⑤ Procedure of solving problems

Formulating the problem $\longrightarrow$ Finding the algorithms $\longrightarrow$ Implementation (often by computer)

Example 5: Minimum Surface

Let $\Omega \subset \mathbb{R}^2$ be a given closed set with boundary $\Gamma = \partial\Omega$ and let $r : \Gamma \to \mathbb{R}$ be a function on the boundary of Ω.

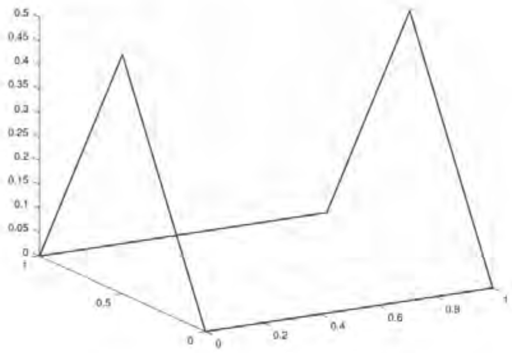
We want to find a function $q : \Omega \to \mathbb{R}$ such that:

- $q|_\Gamma = r|_\Gamma$, i.e., q coincides with the boundary data r on Γ. 边框符合要求
- The graph of q has minimum surface. 使得表面积最小

$\triangleq q(x) = r(x),\ \forall x \in \Gamma$

Example:

- $\Omega = [0,1] \times [0,1]$ and $r(x,y) = \frac{1}{2} - |y - \frac{1}{2}|$.



2. Previous related knowledge

a. Inner product $\longrightarrow$ 本质是个 Function

$\langle x, y\rangle = x^T y = \sum_{i=1}^{n} x_i y_i$ (从形式上看: Linear sum; 从结果上看: Constant)

- Positivity: $\langle x, x\rangle \geq 0$ and $\langle x, x\rangle = 0$ if and only if $x = 0$
- Symmetry: $\langle x, y\rangle = \langle y, x\rangle$
- Linearity: $\langle \alpha x + \beta y, z\rangle = \alpha\langle x, z\rangle + \beta\langle y, z\rangle$

$\longrightarrow$ 满足这3个properties的都是 inner product

b. Norm $\longrightarrow$ 本质是个 Function

$\|x\| = \|x\|_2 = \sqrt{\langle x, x\rangle} = \sqrt{x^T x} = \sqrt{x_1^2 + \cdots + x_n^2}$

- Positivity: $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = 0$
- Homogeneity: $\|\alpha x\| = |\alpha|\,\|x\|$
- Triangle inequ: $\|x + y\| \leq \|x\| + \|y\|$
- Cauchy-Schwarz inequ: $|\langle x, y\rangle| \leq \|x\|\,\|y\|$

Other norms: ("Norm" 是 L2-Norm)

L1-norm: $\|x\|_1 = \sum_{i=1}^{n} |x_i|$

Lp-norm: $\|x\|_p = \left[\sum_{i=1}^{n} |x_i|^p\right]^{\frac{1}{p}}$

maximum norm $\|x\|_\infty = \max_{1\leq i\leq n} |x_i| = \max_{1\leq i\leq n} \{x_i, -x_i\}$ 两个max合成一个

c. Supremum & Infimum 集合的最值

- Supremum: The smallest scalar y s.t. $y \geq x$, $\forall x \in \Omega$ If $\sup\Omega \in \Omega$, then $\sup\Omega = \max\Omega$
- Infimum: The largest scalar y s.t. $y \leq x$, $\forall x \in \Omega$ If $\inf\Omega \in \Omega$, then $\inf\Omega = \min\Omega$

d. Minimizer (Minima $\triangleq$ Minimizer)

If $x^* \in \Omega$, and

- loc. minimizer: $\forall x \in \Omega \cap B_\varepsilon(x^*)$, $f(x^*) \leq f(x)$
- strict loc. minimizer: $\forall x \in \Omega \cap B_\varepsilon(x^*) \setminus \{x^*\}$, $f(x^*) < f(x)$
- global minimizer: $\forall x \in \Omega$, $f(x^*) \leq f(x)$
- strict global minimizer: $\forall x \in \Omega \setminus \{x^*\}$, $f(x^*) < f(x)$

e. Linear function

- Linear: $f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$, $\forall x, y \in \mathbb{R}^n$, $\alpha, \beta \in \mathbb{R}$ Linear algebra form: $g(x) = Ax - b$
- Affine-linear: $g(x) = f(x) - b$

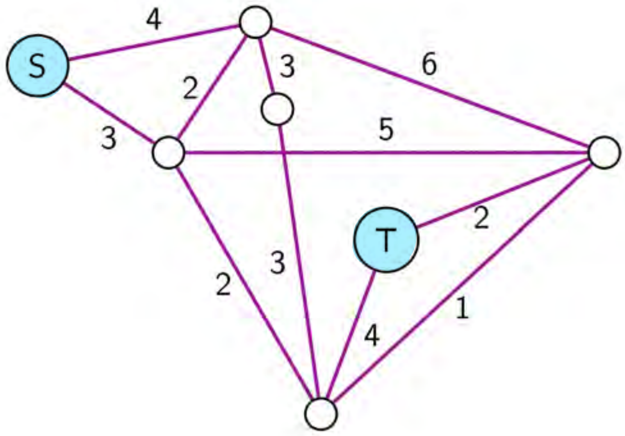
L2-3 Formulating / Modeling Opt. Problem

