2024254019 유진호.

Q1. Simplex Method

max. $20x_1 + 10x_2 + 15x_3$

s.t. $3x_1 + 2x_2 + 5x_3 \le 55$

$2x_1 + x_2 + x_3 \le 26$

$x_1 + x_2 + 3x_3 \le 30$

$5x_1 + 2x_2 + 4x_3 \le 57$

$x_1, x_2, x_3 \ge 0$

max -> min으로 변환 min $-20x_1 - 10x_2 - 15x_3$

제약조건은 그대로.

Canonical Form으로 제약조건을 등식으로 만들기. ($x_1, x_2, x_3, S_1, S_2, S_3, S_4 \ge 0$)

$3x_1 + 2x_2 + 5x_3 + s_1 = 55$

$2x_1 + x_2 + x_3 + S_2 = 26$

$x_1 + x_2 + 3x_3 + S_3 = 30$

$5x_1 + 2x_2 + 4x_3 + S_4 = 57$

변수를 S_n으로 표현함.

Tableau로 변환.

	x_1	x_2	x_3	S_1	S_2	S_3	S_4	
S_1	3	2	5	1	0	0	0	55
S_2	2	1	1	0	1	0	0	26
S_3	1	1	3	0	0	1	0	30
S_4	5	2	4	0	0	0	1	57
$-z$	-20	-10	-15	0	0	0	0	0

Tableau에서 제일 큰 음계수 -20을 pivoting 변수로 선택하여 계산 및 반복. 더이상의 음수계수가 없을 때까지 반복.

최적해 = $x_1 = 1.8$, $x_2 = 20.8$, $x_3 = 1.6$

최대목적함수 = 268.

2024254019 유진호

Q1. Least Squares.

$$y = c_1 e^{-(x-0.25)^2} + c_2 e^{-(x-0.5)^2} + c_3 e^{-(x-0.75)^2}.$$

목적함수 설정.

$$r \text{ (residual)} = Ac - y$$

$$r = \left(c_1 e^{-(x-0.25)^2} + c_2 e^{-(x-0.5)^2} + c_3 e^{-(x-0.75)^2}\right) - y$$

QP 문제 형식화.

$$\min_c \ \frac{1}{2}\|r\|^2 = \frac{1}{2} c^T A^T A c - y^T A c + \frac{1}{2} y^T y.$$

$\because$ $c = [c_1, c_2, c_3]^T$

$Q =$ 계수들의 제곱항을 나타내는 대칭행렬

$q =$ 선형 항을 나타내는 벡터.

$$\therefore Q = \sum_{i=1}^{N} \begin{bmatrix} \left(e^{-(x_i-0.25)^2}\right)^2 & e^{-(x_i-0.25)^2} \cdot e^{-(x_i-0.5)^2} & e^{-(x_i-0.25)^2} \cdot e^{-(x_i-0.75)^2} \\ e^{-(x_i-0.25)^2} \cdot e^{-(x_i-0.5)^2} & \left(e^{-(x_i-0.5)^2}\right)^2 & e^{-(x_i-0.5)^2} \cdot e^{-(x_i-0.75)^2} \\ e^{-(x_i-0.25)^2} \cdot e^{-(x_i-0.75)^2} & e^{-(x_i-0.5)^2} \cdot e^{-(x_i-0.75)^2} & \left(e^{-(x_i-0.75)^2}\right)^2 \end{bmatrix}$$

$$q = -2 \sum_{i=1}^{N} y_i \begin{bmatrix} e^{-(x_i-0.25)^2} \\ e^{-(x_i-0.5)^2} \\ e^{-(x_i-0.75)^2} \end{bmatrix}$$

