WROCLAW UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF ELECTRONICS

Field: Electronics
Speciality: Advanced Applied Electronics

Optimization Methods: Constrained Optimization

Author:
Jaroslaw M. Szumega

Supervisor:
Rafal Zdunek, D.Sc, K-4/W4

Grade:

WROCŁAW 08.06.2017

Contents

Chapter 1

Solution to the given problems

Problem 1: Find numbers x_1 and x_2 that minimize the sum $(x_1 + x_2)$ subject to the constraints

$$x_1^2 + x_2^2 = 2.$$

Draw the feasible set on $\Re^2$ and find all the points that satisfy the KKT conditions. Find the optimal solution that satisfy the second-order necessary conditions. Illustrate the problem geometrically.

According to the task description, we will show the formula of the *c(x)*:

$$c(x) = x_1^2 + x_2^2 - 2 = 0$$

The lagrangian function is (λ is the lagrangian multiplier):

$$L(x, \lambda) = f(x) - \lambda c(x) = x_1 + x_2 - \lambda(x_1^2 + x_2^2 - 2)$$

Now we will formulate the KKT conditions for the given system:

$$\frac{\delta L(x,\lambda)}{\delta x_1} = 1 - \lambda x_1 \triangleq 0,\ then\ x_1 = \frac{1}{2\lambda}$$
$$\frac{\delta L(x,\lambda)}{\delta x_2} = 1 - \lambda x_1 \triangleq 0,\ then\ x_2 = \frac{1}{2\lambda}$$

The c(x) function, subject to calculated x, shall be zero:

$$x_1^2 + x_2^2 - 2 = 0$$
$$(\frac{1}{2\lambda})^2 + (\frac{1}{2\lambda})^2 - 2 = 0$$

$$\lambda^2 = \frac{1}{4}$$
$$\lambda = \frac{1}{2} \vee \lambda = -\frac{1}{2}$$

Now we are calculating both x_1 *and* x_2 according to lambda:

$$x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$x = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

Now to establish the min condition:

$$\min_{x}\{x_1 + x_2\}$$

$$for x = \begin{bmatrix} 1 \\ 1 \end{bmatrix} : x_1 + x_2 = 2$$
$$for x = \begin{bmatrix} -1 \\ -1 \end{bmatrix} : x_1 + x_2 = -2$$

Hence, the solution is: $x = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$

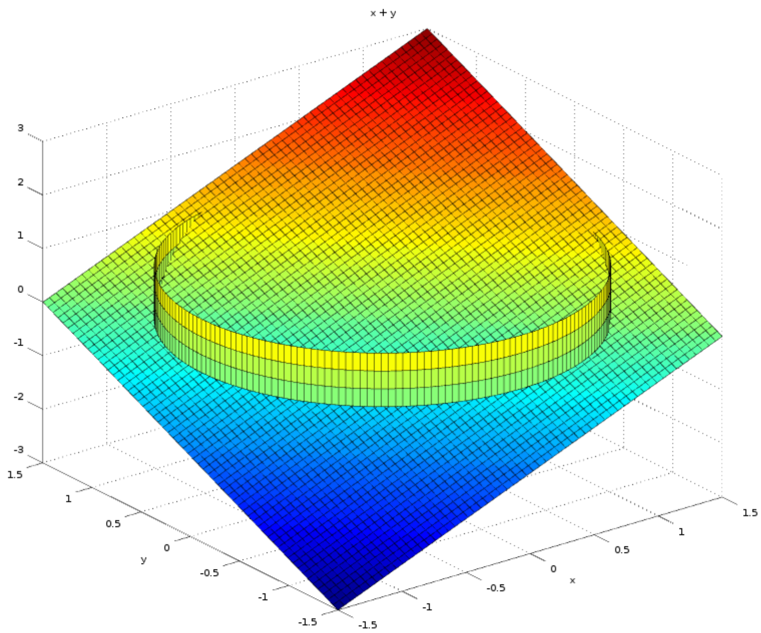

Figure 1.1 The graphical interpretation of given problem (cylinder of radius $\sqrt{2}$) and the plane x+y ($x_1 + x_2$). The solution is in point [-1,-1] - the lowest point in Z-axis that is intersection of the mentioned two surfaces.