1. Golden rules

- Decision → Decision variable
- Objective → Objective function
- Constraints → Constraint functions

2. Examples

① Shortest paths



Some notations:

- Set of nodes : N
- Set of edges : $E \subset N \times N$
- The distance between node i and node j : w_{ij}

a. Decision variable (选什么)

$$x_{ij} = \begin{cases} 1, & \text{if we use edge } \langle i, j \rangle \\ 0, & \text{otherwise} \end{cases}$$

b. Objective function (优化什么)

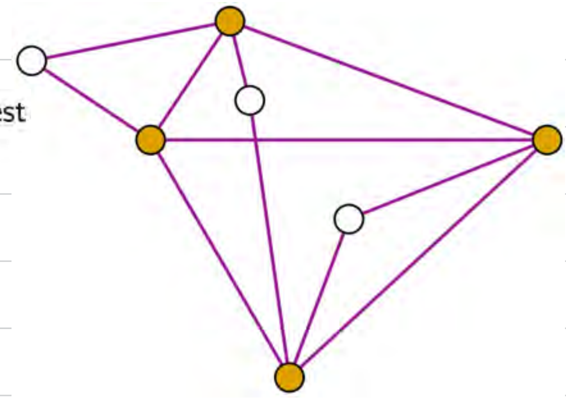
$$\min_{x_{ij}} \sum_{\langle i,j \rangle} w_{ij} x_{ij}$$

c. Constraint functions

s.t.

Domain	$x_{ij} \in \{0, 1\}, \ \forall \langle i, j \rangle \in E$	
For start pt	$\sum_j x_{sj} = 1$	起始点：一出
For termination pt	$\sum_j x_{jt} = 1$	终点：一入
For intermediate pts	$\sum_j x_{ij} = \sum_j x_{ji}, \ \forall i \neq s, t$	中间点：一入一出 / 零入零出

② Vertex covering



Task: Given a graph with nodes V and edges E, find the smallest set of vertices that touch every edge of the graph.

a. Decision variable

$$x_i = \begin{cases} 1, & \text{if we choose vertex } i \\ 0, & \text{otherwise} \end{cases}$$

b. Objective function

$$\min_{x_i} \sum_i x_i$$

c. Constraint function

s.t. Domain $x_i \in \{0,1\}, \forall i \in N$

For each path $x_i + x_j \geq 1, \forall (i,j) \in E$

③ Support vector machine (SVM)

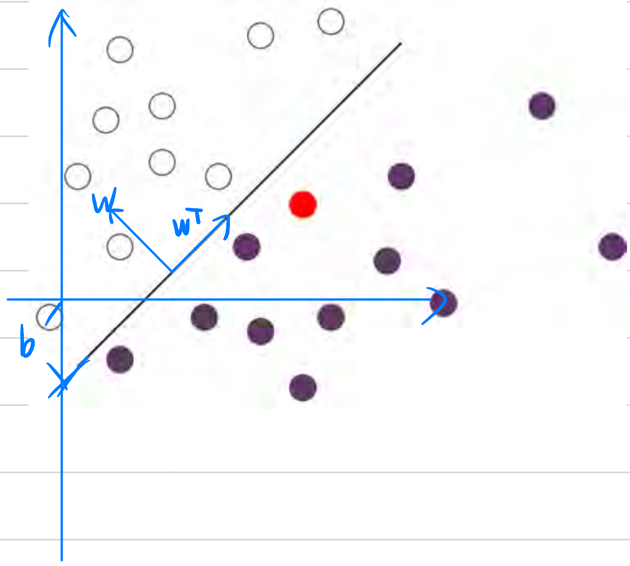
Given: m objects represented by vectors $x_1, \cdots, x_m \in \mathbb{R}^n$ with labels $y_i \in \{-1, 1\}$ (i.e. The data are separated into two classes)

Idea: Learn a function $f: \mathbb{R}^n \to \{-1, 1\}$, based on the training sample $(x_1, y_1), \cdots, (x_m, y_m)$ to do prediction

a. The feasible problem (The optimization problem where we just focuses on the constraints)

△ Decision variable

A hyperplane $f(x) := w^T x + b$: w & b — support vector



Hyperplane本质上是高维空间内的一个平面，可类比二维空间内的一条直线

Such that $y_i = \begin{cases} +1, & \text{if } f(x_i) > 0 \\ -1, & \text{if } f(x_i) < 0 \end{cases}$

This is equivalent to find:

A hyperplane $f(x) := w^T x + b$: w & b

Such that $y_i = \begin{cases} +1, & \text{if } f(x_i) \geq +1 \\ -1, & \text{if } f(x_i) \leq -1 \end{cases}$, $\forall i = 1, \cdots, m$

proof:

前→后	待证
后→前	不用证

证: We can always find a hyperplane $\tilde{f}(x) = \tilde{w}^T x + \tilde{b}$ which is rescaled from a hyperplane satisfying 前 and satisfies 后

