177664") == 0
& 5*d1 + d2 + (sym("41180391572731175")*d3)/sym("1125899906842624") == 0, d)
```

```
ans = struct with fields:
    d1: 0
    d2: 0
    d3: 0
```

CS2-minL are also fulfilled, so xA is a local minimum.

We repeat the process for xC.

```
subs(gradg1.' * d, x, xC)
```

```
ans(x1, x2, x3) =
```

$$\frac{5242499709344185\,d_1}{281474976710656} - 37\,d_2 + 9\,d_3$$

```
subs(gradg2.' * d, x, xC)
```

```
ans(x1, x2, x3) =
```

$$5\,d_1 + d_2 + \frac{43724728865280405\,d_3}{72057594037927936}$$

```
assume((sym("5242499709344185")*d1)/sym("281474976710656") - 37*d2 + 9*d3 ==
0 & 5*d1 + d2 + (sym("4372472886528040 5")*d3)/sym("72057594037927936") == 0)

simplify(d.' * HC * d)
```

```
ans =
```

$$-\frac{345045516049093365191300552148188581147023401707487 d_3^2}{21546136066498305225299114678351732534529874219499}$$

In this case is the other way around. As the sign is lower or equal to 0, CN2-maxL are fullfilled for xC. Now we check CS2-maxL doing exactly the same, but this time it needs to be extrictly negative.

```
solve((sym("5242499709344185")*d1)/sym("281474976710656") - 37*d2 + 9*d3 ==
0 & 5*d1 + d2 + (sym("4372472886528040 5")*d3)/sym("72057594037927936") == 0,
d)
```

```
ans = struct with fields:
    d1: 0
    d2: 0
    d3: 0
```

CS2-maxL are also fulfilled, so xC is a local maximum.

EVALUATED POINTS

x_1	x_2	x_3	**Optima**
-9.6690	5.4567	3.6576	**Local Minimun**
-10.1460	4.1641	-3.7568	**Global Minimun**
4.6563	0.7002	0.0607	**Local Maximun**
-97.1627	509.8128	-0.0101	**Global Maximun**

Part d) is solved in a different pdf.

Problem 2.

We want to maximize the function $f(\mathbf{x})$ where $\mathbf{x} \in \mathbb{R}^3$

$$f_2(x_1, x_2, x_3) = {x_1}^2 + {x_2}^2 - {x_3}^2$$

subjected to:

$$g_1(x_1, x_2, x_3) \triangleq 8{x_1}^2 + 24x_2 - 15x_3 \le 129$$

$$g_2(x_1, x_2, x_3) \triangleq -{x_1}^2 - 2{x_2}^2 - 4{x_3}^2 \le -15$$

```
clear; clc;
syms x1 x2 x3 mu1 mu2 real
```

```
x = [x1 x2 x3];

mu = [mu1 mu2];

xmu = [x mu];

syms f(x1, x2, x3)

syms h1(x1, x2, x3)

syms h2(x1, x2, x3)

f(x1, x2, x3) = x1^2 + x2^2   - x3^2;

g1(x1, x2, x3)  = 8*x1^2 + 24*x2^2  - 15*x3^2;

g2(x1, x2, x3)  = - x1^2 - 2*x2^2   - 4*x3^2;

c = [129 -15];
```

a) NC1-Lmax

Because the constraints are inequalities, we introduce nonnegative multipliers $\mu_1, \mu_2 \geq 0$. Then define the Lagrangian:

$$L(x, \mu_1, \mu_2) = f(x_1, x_2, x_3) - \mu_1\big(g_1(x_1, x_2, x_3) - c_1\big) - \mu_2\big(g_2(x_1, x_2, x_3) - c_2\big).$$

The 1st-order (KKT) conditions for a regular local maximum are:

- **Stationarity (w.r.t. x):**

$$\nabla_x L\big(x^*, \mu^*\big) = \nabla f(x^*) - \mu_1^* \nabla g_1(x^*) - \mu_2^* \nabla g_2(x^*) = \mathbf{0}.$$

- **Complementary Slackness:**

$$\mu_1^* \big(g_1(x^*) - c_1\big) = 0, \quad \mu_2^* \big(g_2(x^*) - c_2\big) = 0.$$

This means each μ_i^* is zero unless its associated constraint is active (i.e., exactly $\leq$ becomes $=$).

- **Feasibility:**

$$g_1(x^*) \leq c_1, \quad g_2(x^*) \leq c_2, \quad \mu_1^*, \mu_2^* \geq 0.$$

- **Regularity (Linear Independence) Assumption:**

Since we are told every feasible point is regular, we do not need additional details about the rank condition here.

