II Constrained optimization

1. Equality constrained problems and the Lagrangian function

⊕ $\min f(x)$

$$\text{s.t. } h_j(x) = 0 \quad , j = 1, 2, \ldots, r < n$$

$$L(x, \lambda) = f(x) + \sum_{j=1}^{r} \lambda_j h_j(x)$$

⊕ Assume $J_h(x)$ is of rank r → necessary condition for (x^*, λ^*) is

$$\frac{\partial L}{\partial x_i}(x^*, \lambda^*) = 0 \quad , i = 1, 2, \ldots, n$$

$$\frac{\partial L}{\partial \lambda_j}(x^*, \lambda^*) = 0 \quad , j = 1, 2, \ldots, r$$

⊕ Sufficient: • f_i and h_j are convex function ; concave ($\lambda_j < 0$)

• $H_L(x)$ is positive definite

⊕ Prove indeed a minimum: for any step Δx compatible with $h(x) = 0$

• $f(x^* + \Delta x) = f(x^*) + \nabla^T f(x^*) \Delta x + \frac{1}{2} \Delta x^T H_f(x) \Delta x$

$$\underbrace{h(x^* + \Delta x)}_{= 0} = \underbrace{h(x^*)}_{= 0} + \nabla^T h(x) \Delta x + \frac{1}{2} \Delta x^T H_h(x) \Delta x$$

→ solve for Δx

→ solve for Δf < <0 → maximum ; >0 → minimum

pwc

⊕ special cases: quadratic function with linear equality constraints

$$f(x) = \frac{1}{2}x^T A x + b^T x + c$$

s.t

$$Cx = d$$

$$\Rightarrow L(x, \lambda) = \frac{1}{2}x^T A x + b^T x + c + \lambda(Cx - d)$$

$$\frac{\partial L}{\partial x} = \frac{1}{2}x^T(A + A^T) + b^T + \lambda^T C = 0$$

$A = A^T$

$$\frac{\partial L}{\partial \lambda} = Cx - d = 0$$

$$\Rightarrow \begin{cases} Ax + b + C^T\lambda = 0 \\ Cx = d \end{cases}$$

$$\Rightarrow \begin{cases} x + C^T\lambda = -b \\ Cx = d \end{cases}$$

$$\Rightarrow \begin{bmatrix} A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} -b \\ d \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x^* \\ \lambda^* \end{bmatrix} = \begin{bmatrix} A & C^T \\ C & 0 \end{bmatrix}^{-1} \begin{bmatrix} -b \\ d \end{bmatrix}$$

2. Optimization with inequality constraints. KKT conditions

⊕ Primal problem:

$$\text{minimize} \quad f(x)$$

$$\text{s.t} \quad g_j(x) \le 0 \quad , j = 1, 2, \dots, m$$

$$L(x, \lambda) = f(x) + \sum_{j=1}^{m} \lambda_j g_j(x)$$

KKT conditions:

$$\frac{\partial L}{\partial x_i} = \frac{\partial f}{\partial x_i}(x^*) + \sum_{j=1}^{m} \lambda_j \frac{\partial g_j}{\partial x_i}(x^*) = 0$$

$$g_j(x^*) \le 0$$

$$\lambda_j^* g_j(x^*) = 0 \quad \begin{cases} \lambda_j^* = 0 \to g_j \text{ inactive} \\ g_j^* = 0 \to \text{active} \end{cases}$$

$$\lambda_j^* \ge 0$$

⊕ Sufficient: f_i, g_j are all convex

⊕ Existence of λ^* is guaranteed $\Leftrightarrow$ $\exists h \in \mathbb{R}^n$ s.t for each active constrain j: $\nabla^T g_j(x^*) h < 0$

$\Leftarrow$ convex
J of active constraints at x^* is full-rank
all constraints are linear.

3. Duality theorem

⊕ $h(\lambda) = \min_{x} L(x,\lambda)$ - dual function

$D = \{\lambda \mid h(\lambda) \ \exists \text{ and } \lambda \geq 0\}$

defines the dual problem

$$\max_{\lambda \in D} h(\lambda)$$

$$= \max_{\lambda \in D} \left(\min_{x} L(x,\lambda) \right)$$

⊕ Need: . x^* is the solution of the primal problem
. λ^* is the solution of the dual problem
. $f(x^*) = h(\lambda^*)$

⊕ Practical approach: Solve DP → to get $x^*(\lambda^*)$
→ Find $\max_{\lambda} h(\lambda)$
→ sub back in $L(x^*, \lambda^*)$

pwc

4. Quadratic programming

⊕ $\min f(x) = \frac{1}{2} x^T A x + b^T x + c$

st $Cx \leq d$

solution is either { interior point: unconstrained / boundary: Active }

→ Key is to find the active set constraints

→ Try different combinations of constraints starting from the unconstrained solutions

5. Sequential Quadratic Programming (SQP)

⊕ $\min f(x)$ → Approximate through a QP

st $g_j(x) \leq 0$, $j = 1, 2, \cdots m$

$h_j(x) = 0$, $j = 1, 2, \ldots, r$

Given estimates (x^k, λ^k, μ^k), $k = 0, 1, \ldots$ of the solutions and the respective Lagrange multiplier values