Let's define $\begin{cases} \text{class } A = \{i : y_i = 1\} \\ \text{class } B = \{i : y_i = -1\} \end{cases}$

Since $f(x)$ satisfies 前,

then $\exists\, \delta > 0$, s.t. $f(x_i) \geq \delta$, $\forall i \in A$,

we can just set $\delta = \min_{i \in A} f(x_i)$

问题△: 常数；问题△: 不等号

Now we have

$$\begin{cases} w^T x_i + b \geq \delta, \forall i \in A \Leftrightarrow (\underbrace{\tfrac{2}{\delta} w}_{\tilde{w}})^T x_i + (\underbrace{\tfrac{2b}{\delta} - 1}_{\tilde{b}}) \geq 1, \forall i \in A \\ w^T x_i + b \leq 0, \forall i \in B \Leftrightarrow (\tfrac{2}{\delta} w)^T x_i + (\tfrac{2b}{\delta} - 1) \leq -1, \forall i \in B \end{cases}$$

△ Objective function

$\min_{w,b} \; 0$ (i.e. No objective function)

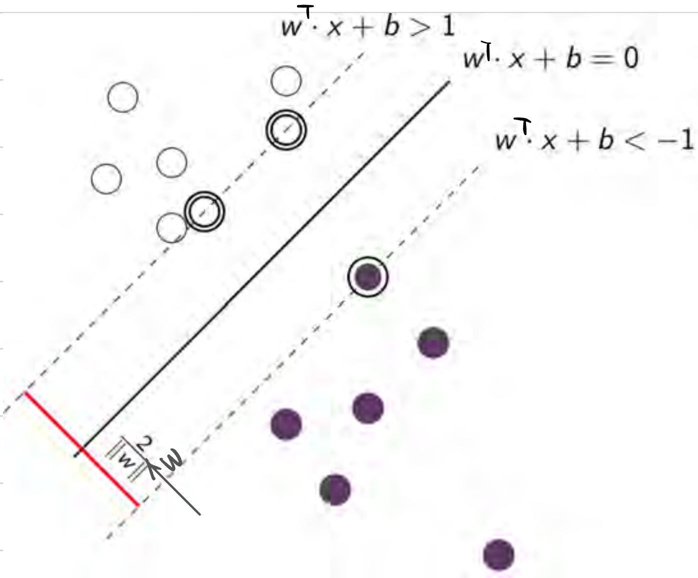
△ Constraint function

s.t. Domain $y_i (x_i^T w + b) \geq 1$, $\forall i$ (Repeated from the decision)

b. The optimization problem with perfectly-separated data

△ Decision variable

Two parallel hyperplanes $\begin{cases} \{x: w^T x + b = 1\} \\ \{x: w^T x + b = -1\} \end{cases}$: w, b



△ Objective function

$\max_{w,b} \frac{2}{\|w\|}$ $\frac{2}{\|w\|}$ is the distance between two hyperplanes

proof:

Let $\begin{cases} H_0 = \{x: w^T x + b = 0\} \\ H_1 = \{x: w^T x + b = 1\} \\ H_{-1} = \{x: w^T x + b = -1\} \end{cases}$

We let $x_0 \in H_{-1}$, and define $x(t) = x_0 + tw$

切向距离

$x(t) \in H_1 \Leftrightarrow w^T x(t) + b = 1$

$\Leftrightarrow \underbrace{w^T x_0}_{=-b-1} + t\underbrace{w^T w}_{=\|w\|^2} + b = 1$

$\Rightarrow t = \frac{2}{\|w\|^2}$

$\Rightarrow \text{Distance} = |t| \|w\| = \frac{2}{\|w\|}$

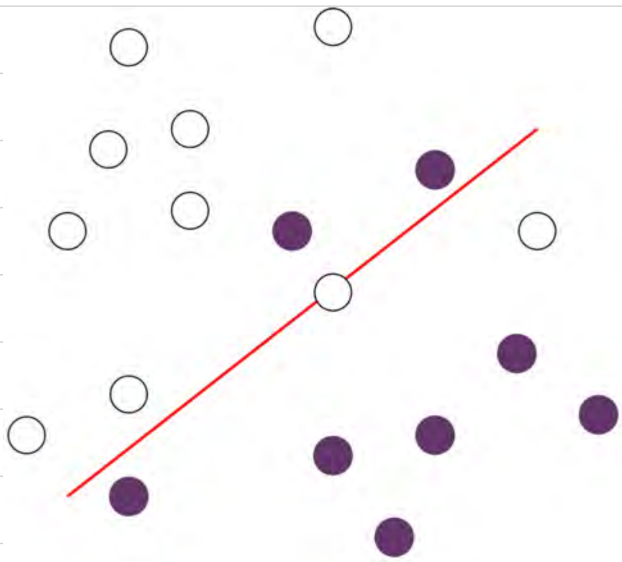
This is equivalent to:

$\min_{w,b} \frac{1}{2}\|w\|^2$ (we prefer to use $\|w\|^2$ instead of $\|w\|$ because $\|w\|$ is not differentiable at 0)