The following code was used to illustrate the problem:

```
pkg load symbolic

syms x y;

ezsurf (@(x, y) x + y, [-1.5 1.5 -1.5 1.5])
hold
[X, Y, Z] = cylinder([sqrt(2) sqrt(2) sqrt(2) sqrt(2)],200);
surf(X,Y,Z);
```

Problem 2: Find numbers x_1 and x_2 that minimize the sum $x_1 + x_2$ subject to the constraints

$$x_2 \geq 0,$$
$$2 - x_1^2 - x_2^2 \geq 0.$$

Draw the feasible set on $\Re^2$ and find all the points that satisfy the KKT conditions. Find the optimal solution that satisfies the second-order necessary conditions. Which constraints are active? Illustrate the problem geometrically.

In the following task, we can distinguish two constrains:

$$f(x) = x_1 + x_2$$
$$c_1(x) = x_2 \geq 0$$
$$c_2(x) = 2 - x_1^2 - x_2^2 \geq 0$$

Therefor the Lagrangian form is:

$$L(x, \lambda) = f(x) - \lambda_1 c_1(x) - \lambda_2 c_2(x) = x_1 + x_2 - \lambda_1 x_2 - \lambda_2(2 - x_1^2 - x_2^2)$$

Calculation for the following KKT conditions:

$$\frac{\delta L(x,\lambda)}{\delta x_1} = 1 - 2\lambda_2 x_1 \triangleq 0,\ then\ x_1 = \frac{1}{2\lambda_2}$$

$$\frac{\delta L(x,\lambda)}{\delta x_2} = 1 - \lambda x_1 \triangleq 0,\ then\ x_2 = \frac{\frac{1-\lambda_1}{2\lambda_2}}{den}$$

We can treat inequalities as "equal to zero":

$$2 - (\frac{1}{2\lambda_2})^2 - (\frac{1-\lambda_1}{2\lambda_2})^2 = 0$$
$$\frac{1-\lambda_1}{2\lambda_2} = 0$$

$$\lambda_1 = 1$$
$$\lambda_2^2 = 1/8,\ so\ \lambda_2 = \frac{1}{2\sqrt{2}} \vee -\frac{1}{2\sqrt{2}}$$

After substituting lambda, we calculating the x (we obtain two variants):

$$x_a = [x_1\ x_2] = [\ \sqrt{2}\ \ 0]$$
$$\vee$$
$$x_b = [x_1\ x_2] = [-\sqrt{2}\ \ 0]$$