2024254019 유진호

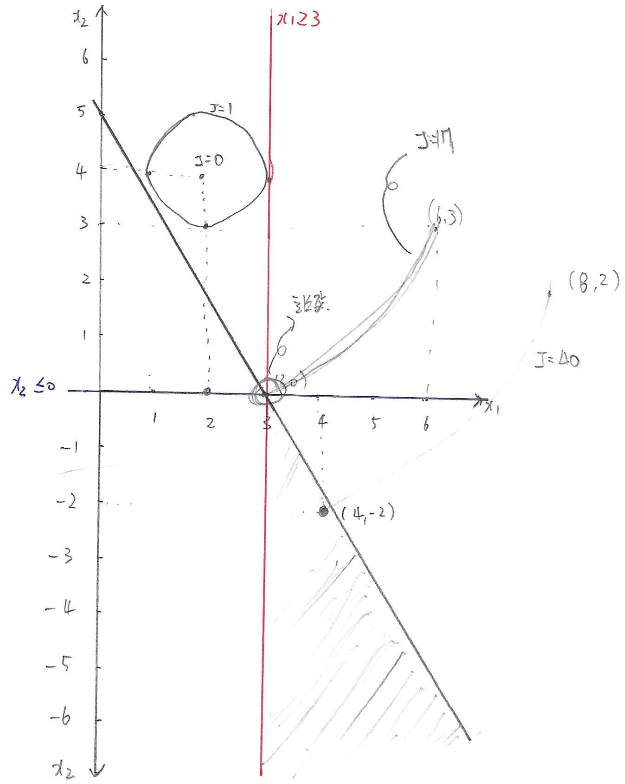
Q2. Graphical Op

min $J = (x_1 - 2)^2 + (x_2 - 4)^2$

s.to $5x_1 + 3x_2 \le 15$

$x_1 \ge 3$

$x_2 \le 0$



$\therefore$ $x_1 = 3$, $x_2 = 0$ 일때

$J = (3-2)^2 + (0-4)^2 = 17.$

2024254019 유진훈

Q3. Unconstrained QP

$f(x_1, x_2) = x_1^2 - 3x_1x_2 + 4x_2^2 + x_1 - x_2$

FONC

$$\frac{\partial f}{\partial x} = \begin{bmatrix} 2x_1 - 3x_2 + 1 \\ -3x_1 + 8x_2 - 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow (x_1^+, x_2^+) = \left(-\frac{5}{7}, -\frac{1}{7}\right)$$

SOSC

$$\frac{\partial^2 f}{\partial x^2} = \begin{bmatrix} 2 & -3 \\ -3 & 8 \end{bmatrix} \rightarrow \text{positive definite} \rightarrow \left(-\frac{5}{7}, -\frac{1}{7}\right) \text{ is a unique minimum point}$$

$\lambda_1 = 2, \lambda_2 = 7$

$f(x_1, x_2) = 2x_1^2 - 4x_1x_2 + 1.5x_2^2 + x_2$

FONC

$$\frac{\partial f}{\partial x} = \begin{bmatrix} 4x_1 - 4x_2 \\ -4x_1 + 3x_2 + 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow (x_1^+, x_2^+) = (1, 1)$$

SOSC

$$\frac{\partial^2 f}{\partial x^2} = \begin{bmatrix} 4 & -4 \\ -4 & 3 \end{bmatrix} \rightarrow \text{indefinite} \rightarrow (1,1) \text{ is a saddle point.}$$

$\lambda_1 = 4, \lambda_2 = -4$

$f(x_1, x_2) = x_1^2 + 2x_2^2 + 3x_3^2 + 3x_1x_2 + 4x_1x_3 - 3x_2x_3$

FONC

$$\frac{\partial f}{\partial x} = \begin{bmatrix} 2x_1 + 3x_2 + 4x_3 \\ 4x_2 + 3x_1 - 3x_3 \\ 6x_3 + 4x_1 - 3x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow (x_1^+, x_2^+, x_3^+) = (0, 0, 0)$$

SOSC

$$\frac{\partial^2 f}{\partial x^2} = \begin{bmatrix} 2 & 3 & 4 \\ 3 & 4 & -3 \\ 4 & -3 & 6 \end{bmatrix} \rightarrow \text{indefinite} \rightarrow (0,0,0) \text{ is a saddle point}$$

$\lambda_1 = 2, \lambda_2 = -1$

2024254019 유진호

Q4. constrained QP

$$\min \frac{1}{2}x^TQx + w^Tx \quad , \quad Q = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}, \; w = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

$0 \le x_1$
$0 \le x_2$
$3x_1 + 2x_2 \le 4$

$$f(x) = \frac{1}{2}\begin{bmatrix} x_1 & x_2 \end{bmatrix}\begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} -1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

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

Lagrangian

$$L(x_1, x_2, \lambda_1, \lambda_2, \lambda_3) = f(x_1, x_2) + \lambda_1(-x_1) + \lambda_2(-x_2) + \lambda_3(3x_1 + 2x_2 - 4)$$

$$= \frac{1}{2}(2x_1^2 - 2x_1x_2 + x_2^2) + \lambda_1(-x_1) + \lambda_2(-x_2) + \lambda_3(3x_1 + 2x_2 - 4)$$

KKT

$$\frac{\partial L}{\partial x_1} = 0, \quad \frac{\partial L}{\partial x_2} = 0$$