△ Constraint function

s.t. $y_i(x_i^T w + b) \geq 1, \forall i$

c. The optimization problem with not-perfectly-separated data



△ Decision variable

Two parallel hyperplanes $\begin{cases} \{x: w^T x+b=1\} \\ \{x: w^T x+b=-1\} \end{cases}$: w, b

△ Objective function

当无法很好地进行预测时，这个部分会小于1

$$\min_{w,b} \frac{\lambda}{2}\|w\|^2+\underbrace{\sum_{i=1}^{m} \max\{0, 1-\boxed{y_i(x_i^T w+b)}\}}_{\text{Hinge-Loss function 合页损失函数 (Linear function)}}, \quad \lambda \geq 0$$

λ is chosen to take balance, typically, $\lambda=\frac{1}{m}$ (当数目更多时，我们更看重误差水平)

when $\lambda=0$, it can be written as:

$$\min_{w,b,t} \sum_{i} t_i, \quad \text{where } t_i =: (1-y_i(x_i^T w+b))^+ \qquad a^+ \triangleq \max\{0, a\}$$

△ Constraint function (when $\lambda=0$)

s.t. $t_i=(1-y_i(x_i^T w+b))^+, \ \forall i$

we can relax "=" to "≥":

s.t. $t_i \geq (1-y_i(x_i^T w+b))^+, \ \forall i$

proof:

前→后：不用证 (结果) 后→前：待证	= ：不用证 > ：待证	If $t_i^* > (1-y_i(x_i^T w^*+b^*))^+$, and $t_i := (1-y_i(x_i^T w^*+b^*))^+$,
证：The optimal sol of 后 is also the optimal sol of 前	即证：$t_i^* > (1-y_i(x_i^T w^*+b^*))^+$ 不存在	then contradiction!

we can remove "+": $t_i \geq (1-y_i(x_i^T w+b))^+ \Rightarrow t_i \geq 1-y_i(x_i^T w+b)$ and $t_i \geq 0$:

s.t. $y_i(x_i^T w+b)+t_i \geq 1, \ \forall i$

$\qquad t_i \geq 0, \ \forall i$

④ Portfolio management

a. Decision variable

The proportion we invest to each securities: $x = [x_1, \cdots, x_N]^T$

b. Objective function

$$\max_{x} \ x^T u - \lambda x^T \Sigma x$$

Return: $x^T u$ $\quad u = \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix}$

Risk: $x^T \Sigma x$ $\quad \Sigma = \begin{bmatrix} Var(r_1) & \cdots & Cov(r_1, r_N) \\ \vdots & & \vdots \\ Cov(r_N, r_1) & \cdots & Var(r_N) \end{bmatrix}$

λ is small: Return is more important

λ is large: Risk is more important

c. Constraint function

s.t.
- Domain $\quad x_i \geq 0 \ , \forall i = 1, \cdots, N$
- Sum of the proportions $\quad x^T \mathbb{1} = 1$

⑤ Linear regression

a. Decision variable 实际上是个Hyperplane

A linear model $\hat{y}_i = \theta^T x_i + \theta_0$: θ, θ_0

b. Objective function

$$\min_{\theta, \theta_0} \ \frac{1}{N} \sum_{i=1}^{N} (\theta^T x_i + \theta_0 - y_i)^2$$

c. Constraint function

No.

in matrix form

a. Decision variable

$\hat{Y} = X\theta$: θ (a matrix)

b. Objective function

$$\min_{\theta} \ \frac{1}{N} \|X\theta - Y\|^2$$

$\| 向量 \|^2 = \sum (\ )^2$

where

$X = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{Nd} \end{bmatrix}$ $\quad N \times (d+1)$

$\theta^T x_i + 1 \theta_0$

$\theta = \begin{bmatrix} \theta_0 \\ \theta \end{bmatrix}$ $\quad (d+1) \times 1$

这是个向量

$Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$ $\quad N \times 1$

c. Constraint function

No.

⑥Chebyshev center

这实际上是个Polyhedron

- Consider a set P described by linear inequality Constraints, i.e. $P = \{x \in R^n \mid a_i^T x \leq b_i, i = 1, \cdots, m\}$
- We are interested in finding a ball with the largest possible radius, which is entirely contained with the set P (the center of this ball is called the **Chebyshev center** of P). Provide a concise formulation for this optimization problem.

a. Decision variable

A open ball $B_r(y)$: y, r

b. Objective function

$$\min_{y,r} (-r)$$

c. Constraint function

$$\text{s.t.} \begin{cases} y & a_i^T y \leq b_i \quad , \forall i = 1, \cdots, m \\ r & r \leq \dfrac{|a_i^T y - b_i|}{\|a_i\|} \quad , \forall i = 1, \cdots, m \end{cases}$$