```
L = f - mu1*(g1-c(1)) - mu2*(g2-c(2));

gradxL = simplify(gradient(L, x))
```

gradxL(x1, x2, x3) =

$$\begin{pmatrix} 2\,x_1\,(\mu_2 - 8\,\mu_1 + 1) \\ 2\,x_2\,(2\,\mu_2 - 24\,\mu_1 + 1) \\ 2\,x_3\,(15\,\mu_1 + 4\,\mu_2 - 1) \end{pmatrix}$$

So, computing the gradient and from stationarity condition we have:

$$\begin{cases} 2\,x_1^* \;-\; \mu_1^*(16\,x_1^*) \;-\; \mu_2^*\left(-2\,x_1^*\right) = 0, \\ 2\,x_2^* \;-\; \mu_1^*(48\,x_2^*) \;-\; \mu_2^*\left(-4\,x_2^*\right) = 0, \\ -2\,x_3^* \;-\; \mu_1^*\left(-30\,x_3^*\right) \;-\; \mu_2^*\left(-8\,x_3^*\right) = 0. \end{cases}$$

together with the multipliers satisfying the complementary slackness condition:

$$\mu_1^*\left(h_1(x^*) - 129\right) = 0, \quad \mu_2^*\left(h_2(x^*) + 15\right) = 0, \quad h_1(x^*) \le 129,\; h_2(x^*) \le -15,\; \mu_1^*, \mu_2^* \ge 0.$$

```
subs(gradxL, mu, [0 0])
```

ans(x1, x2, x3) =

$$\begin{pmatrix} 2\,x_1 \\ 2\,x_2 \\ -2\,x_3 \end{pmatrix}$$

```
 % We need the multipliers to be non-negative
assume(mu1 >= 0 & mu2 >= 0 )

kktp = solve(gradxL == 0, mu(1)*(g1 - c(1)) == 0, mu(2)*(g2 - c(2)) == 0, g1
<= c(1), g2 <= c(2), xmu)
```

```
kktp = struct with fields:
     x1: [8x1 sym]
     x2: [8x1 sym]
     x3: [8x1 sym]
    mu1: [8x1 sym]
    mu2: [8x1 sym]
```

```
kktpts = double([kktp.x1 kktp.x2 kktp.x3 kktp.mu1  kktp.mu2 ])
```

kktpts = 8×5

```
        0   -2.4251   -0.8997    0.0476    0.0714
        0    2.4251   -0.8997    0.0476    0.0714
        0   -2.4251    0.8997    0.0476    0.0714
        0    2.4251    0.8997    0.0476    0.0714
        0         0   -1.9365         0    0.2500
        0         0    1.9365         0    0.2500
  -4.0156         0         0    0.1250         0
   4.0156         0         0    0.1250         0
```

We obtained a total of 8 candidate points. Regard that there are only 3 sets of multipliers. This is because the candidate points contain the same numerical values and just vary in their signs.

b) NC2-Lmax

Once we have identified a feasible point $\mathbf{x}^*$ (with KKT multipliers μ^*) that satisfies the 1st-order conditions, the 2nd-order condition for a *regular local maximum* states that:

1. For each constraint $g_i(\mathbf{x}) \leq 0$, only the constraints that are *active* at $\mathbf{x}^*$ (i.e., those with $g_i(\mathbf{x}) \leq 0$) matter for the tangent directions.
2. In every direction $\mathbf{d}$ that remains *tangent to* the set of active constraints at $\mathbf{x}^*$, the quadratic form defined by the Hessian of the Lagrangian must satisfy $d^\top \mathrm{Hess}\big(L(x^*, \mu^*)\big)\, d \ \geq 0$ for a maximization problem. (Equivalently, the Hessian is **negative semidefinite** when restricted to the tangent subspace.

Handling Active Constraints

- If some constraints are strictly inactive at $\mathbf{x}^*$ (i.e., $g_i(\mathbf{x}) < 0$), they do not affect the tangent directions.
- If certain constraints are active, we must impose $\nabla g_i(\mathbf{x}^*)^\top \mathbf{d} = 0$ for each such constraint in order to find the feasible directions $\mathbf{d}$. Then you evaluate $\mathbf{d}^\top \mathrm{Hess}(\mathbf{x}^*, \mu^*)\, \mathbf{d}$. If it is nonpositive for all such directions, $\mathbf{x}^*$ meets the second - order condition for a local maximum.

