1/ Quadratic Programming

The Problem

$$\min_{x} f(x) = C^T x + \frac{1}{2} x^T D x$$

s.t. $Ax \le B$, $x \ge 0$.

where $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ $C = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}$ $B = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$

$D = \begin{pmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ \vdots & & & \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{pmatrix}$. $A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$

Symmetric p.d. matrix.

Introducing slack variables, we have.

$$\min f(x)$$

$$\text{s.t. } A_i^T x + s_i^2 = b_i \quad \text{for } i = 1(1)m$$

$$-x_j + t_j^2 = 0 \quad \text{for } j = 1(1)n$$

where $A_i = \begin{pmatrix} a_{i1} \\ \vdots \\ a_{in} \end{pmatrix}$

The Lagrange function can be written as

$$L(x, S, T, \lambda, \theta) = C^T x + \frac{1}{2} x^T D x + \sum_{i=1}^{m} \lambda_i \left(A_i^T x + s_i^2 - b_i \right) + \sum_{j=1}^{n} \theta_j \left(-x_j + t_j^2 \right)$$

The necessary conditions for the stationaries of L give

$$\frac{\partial L}{\partial x_j} = c_j + \sum_{i=1}^{n} d_{ij} x_i + \sum_{i=1}^{m} \lambda_i a_{ij} - \theta_j = 0 \quad \text{for } j = 1(1)n \qquad -(1)$$

$$\frac{\partial L}{\partial s_i} = 2 \lambda_i s_i = 0 \quad \text{for } i = 1(1)m \qquad -(2)$$

$$\frac{\partial L}{\partial t_j} = 2 \theta_j t_j = 0 \quad \text{for } j = 1(1)n \qquad -(3)$$

$$\frac{\partial L}{\partial \lambda_i} = A_i^T x + s_i^2 - b_i = 0 \quad \text{for } i = 1(1)m \qquad -(4)$$

$$\frac{\partial L}{\partial \theta_j} = -x_j + t_j^2 = 0 \quad \text{for } j = 1(1)n \qquad -(5)$$

By defining a set of new variables y_i as

$$y_i = s_i^2 \geq 0 \quad i = 1(1)m$$

we have from (4) $A_i^T x - b_i = -y_i$ for $i = 1(1)m$ $\qquad -(6)$

Multiplying (2) by s_i and (3) by t_j we have

$$\lambda_i s_i^2 = \lambda_i y_i = 0 \quad \text{for } i = 1(1)m \qquad -(7)$$

$$\theta_j t_j^2 = 0 \quad \text{for } j = 1(1)n. \qquad -(8)$$

Combining (6) & (7) we have

$$\lambda_i (A_i^T x - b_i) = 0 \quad \text{for } i = 1(1)m \qquad -(9)$$

$$\theta_j x_j = 0 \quad \text{for } j = 1(1)n \qquad -(10)$$

Thus the necessary conditions can be summarized as follows:

$$c_j - \theta_j - \sum_{i=1}^{n} x_i d_{ij} + \sum_{i=1}^{m} \lambda_i a_{ij} = 0 \quad \text{for } j = 1(1)n \quad — \text{(A)}$$

$$A_i^T x - b_i = -y_i \quad i = 1(1)m \quad — \text{(B)}$$

$$x_j \geq 0 \quad j = 1(1)n \quad — \text{(C)}$$

$$y_i \geq 0 \quad i = 1(1)m \quad — \text{(D)}$$

$$\lambda_i \geq 0 \quad i = 1(1)m \quad — \text{(D)}$$

$$\theta_j \geq 0 \quad j = 1(1)n \quad — \text{(E)}$$

$$\lambda_i y_i = 0 \quad i = 1(1)m \quad — \text{(F)}$$

$$\theta_j x_j = 0 \quad j = 1(1)n \quad — \text{(G)}$$

Thus we find that equations A — E are all linear and equation (F) and (G) are quadratic in $x_j, y_i, \lambda_i, \theta_j$

Thus the soln. of the original quadratic programming problem can be obtained by finding a nonnegative solution of the set of $O(m+n)$ linear equations that also satisfies the $(m+n)$ equations stated in (F) and (G).

Since D is a positive definite matrix, $f(x)$ will be a convex function and the feasible space is convex (because of linear constraints). Thus the local minimum of the problem is global minimum.

Further it can be seen that there are $2(m+n)$ variables and $2(m+n)$ equations (A), (B), (F) & (G)

Thus the soln. is unique.