⑦Nurses scheduling

- A hospital wants to make a weekly night shift (12pm-8pm) schedule for its nurses. The demand for nurses for the night shift on day j is d_j, j=1, 2, $\cdots$, 7.
- **Every nurse works 5 days in a row**. Formulating a optimization problem to find the minimal number of nurses the hospital needs to hire

a. Decision variable

The # of the nurses starting their works on day j : x_j

b. Objective function

$$\min_x \sum_{j=1}^{7} x_j$$

c. Constraint function

$$
\begin{aligned}
\text{s.t.} \quad & x_1 + x_4 + x_5 + x_6 + x_7 \geq d_1 \\
& x_1 + x_2 + x_5 + x_6 + x_7 \geq d_2 \\
& x_1 + x_2 + x_3 + x_6 + x_7 \geq d_3 \\
& x_1 + x_2 + x_3 + x_4 + x_7 \geq d_4 \\
& x_1 + x_2 + x_3 + x_4 + x_5 \geq d_5 \\
& x_2 + x_3 + x_4 + x_5 + x_6 \geq d_6 \\
& x_3 + x_4 + x_5 + x_6 + x_7 \geq d_7 \\
& x_j \geq 0, x_j \in \mathbb{Z}, \forall j = 1, 2, \cdots, 7
\end{aligned}
$$

Supply ≥ Demand

⑧ **Rod cutting**

- Assume there are multiple complete rods with fixed length of L.
- And there are m orders. For order i, it requires n_i pieces with length ℓ_i.
  - $\ell_i \neq \ell_j$ when $i \neq j$.
  - $\ell_i \in \mathbb{Z}$ and $\ell < L$.
- How to cut the rods so that we can satisfy all orders while minimize waste? Formulate the corresponding optimization problem
- First of all, we need to enumerate all possible patterns.
- As an example, assume $L = 10$, and there are two orders with $n = [5, 8]$, and $\ell = [3, 5]$.
  - There are total 3 valid patterns: $\boxed{(3, 3, 3, 1)}$, $\boxed{(3, 5, 2)}$, $\boxed{(5, 5)}$.
  - Let a_{ij} represent the number of times order i appears in pattern j.

$$\begin{bmatrix} 3 & 1 & 0 \\ 0 & 1 & 2 \end{bmatrix}$$

  - For each pattern j, we waste $w_j = L - \sum_{i=1}^{2} a_{ij} \cdot \ell_i$. (In this example, $w = [1, 2, 0]$)
  - Let x_j denote the number of rods using pattern j, $x_j \in \mathbb{Z}$.
  - For each order i, we need to ensure that $\sum_{j=1}^{3} x_j \cdot a_{ij} \geq n_i$.
  - We want to minimize the total waste $\sum_{j=1}^{3} x_j \cdot w_j$.

a. Decision variable

The # of rods using pattern j: x_j

b. Objective function

被用到的部分

$\min_x \sum_j w_j x_j$, where w_j (the waste of pattern j) $= L - \boxed{\sum_j a_{ij} l_i}$

c. Constraint function

s.t. Domain: $x_j \geq 0$, $x \in \mathbb{Z}$, $\forall j = 1, \cdots, n$ n 是 pattern 数

Orders: $\sum_j a_{ij} x_j \geq n_i$, $\forall i = 1, \cdots, m$ m 是 order 数

已知长度L
已知需求orders
→ 穷举patterns → 以每种pattern所使用的材料数作为DV →
OF: 浪费总和最小化
CF: 满足orders

L4 Adjusting LP

1. Forms of LP

① **General form**

$\boxed{I^T y = \sum y_i}$

$$\min_x \quad c^T x \quad (\triangleq c^T x + b)$$

s.t. $a_i^T x \geq b_i, \ \forall i \in M_1$ (m_1个不等式)

$a_i^T x \leq d_i, \ \forall i \in M_2$

$a_i^T x = e_i, \ \forall i \in M_3$

$x_i \geq 0, \ \forall i \in N_1$

$x_i \leq 0, \ \forall i \in N_2$

x_i free, $\forall i \in N_3$

(A more compact way)

$$\min_x \quad c^T x$$

s.t. $A_1 x \geq b$ ← 是一个方程组

$A_2 x \leq d$

$A_3 y = e$

$x_i \geq 0 \quad \forall i \in N_1$

$x_i \leq 0 \quad \forall i \in N_2$

x_i free $\quad \forall i \in N_3$

② **Standard form** (General LP → Standard LP)

$$\min_x \quad c^T x$$

s.t. $Ax = b$

$x \geq 0$

a. 检查Object function: min Linear function

b. 检查Linear constraint: equality

c. 检查Variable constraint: ≥ 0 (若不是0, 则为Linear constraint)

所有variables, 包括自己定义的slack variables, 都要≥ 0!!!

a. 改正方法: $c \longrightarrow -c$

b. 改正方法: $\begin{cases} Ax \leq b \longrightarrow Ax + s = b, \ s \geq 0 \\ Ax \geq b \longrightarrow Ax - s = b, \ s \geq 0 \end{cases}$ s is called "slack variable"

c. 改正方法: $\begin{cases} x_i \leq 0 \longrightarrow \text{Define } y_i = -x_i \geq 0 \\ x_i \text{ free} \longrightarrow x_i = x_i^+ - x_i^-, \text{ with } x_i^+, x_i^- \geq 0 \end{cases}$ 此时要换元!!!