$0 \le x_1$, $0 \le x_2$, $3x_1 + 2x_2 \le 4$

$\lambda_1, \lambda_2, \lambda_3 \ge 0$

$\therefore \lambda_1 x_1 = 0$. $\lambda_2 x_2 = 0$, $\lambda_3(3x_1 + 2x_2 - 4) = 0$

$$\frac{\partial L}{\partial x_1} = 2x_1 - 2x_2 - 1 - \lambda_1 + 3\lambda_3 = 0$$

$$\frac{\partial L}{\partial x_2} = -x_1 + x_2 - \lambda_2 + 2\lambda_3 = 0$$

제약조건 $x_1 = 0$, $x_2 = 0$, $3x_1 + 2x_2 = 4$ 인 경우 각각 확인 후 최소값 결정

Q5. Programming

```
import osqp    가져오기 "osqp"을(를) 확인할 수 없습니다.
import numpy as np
from scipy import sparse

# 목적 함수의 계수 행렬 P와 벡터 q 정의
# P는 이차 항 계수 행렬이며, 목적 함수는 x1^2 + 4x2^2로 정의됩니다.
# q는 선형 항 계수 벡터이며, -8x1 - 16x2로 정의됩니다.
P = sparse.csc_matrix([[2, 0], [0, 8]])  # 이차 항 계수: x1^2 + 4x2^2
q = np.array([-8, -16])  # 선형 항 계수: -8x1 - 16x2

# 제약 조건 행렬 A와 상한 경계값 b_upper 정의
# A는 제약 조건을 나타내는 행렬입니다. 여기서는 각 제약 조건을 차례대로 정의합니다.
A = sparse.csc_matrix([
    [1, 1],  # x1 + x2 <= 5: x1과 x2의 합이 5 이하
    [1, 0],  # x1 <= 3: x1의 값이 3 이하
    [-1, 0], # x1 >= 0: x1의 값이 0 이상 (음수가 될 수 없음)
    [0, -1]  # x2 >= 0: x2의 값이 0 이상 (음수가 될 수 없음)
])

# 제약 조건의 상한(b_upper)만 설정
# 여기서는 각 제약 조건에 대한 상한 경계값을 정의합니다.
# 하한은 설정하지 않기 때문에 b_lower는 None으로 둡니다.
b_upper = np.array([5, 3, 0, 0])  # 상한 경계 설정 (각 제약 조건의 최대값)
b_lower = None  # 하한 경계는 따로 설정하지 않음

# OSQP 문제 설정 및 해결    "OSQP": Unknown word.
# OSQP 솔버를 이용해 P, q, A, b_upper를 바탕으로 최적화 문제를 설정하고 풀이합니다.    "OSQP": Unknown word.
prob = osqp.OSQP()    "osqp": Unknown word.
prob.setup(P, q, A, None, b_upper, verbose=True)  # 하한 경계를 None으로 설정하고 문제를 설정
res = prob.solve()

# 결과 출력
# 최적해 (x1, x2)와 목적 함수의 최종 값 출력
print(f"Optimal solution: x1 = {res.x[0]}, x2 = {res.x[1]}")
print(f"Objective value: {res.info.obj_val}")
[EOF]
```

```
(myenv) yoo@jung-G5-5500:~/Desktop$ python3 qp_solver.py
-----------------------------------------------------------------
           OSQP v0.6.3  -  Operator Splitting QP Solver
              (c) Bartolomeo Stellato,  Goran Banjac
        University of Oxford  -  Stanford University 2021
-----------------------------------------------------------------
problem:  variables n = 2, constraints m = 4
          nnz(P) + nnz(A) = 7
settings: linear system solver = qdldl,
          eps_abs = 1.0e-03, eps_rel = 1.0e-03,
          eps_prim_inf = 1.0e-04, eps_dual_inf = 1.0e-04,
          rho = 1.00e-01 (adaptive),
          sigma = 1.00e-06, alpha = 1.60, max_iter = 4000
          check_termination: on (interval 25),
          scaling: on, scaled_termination: off
          warm start: on, polish: off, time_limit: off

iter   objective    pri res    dua res    rho        time
   1  -2.6642e+01   7.47e-16   4.62e+00   1.00e-01   7.22e-05s
  25  -3.1007e+01   3.73e-03   1.54e-03   1.00e-01   1.27e-04s

status:               solved
number of iterations: 25
optimal objective:    -31.0072
run time:             1.77e-04s
optimal rho estimate: 2.79e-01

Optimal solution: x1 = 3.003730060908788, x2 = 1.9929656084839698
Objective value: -31.007248277807193
```