Point x_b is a solution

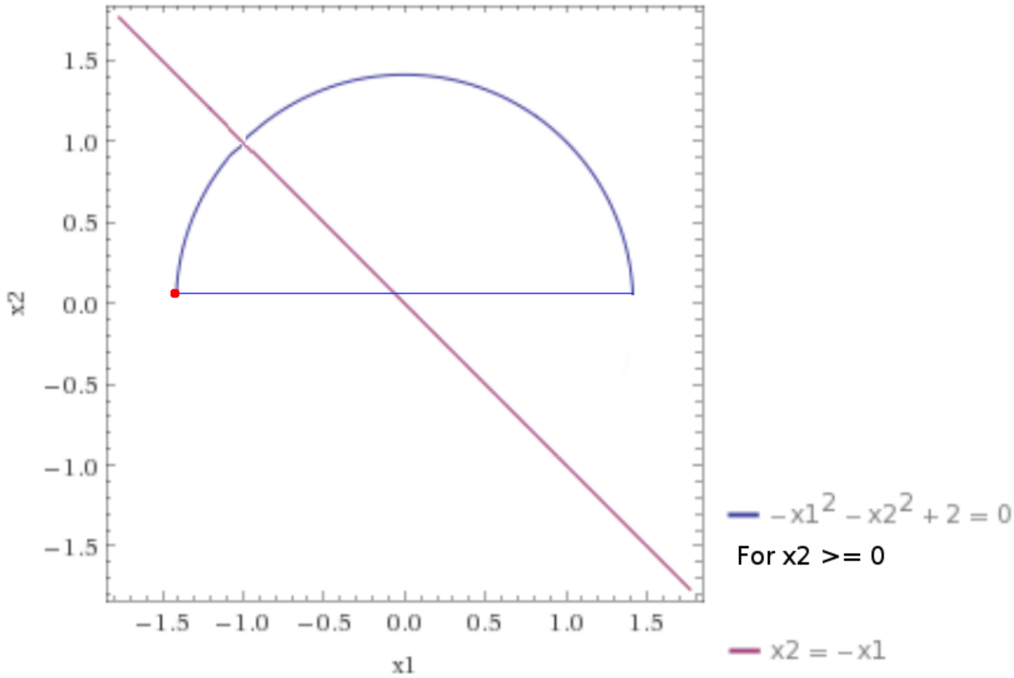

Figure 1.2 Graphical representation. The solution is the point below the line and inside the circle half (including edges). In this case it is the point the most at the bottom left.

Problem 3: Let

$$\min_x x_1^2 + x_2^2, \quad \text{s.t.} \quad x_1 + x_2 \geq 1.$$

Find the solution to the above constrained quadratic problem.

The problem can be easily solved using analytic geometry. The minimization problem represents a circle of unknown radius, while the constraint is a semi–surface, but we can assume it as a line. Looking at first line equation, we can deduct that perpendicular

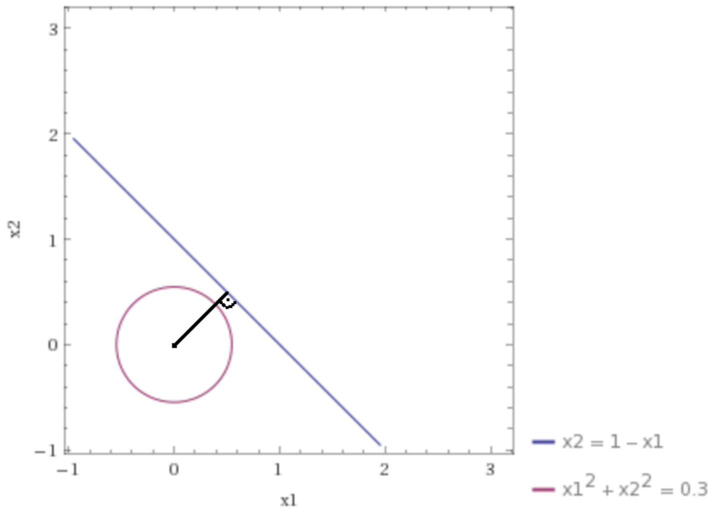

Figure 1.3 The solution will be a point lying on the line, which is tangent to unknown circle of radius R (circle on the figure is just example).

line will have first coefficient equal to $a = 1$. So the thing is just to calculate the free coefficient.
However we know that certainly it has to go straight through the circle center (0,0). Therefore the free coefficient $b = 0$ The two lines line has the form

$$x_2 = 1 - x_1 x_{2p} = x_{1p}$$

The point they cross, is the solution. After solving this problem, we obtain:

$$x = [x_1 \ x_2] = [\frac{1}{2}, \ \frac{1}{2}] = [0.5, \ 0.5]$$

Problem 5: The QP problem is given by:

$$\min_{x} \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x}+\mathbf{c}^T\mathbf{x}\,,$$

subject to:

$$\mathbf{A}\mathbf{x} \geq \mathbf{b}\,,$$
$$\mathbf{x} \geq \mathbf{0}\,.$$

Derive the dual function for this problem, formulate the dual problem, and present the solution in the analytical form. Then using the derived formula solve the following problem:

$$\min_{x} \frac{1}{2}x_1^2 + x_2^2 - x_1x_2 - 2x_1 - 6x_2\,,$$

s.t. $x_1 + x_2 \leq 2,\ -x_1 + 2x_2 \leq 2,\ 2x_1 + x_2 \leq 2,\ x_1 \geq 0,\ x_2 \geq 0.$