$\begin{cases} x_i^+ = \max\{0, x_i\} \\ x_i^- = -\min\{0, x_i\} \end{cases}$ ⇒ Both are positive!

e.g. $x = \begin{pmatrix} 1 \\ 0 \\ -2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}$

Examples

$$\max_x \sum_{j=1}^n c_j x_j$$

s.t. $\sum_{j=1}^n a_{ij} x_j \leq b_i, \ i \in I$

$\sum_{j=1}^n a_{ij} x_j = b_i, \ i \in E$

$x_j \geq 0, \ j \in S$

$\Longrightarrow$

$$\min_x \ -\sum_{j \in S} c_j x_j - \sum_{j \notin S} c_j (x_j^+ - x_j^-)$$

s.t. $\sum_{j \in S} a_{ij} x_j + \sum_{j \notin S} a_{ij} (x_j^+ - x_j^-) + s_i = b_i, \ i \in I$

$s_i \geq 0, \ i \in I$

$\sum_{j \in S} a_{ij} x_j + \sum_{j \notin S} a_{ij} (x_j^+ - x_j^-) = b_i, \ i \in E$

$x_j \geq 0, \ j \in S$

$\boxed{\begin{array}{l} x_j^+ \geq 0, \ j \notin S \\ x_j^- \geq 0, \ j \notin S \end{array}}$ 非常小心有遗落的!

2. Reformulating (NLP → LP)

① Process

a. Introduce auxiliary variables (使得 objective function 变成 Linear function) ⇒ 此时会使得 complexity 增加

b. Relax binding constraints ⟵ Modeling Tool

② Examples

SVM ($\lambda=0$), maximin problem, minimax problem, absolute values (可以用图形理解)

本质上

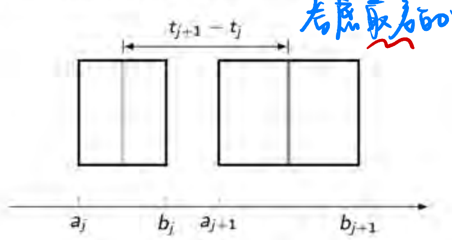
本质上

a. Maximin problem

An air traffic controller needs to control the landing times of n aircrafts:

- Flights must land in the order $1, \dots, n$.
- Flight j must land in time interval $[a_j, b_j]$.
- The objective is to maximize the minimum separation time, which is the interval between two landings.

考虑最差的情况



△ Original (NLP)

Decision variable: t_j: the landing time of the flight j

Opt problem:
$$\max_{t} \min_{j=1,\dots,n-1} \{t_{j+1} - t_j\}$$
$$\text{s.t.} \begin{cases} a_j \le t_j \le b_j, & j=1,\dots,n \\ t_j \le t_{j+1}, & j=1,\dots,n-1 \end{cases}$$

△ Reformulated (LP)

We define $\Delta := \min_{j=1,\dots,n-1} \{t_{j+1} - t_j\}$.

⇒ Opt problem:
$$\max_{t,\Delta} \ \Delta$$
$$\text{s.t.} \begin{cases} \Delta \le t_{j+1} - t_j, & j=1,\dots,n-1 \\ a_j \le t_j \le b_j, & j=1,\dots,n \\ t_j \le t_{j+1}, & j=1,\dots,n-1 \end{cases}$$

proofs: Two representations are equivalent! (解集相同)

前→后: V

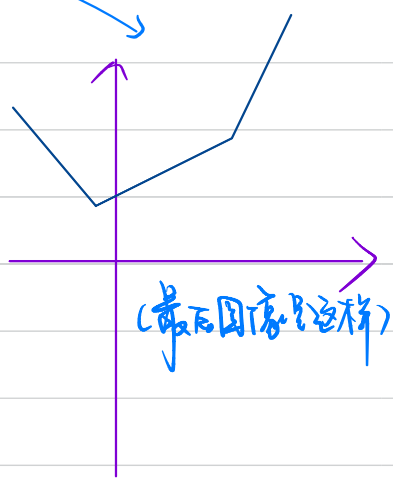
后→前: $\begin{cases} = : V \\ < : \text{if } \Delta^* < t_{j+1}^* - t_j^*, \text{ then } \Delta^* < \min_j \{t_{j+1}^* - t_j^*\}, \\ \quad \text{but we can set a } \bar{\Delta} = \min_j \{t_{j+1}^* - t_j^*\} > \Delta^* \\ \quad \Rightarrow \text{"<" 的情况不存在} \end{cases}$

⇒ 前与后的解集相同

b. Minimax problem

△ Original (NLP)

$$\min_x \max_{i=1,\dots,n} \{c_i^T x + d_i\}$$
$$\text{s.t.} \quad Ax = b$$
$$x \geq 0$$



(最后图像是这样)

△ Reformulated (LP)

We define $y = \max_{i=1,\dots,n} \{c_i^T x + d_i\}$.

$$\Rightarrow \min_{x,y} \; y$$
$$\text{s.t.} \quad y \geq c_i^T x + d_i, \; \forall i$$
$$Ax = b$$
$$x \geq 0$$

c. Absolute value $|x_i| = \max\{x_i, -x_i\}$

△ Original (NLP)

$$\min_x \sum_{i=1}^n |x_i|$$
$$\text{s.t.} \quad Ax = b$$

△ Reformulated (LP)