Thus, if the soln. exists, it must give the optimum solution of the quadratic programming problem directly.

Solving the constraint satisfaction problem by simplex method.

Introduce n artificial variables.

$$c_j - \theta_j + \sum_{i=1}^{n} x_i d_{ij} + \sum_{i=1}^{m} \lambda_i a_{ij} + z_j = 0 \quad j = 1(1)n$$

~~Objective Minimize $F = \sum_{j=1}^{n} \theta_j$~~

The other linear constraints are

$$A_i^T x + y_i = b_i \quad i = 1(1)m$$

$$x \geq 0, \; y \geq 0, \; \lambda \geq 0 \quad \theta \geq 0.$$

Objective fn.
Minimize $\quad F = \sum_{j=1}^{n} z_j$

We also need to take care of the additional condition

$$\left.\begin{array}{ll} \lambda_i y_i = 0 & i = 1(1)m \\ \theta_j x_j = 0 & j = 1(1)n \end{array}\right\}$$

implying λ_i & y_i can not stay in the basis simultaneously, and θ_j and x_j can not stay in the basis simultaneously.

Thus while entering y_i in the basis, we need to ensure that λ_i is not present in the basis.

Otherwise, we need to first remove λ_i then y_i can be entered into the basis.

Example.

Minimize $f = -4x_1 + x_1^2 - 2x_1x_2 + 2x_2^2$

s.t. $2x_1 + x_2 \le 6$

$x_1 - 4x_2 \le 0$

$x_1, x_2 \ge 0$.

Solution

Introduce slack variables $y_1 = s_1^2$, $y_2 = s_2^2$ and surplus variables $\theta_1 = t_1^2$ and $\theta_2 = t_2^2$

The problem is

$$\text{Minimize } f = \underbrace{(-4, 0)}_{C}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \frac{1}{2}(x_1\ x_2)\underbrace{\begin{pmatrix} 2 & -2 \\ -2 & 4 \end{pmatrix}}_{D}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$\text{s.t. } \underbrace{\begin{bmatrix} 2 & 1 \\ 1 & -4 \end{bmatrix}}_{A}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \underbrace{\begin{pmatrix} 6 \\ 0 \end{pmatrix}}_{b}$$

$$-x_1 + \theta_1 = 0$$

$$-x_2 + \theta_2 = 0$$

The feasibility constraints are

$$-4 - \theta_1 + 2x_1 - 2x_2 + 2\lambda_1 + \lambda_2 = 0$$

$$0 - \theta_2 - 2x_1 + 4x_2 + \lambda_1 - 4\lambda_2 = 0$$

$$2x_1 + x_2 - 6 = -y_1$$

$$x_1 - 4x_2 - 0 = -y_2$$

$x_1 \ge 0$, $x_2 \ge 0$, $y_1 \ge 0$, $y_2 \ge 0$, $\lambda_1 \ge 0$, $\lambda_2 \ge 0$, $\theta_1 \ge 0$, $\theta_2 \ge 0$

$\lambda_1 y_1 = 0 \quad \lambda_2 y_2 = 0$

$\theta_1 x_1 = 0 \quad \theta_2 x_2 = 0$

$\equiv$

$$2x_1 - 2x_2 + 2\lambda_1 + \lambda_2 - \theta_1 + z_1 = 4$$

$$-2x_1 + 4x_2 + \lambda_1 - 4\lambda_2 - \theta_2 + z_2 = 0$$

$$2x_1 + x_2 + y_1 = 6$$

$$x_1 - 4x_2 + y_2 = 0$$

~~z_1 z_2~~ z_1, z_2 are artificial variables. The objective to minimize $F = z_1 + z_2$.

			0	0	0	0	0	0	0	0	-1	-1	
C	Basic variable	x_{bi}	x_4	x_2	λ_1	λ_2	θ_1	θ_2	y_1	y_2	z_1	z_2	x_{bi}/y_{ij} for y_{ij} >
0	y_1	6	2	1	0	0	0	0	1	0	0	0	6
0	y_2	0	1	-4	0	0	0	0	0	1	0	0	~~0~~
-1	z_1	4	2	-2	2	1	-1	0	0	0	1	0	~~2~~
-1	z_2	0	-2	$\boxed{4}$	1	-4	0	-1	0	0	0	1	0 ← smallest
$z_j - c_j$		-4	0	-2	-3	3	1	1	0	0	0	0	