```
H = hessian(L, x)
```

H(x1, x2, x3) =

$$\begin{pmatrix} 2\mu_2 - 16\mu_1 + 2 & 0 & 0 \\ 0 & 4\mu_2 - 48\mu_1 + 2 & 0 \\ 0 & 0 & 30\mu_1 + 8\mu_2 - 2 \end{pmatrix}$$

As the Hessian only depends on μ_i, we are going to save as a variable their values (Recall that there are only 3 pairs).

```
muA = double([kktp.mu1(1) kktp.mu2(1)])
```

```
muA = 1×2
    0.0476    0.0714
```

```
muB = double([kktp.mu1(5) kktp.mu2(5)])
```

```
muB = 1×2
         0    0.2500
```

```
muC = double([kktp.mu1(7) kktp.mu2(7)])
```

```
muC = 1×2
    0.1250         0
```

```
HA = double(subs(H, mu, muA))
```

```
HA = 3×3
    1.3810         0         0
         0         0         0
         0         0         0
```

The first four points can't be a maximum because the Hessian matrix is PSD. This means that they can be minimums, but no maximums.

```
HB = double(subs(H, mu, muB))
```

```
HB = 3×3
    2.5000         0         0
         0    3.0000         0
         0         0         0
```

It happens exactly the same with the fith and sixth points.

```
HC = double(subs(H, mu, muC))
```

```
HC = 3×3
         0         0         0
         0   -4.0000         0
         0         0    1.7500
```

Now we got an indefinite matrix, which means that we need to dive into the tangent directions to see if they satisfy the NC conditions to be a local maxima. Remember that there are two solutions which are reffered to these multipliers. Let's look at their tangent directions.

```
gradg1 = gradient(g1, x)
```

gradg1(x1, x2, x3) =

$$\begin{pmatrix} 16x_1 \\ 48x_2 \\ -30x_3 \end{pmatrix}$$

```
gradg2 = gradient(g2, x)
```

gradg2(x1, x2, x3) =

$$\begin{pmatrix} -2x_1 \\ -4x_2 \\ -8x_3 \end{pmatrix}$$

To study the tangent directions, recall that first we need to evaluate which restrictions are active for this first point.

```
g1_eval_x7 = double(g1(kktp.x1(7), kktp.x2(7), kktp.x3(7)))
```

```
g1_eval_x7 =
129
```

```
g2_eval_x7 = double(g2(kktp.x1(7), kktp.x2(7), kktp.x3(7)))
```

```
g2_eval_x7 =
-16.1250
```

A constraint is considered active at a candidate solution if its inequality holds as an equality at that point. In our problem, after substituting the candidate values into the constraint functions, we find that the value of $g_1(x)$ is exactly equal to its bound (for example, $g_1(x^*) = 129$), whereas the value of $g_2(x)$ does not equal its bound.

```
gradg1_x7 = gradg1(kktp.x1(7), kktp.x2(7), kktp.x3(7))
```

gradg1_x7 =

$$\begin{pmatrix} -4\sqrt{258} \\ 0 \\ 0 \end{pmatrix}$$

we can fix $d_1 = 0$ and then examine the resulting expression under the 2nd-order condition.

```
syms d1 d2 d3 real
d = [d1, d2, d3].'
```

d =

$$\begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix}$$

```
assume(d1==0)
simplify(d.' * HC * d)
```

ans =

$$\frac{7\,d_3^2}{4} - 4\,d_2^2$$

As the sign of the expression can be both negative or positive, this point is not a maximum.

For the second point under inspection we perform the same steps. Regard that as they share almost the same coordinates values, I've only changed the values of x_1.

```
g1_eval_x8 = double(g1(kktp.x1(8), kktp.x2(7), kktp.x3(7)))
```

```
g1_eval_x8 =
129
```

```
g2_eval_x8 = double(g2(kktp.x1(8), kktp.x2(7), kktp.x3(7)))
```

```
g2_eval_x8 =
-16.1250
```

```
gradg1_x8 = gradg1(kktp.x1(8), kktp.x2(7), kktp.x3(7))
```

gradg1_x8 =

$$\begin{pmatrix} 4\sqrt{258} \\ 0 \\ 0 \end{pmatrix}$$

ans =

$$\frac{7\,d_3^2}{4} - 4\,d_2^2$$

There's no need to continue, as the solution will be exactly the same. The second point under surveillace is not a maximum.

So, in conclusion, there are no candidate points to be a maximum.