The problem will be solved using Quadratic Programming.
To prepare the data for algorithms, we need to calculate the A,b Q and c matrices. Matrix A and c are based on given constraints:

$$A = \begin{bmatrix} 1 & 1 \\ -1 & 2 \\ 2 & 1 \end{bmatrix},\ b = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$$

The next step is to calculate the matrices Q and c. They can be formulate by looking at the general form of the formula:

$$f(x) = \frac{1}{2}q_{11}x_1^2 + \frac{1}{2}q_{22}x_2^2 + q_{12}x_1x_2 + c_1x_1 + c_2x_2$$

$$f(x) = \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2 - x_1x_2 + -2x_1 - 6x_2$$

$$Q = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix},\ c = \begin{bmatrix} -2 \\ -6 \end{bmatrix}$$

```
QP range
x =
0.45714
1.25714

Elapsed time is 0.00281096 seconds.


QP conjugacy
x =
0.45714
1.25714

Elapsed time is 0.00119114 seconds.
```

Problem 6: Solve the problem: $\min_x x_1^2 + x_2^2 + x_3^2$,

s.t. $x_1 + 2x_2 - x_3 = 4$,

$x_1 - x_2 + x_3 = -2$.

Based on the given data, the following data was prepared:

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 1 & -1 & 1 \end{bmatrix} \qquad b = \begin{bmatrix} 4 \\ -2 \end{bmatrix}$$

$$Q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \qquad c = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

and Octave code was coded to solve this task:

```
A = [1 2 -1; 1 -1 1 ]
b = [4; -2]

Q = [2 0 0; 0 2 0; 0 0 2]
c = [0;0;0]

tic
x = QP_range(A,b,Q,c)
toc

tic
x = QP_conjugacy(A,b,Q,c)
toc
```

The computations were made and there are the results. This time the QP range-space method was faster:

```
QP range
x =
0.28571
1.42857
-0.85714

Elapsed time is 0.000784874 seconds.


QP conjugacy
x =
0.28571
1.42857
-0.85714

Elapsed time is 0.00107598 seconds.
>>
```

Problem 7: Solve the problem: $\min_{x} x_1^2 - x_2^2$,

s.t. $x_1 + 2x_2 \geq 2$, $-5x_1 + 4x_2 \leq 10$, $x_1 \leq 0$, $x_2 \geq 0$.

Once again, the matrices was prepared in order to start the algorithm:

$$A = \begin{bmatrix} 1 & 2 \\ 5 & -4 \end{bmatrix}, \ b = \begin{bmatrix} 2 \\ -10 \end{bmatrix}$$

$$Q = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \ c = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The matlab code construction is exactly the same as in previous task, only the data changes.

```
QP range
x =
-0.85714
1.42857

Elapsed time is 0.000548124 seconds.



QP conjugacy
x =
-0.85714
1.42857

Elapsed time is 0.000355005 seconds.
```

Chapter 2

Listings of algorithms

Algorithm 1 - QP range–space

```
function [x_sol] = QP_range(A,b,Q,c)
    disp("QP range")
    x = A\b;
    g = c+Q*x;
    r = A*x -b;
    invQ = inv(Q);
    lambda = (A*invQ*A')\(A*invQ*g-r);
    p= Q\(A'*lambda -g);
    x_sol = x + p;
endfunction
```

Algorithm 2 - QP Conjugacy based

```
function [x_sol] = QP_conjugacy(A,b,Q,c)

    disp("QP conjugacy")

    x = A\b;
    g = c+Q*x;
    r = A*x -b;

    [Qa, Ra] = qr(A');
    L = chol(Qa'*Q*Qa);

    W = Qa*inv(L);
    [m,n] = size(W);

    Z = W(1:end, 1:n-m);
    Y = W(1:end, n-m+1:end);

    py = (A*Y)\(-r);
    pz = -Z'*g;
    p = Y*py + Z*pz;

    x_sol = x + p;
endfunction
```

Bibliography

[1] J. Nocedal, S. J. Wright, Numerical Optimization, Springer, 1999,

[2] Zdunek R., Optimization Methods - lecture slides.

[3] Methods for Constrained Optimization, Coralia Cartis, University of Oxford