↑ (x_2) Selected for entering into the basis

↑ (λ_1) most negative But y_1 is in basis so λ_1 does not enter in the basis

			0	0	0			0	0	0	-1	-1	
C	Basic variable	x_{bi}	x_4	x_2	λ_1	λ_2	θ_1	θ_2	y_1	y_2	z_1	z_2	
0	y_1	6	$\boxed{5/2}$	0	-1/4	1	0	1/4	1	0	0	-1/4	12/5 ← smaller
0	y_2	0	-1	0	1	-4	0	-1	0	1	0	1	
-1	z_1	4	1	0	5/2	-1	-1	-1/2	0	0	1	1/2	4
0	x_2	0	-1/2	1	1/4	-1	0	-1/4	0	0	0	1/4	
$z_j - c_j$		-4	-1	0	-5/2	1	1	1/2	0	0	0	1/2	

↑ (x_4) x_4 is selected for entering

↑ (λ_1) most negative.

C	Basic variable	x_{bi}	x_4	x_2	λ_1	λ_2	θ_1	θ_2	y_1	y_2	z_1	z_2	
0	x_4	12/5	1	0	-1/10	2/5	0	1/10	2/5	0	0	-1/10	
0	y_2	12/5	0	0	9/10	-18/5	0	-9/10	2/5	1	0	9/10	8/3
-1	z_1	8/5	0	0	$\boxed{13/5}$	-7/5	-1	-3/5	-2/5	0	1	3/5	8/13 ← smaller
0	x_2	6/5	0	1	1/5	-4/5	0	-1/5	1/5	0	0	1/5	6
$z_j - c_j$		-8/5	0	0	-13/5	7/5	1	3/5	2/5	0	0	2/5	

↑ (λ_1) most negative

C	Basic variable	x_{bi}	x_4	x_2	λ_1	λ_2	θ_1	θ_2	y_1	y_2	z_1	z_2
0	x_4	32/13	1	0	0							
0	y_2	24/13	0	0	0					0		
0	λ_1	8/13	0	0	1							
0	x_2	14/13	0	1	0							

c	Basic variable	x_{B_i}	x_1	x_2	λ_1	λ_2	θ_1	θ_2	y_1	y_2	z_1	z_2
0	x_1	$32/13$	1	0	0	$9/26$	$-1/26$	$1/13$	$5/13$	0	$1/26$	$-1/13$
0	y_2	$24/13$	0	0	0	$-81/26$	$9/26$	$-9/13$	$7/13$	1	$-9/26$	$9/13$
0	λ_1	$8/13$	0	0	1	$-7/13$	$-5/13$	$-3/13$	$-2/13$	0	$5/13$	$3/13$
0	x_2	$14/13$	0	1	0	$-9/13$	$1/13$	$-2/13$	$3/13$	0	$-1/13$	$2/13$
			0	0	0	0	0	0	0	0	1	1

Thus opt sol. $x_1^* = 32/13 \qquad x_2^* = 14/13$

$$f_{min} = f(x_1^*, x_2^*) = -88/13.$$

Quadratic Programming Problem is

$$\text{Max } Z = c\underset{\sim}{x} + x'Qx$$

$$\text{Subject to } Ax \le b,\ x \geqslant 0$$

is polynomially solvable if the matrix Q is nnd.
However, if Q is n.d. the problem is NP-hard.
A $\left(1 - \frac{1-\epsilon}{(m(1+\epsilon))^2}\right)$ factor approximation algorithm is available for $\epsilon \in \left(0, 1-\frac{1}{\sqrt{2}}\right)$. The time complexity is $O\left(n^3\left(m \log\frac{1}{\delta} + \log\log\frac{1}{\epsilon}\right)\right)$ where δ is the radius of the largest ball inside the feasible region.

S.P. Tarasov L.G. Khachiyan M.K. Kozlov
Poly time Solvability of Convex quadratic Programming
USSR Computational Math & Math'l Physics, 1980

P.M. Pardalos & S.A. Vavasis Quadratic Programming with one negative eigen value is NP hard J. Global Optimisation 1991

M. Fu Z Luo Y. Ye Approximation Algo for quadratic Programming J. Comb. Opt. 1998

The Lagrange function can be written as

$$L(x, s, T, \lambda, \theta) = C^T x + \frac{1}{2}x^T D x + \sum_{i=1}^{m} \lambda_i \left(A_i^T x + s_i^2 - b_i\right) + \sum_{j=1}^{n} \theta_j \left(-x_j + t_j^2\right)$$