$\lambda^k \geq 0$ → step s of iteration $k+1$ st

$x^{k+1} = x^k + s$ is given by

the solution of kth Q-P problem

pwc

⊕ QP-k (x^k, λ^k, μ^k) :

$$\underset{s}{\operatorname{argmin}} \; F(s) = f(x^k) + \nabla^T f(x^k) s + \frac{1}{2} s^T H_L(x^k) s$$

$$\text{st} \quad g(x^k) + \left[\frac{\partial g(x^k)}{\partial x}\right]^T s \le 0$$

$$h(x^k) + \left[\frac{\partial h(x^k)}{\partial x}\right]^T s = 0$$

$$H_L(x^k) = \nabla^2 f(x^k) + \sum_{j=1}^{m} \lambda_j^k \nabla^2 g_j(x^k)$$

$$+ \sum_{j=1}^{l} \mu_j^k \nabla^2 h_j(x^k)$$

→ Solve for s, λ^{k+1}, μ^{k+1}

⊕ on each iteration, active set method is often used to determine the active set at that iteration

⊕ once active set is determined

→ equality constrained problems

→ use Newton's SQP.

④ After verify convergence of the sub-problem
→ we look to update working set of the next iteration from the initial working set.

- Look for violation of constraints : If violated
→ add to working set

- No violation → check if the solution can be improved by :

  - Add active constraints :
    $\text{argmax } g_j ~~(x_\ell + \alpha(x_k - x_\ell)), j \notin W_k~~$
    $\alpha \quad g_j(x_\ell + \alpha(x_k - x_\ell)), j \notin W_k$
    x_ℓ : last when working set was updated
    If $\alpha < 1 \Leftrightarrow x_\ell + \alpha(x_k - x_\ell) < x_k$
    → update : $x_{k+1} = x_\ell + \alpha_k(x_k - x_\ell)$
    → add g_j to working set

  - Remove active constraints :
    - Remove negative multiplier (KKT)
    - All positive → completed and check KKT condition is satisfied

- Initial working set, guess is important.

$h: \mathbb{R}^n \to \mathbb{R}^r$, $\lambda: \mathbb{R}^r$

⊕ Newton SQP:

- $\min f(x)$
  s.t $h_i(x) = 0 \quad i = 1, \dots, r$

$\to L(x, \lambda) = f(x) + \sum_{i=1}^{r} h_i(x)\lambda_i = f(x) + \nabla^T h(x)\lambda$

$\to$ conditions: $\frac{\partial L}{\partial x} = 0$, $\frac{\partial L}{\partial \lambda} = 0$ $\to$ Jacobian : $r \times n$

$$\nabla L(x, \lambda) = \begin{pmatrix} \nabla f + \nabla^T h \lambda \\ h \end{pmatrix} = 0$$

- approx gradient:

$$\nabla L(x, \lambda) = \nabla L(x_k, \lambda) + \nabla^2 L(x_k, \lambda) \begin{pmatrix} \Delta x \\ \Delta H \end{pmatrix}$$

$$= 0$$

$$\to \begin{pmatrix} \Delta x \\ \Delta H \end{pmatrix} = -(\nabla^2 L(x_k, \lambda))^{-1} \nabla L(x_k, \lambda)$$

$\to$ iterate: $\begin{pmatrix} x^{k+1} \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ \lambda^k \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta h \end{pmatrix}$

- $\nabla^2 L(x, \lambda) = H = \begin{pmatrix} \partial_{xx} L & \nabla h \\ \nabla^T h & 0 \end{pmatrix}$

$$\frac{\partial L}{\partial \lambda} = \begin{pmatrix} h_1 \\ \vdots \\ h_r \end{pmatrix} = h(x)$$

⊕ SQP application with SWT:

$$\min \; SP(x) = \sum_{i=1}^{n-2} \left[\frac{f(t_{i+1}, t_{i+2}) - f(t_i, t_{i+1})}{t_{i+2} - t_i} - \frac{f(t_i, t_{i+1}) - f(t_{i-1}, t_i)}{t_{i+1} - t_{i-1}} \right]^2$$

$$\text{s.t } (\hat{R}_j(x) - R_j) = 0$$

$$R_k - \varepsilon_k \leq \hat{R}_k(x) \leq R_k + \varepsilon_k$$

x: Curve's parameters: In DF, etc.

→ Transform to L_1 pen: $|SP|$

$$\min |x| = \min \; u + v \quad \text{s.t} \begin{cases} u - v = x \\ u \geq 0, v \geq 0 \end{cases}$$

→ problem is

$$\min \sum u_i + v_i$$

$$\text{st } u_i - v_i = \frac{f(t_{i+1}, t_{i+2}) - f(t_i, t_{i+1})}{t_{i+2} - t_i} - \frac{f(t_i, t_{i+1}) - f(t_{i-1}, t_i)}{t_{i+1} - t_{i-1}}$$

and remaining conditions

→ We linearise $R_j(x) = R_j(x^k) + \nabla R_j(x^k)^T \nabla x$

→ solve by simplex