We define $t_i = |x_i|$.

$$\Rightarrow \min_{x,t} \sum_{i=1}^n t_i$$
$$\text{s.t.} \quad \boxed{\begin{array}{l} t_i \geq x_i \\ t_i \geq -x_i \end{array}}$$
$$Ax = b$$

Relax

$t_i = |x_i| \Rightarrow |x_i| \leq t_i \Rightarrow \begin{cases} t_i \geq x_i \\ t_i \geq -x_i \end{cases}$

③ Modeling Tool

If two problems have the same optimal solution x^* and the same optimal value, then they are equivalent.

$$\min_x \sum_{i=1}^n f_i(x) \quad \xleftrightarrow{\text{Equivalent}} \quad \min_{x,t} \sum_{i=1}^n t_i$$
$$\text{s.t. } x \in \Omega \qquad\qquad \text{s.t. } x \in \Omega$$
$$t_i \geq f_i(x), \; \forall i$$

proof:

$\rightarrow$: if x^* is the opt. sol. of problem 1, then $f(x^*) \leq f(x) \leq t_i, \; \forall i$,
then $t^* = f(x^*)$ is optimal, then $(x^*, t^*) = (x^*, f(x^*)) \Rightarrow \checkmark$

$\leftarrow$: if (x^*, t^*) is the opt. sol of problem 2, then $f(x^*) \leq t^*$, $x^* \in \Omega$
Since $t^* \leq t_i, \forall i$, then we have $f(x^*) \leq t^* \leq t = f(x)$.
then x^* is the opt. sol of problem 1 $\Rightarrow \checkmark$

3. Fractional programming

① Process

a. Transform the objective function

b. Transform the definition

c. Transform the constraints (注意符号限制)

② Example

△ Original

$$\min_{x} \frac{c^T x + d}{e^T x + f}$$

$$\text{s.t.} \quad Ax \le b$$

△ Reformulated

We define $y = \frac{x}{e^T x + f}$, $z = \frac{1}{e^T x + f}$

then:

$$\begin{aligned} \min_{y,z} \quad & c^T y + dz \\ \text{s.t.} \quad & e^T y + f z = 1 \\ & Ay - bz \le 0 \\ & z \geq 0 \end{aligned}$$

L5 Simplex method: Graphic Solution

1. Overview of Transformation

Objective function ⟶ Contour lines

Constraints ⟶ Polyhedron
- Active 可取等
- Inactive 不可取等

Optimal sol ⟶ Extreme point / Vertex / Corner (顶点和边都是最突出的点)

2. Concepts

① Polyhedron

$\{x \in R^n : Ax \geq b\} \Rightarrow$ Polyhedron 凹凸图形

即：一个有符号限制的线性方程组 (可以是 unbounded) 当 Bounded, 此时叫 "Polytope"

Finite　　Linear

(a) ✗ the set of all $(x, y) \in \mathbb{R}^2$ satisfying the constraints.　　Infinite

$$x\cos\theta + y\sin\theta \leq 1 \qquad \forall\theta \in [0, \pi/2].$$

(b) ✗ $S = \{(x, y) \in \mathbb{R}^2 \mid x^2 - 8x + 15 \leq y\}$.　　Non-linear

(c) ✓ $\emptyset$.

(d) ✗ $S = \left\{x \in \mathbb{R}^n \mid x \geq 0, x^\top y \leq 1, \text{ for all y with } \|y\|_2 = 1\right\}$.　　Non-linear

proof:

Let $S' = \{x \in R^n \mid x \geq 0, \|x\|_2 \leq 1\}$, then $S = S'$

- $S \subseteq S'$: if $x \in S$
  - if $x = 0 \Rightarrow x \in S'$
  - if $x \neq 0$, $y = \frac{x}{\|x\|_2}$ (单位方向向量), $x^T y = \frac{x^T x}{\|x\|_2} = \frac{\|x\|_2^2}{\|x\|_2} = \|x\| \Rightarrow x \in S'$
- $S' \subseteq S$: if $x \in S'$, $y \in R^n$, $\|y\|_2 = 1$, $x^T y \leq \|x\|_2 \|y\|_2 \leq 1 \Rightarrow x \in S$

柯西不等式 ($\|x\|_2 = \|x\|$)

Since S' is not a polyhedron, then S is not a polyhedron

a. The constraints in standard form construct a polyhydron:

$$\begin{cases} Ax = b \\ x \geq 0 \end{cases} \Rightarrow \begin{cases} Ax \geq b \\ Ax \leq b \\ I \cdot x \geq 0 \end{cases} \Rightarrow \begin{cases} Ax \geq b \\ -Ax \geq -b \\ I \cdot x \geq 0 \end{cases} \Rightarrow \underbrace{\begin{bmatrix} A \\ -A \\ I \end{bmatrix}}_{\text{New } A} x \geq \underbrace{\begin{bmatrix} b \\ -b \\ 0 \end{bmatrix}}_{\text{New } b}$$

b. Polyhedron is a convex set

proof:

Let $x_i \in S$, we have $Ax_i \geq b$

Let $x_j \in S$, we have $Ax_j \geq b$

then let $\lambda \in [0, 1]$, $\lambda x_i + (1-\lambda)x_j \geq \lambda b + (1-\lambda) b = b \Rightarrow \lambda x_i + (1-\lambda)x_j \geq b \Rightarrow$ Polyhedron is a convex set

② Convex set

A set S is convex iff $\forall x, y \in S$ and $\lambda \in [0,1]$, $\lambda x + (1-\lambda) y \in S$

即：任意集合中的两点的convex combination都在集合中，则这个集合为Convex set

③ Convex combination

$\forall x_1, \cdots, x_n$ and $\lambda_1, \cdots, \lambda_n \geq 0$ with $\sum \lambda_i = 1$, then $\sum \lambda_i x_i$ is a convex combination of $x_1, \cdots, x_n$

④ Extreme point

In a polyhedron, x is an extreme point if we cannot find other two points y, z, s.t. $x = \lambda y + (1-\lambda) z$, where $\lambda \in [0,1]$

前提！Extreme point只存在于Polyhedron中！ 所以我们在前面先定义了Polyhedron

即：x is an Extreme point $\triangleq$ x is not a convex combination of other two points

L6 Simplex method: BS & BFS

1. Introduction

It's hard for us to directly find extreme points, then we convert it algebraically.

2. BS

① Assumptions ☆ (Constraints是一个好的 Polyhedron)

A1: LP is in standard form

$$\min_{x} \; c^T x$$
$$\text{s.t.} \; \boxed{\begin{matrix} Ax=b \\ x \geq 0 \end{matrix}}$$

从而保证我们的Constraints被转化成Polyhedron的形式

m constraints
n variables

A2: A is a $m \times n$ matrix where $m<n$ 我们想要在constraint部分获得一个区域,而非一个解

↑包括

A3: A has linearly independent rows / A is of full rank m $(m<n)$

列(大秩)之间可以不independent,但行(小秩)之间必须independent ⇒ 否则 $\begin{cases} \text{有多余constraints} \\ \text{无解} \end{cases}$

② Definition

Under the assumptions above, x is a BS iff

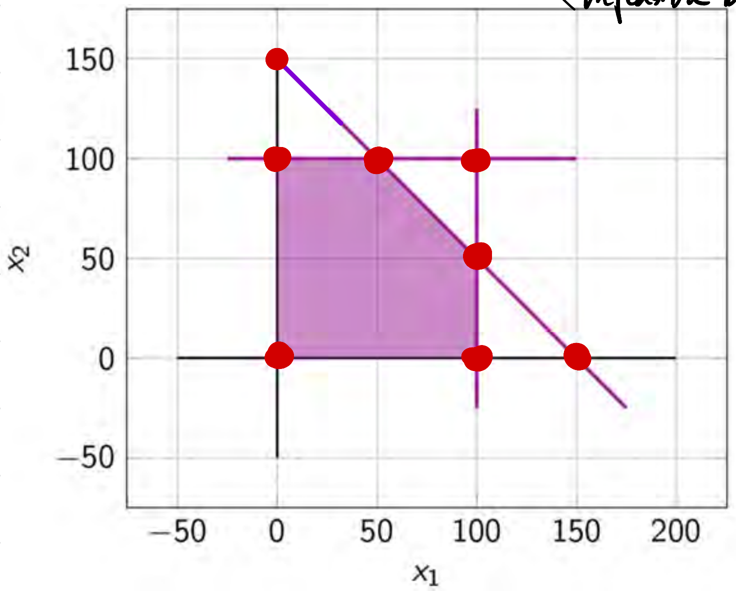
$$\begin{cases} Ax=b \quad \text{(满足第一个Constraint)} \\ \text{For } i=B(1),\cdots,B(m)\text{: The columns } A_{B(1)},\cdots,A_{B(m)} \text{ are linearly independent} \\ \text{For } i\neq B(1),\cdots,B(m)\text{: } \underline{x_i=0} \quad \text{目的是消除冗余的,即dependent的列.} \end{cases}$$

There are some denotations:

- Basic / Non-basic index
- Basic / Non-basic col
- Basic / Non-basic matrix
- Basic / Non-basic variable

③ Graphic meaning

The intersections of any two constraints
- Feasible BS : 有交点
- Infeasible BS : 没有交点 (constraints 平行)



④ Procedure to find a BS

Step 1: Check standard form and express $Ax=b$ in matrix form

Step 2: Find m (行数) independent columns: $A_{B(1)}, \cdots, A_{B(m)}$ → 因为A is of Full rank m, 所以一组ind cols 内最多是 m个 cols

Step 3: Let $x_i=0$ for $i \neq B(1), \cdots, B(m)$

Step 4: Solve $A_B x_B = b \Rightarrow$ get x_i for $i = B(1), \cdots, B(m)$

Combine to get the final x (BS)

此时A_B一定是个square matrix　此时x_B是unique，因为A_B的Basic cols是linearly independent

⑤ Some countings

a. If there are m constraints (即A的行数为m), then there are (no more than m) non-zeros in a BS

m
- independent 部分
- dependent 部分 $\Rightarrow x_i = 0$

m = 行数 = Basic cols 数

b. If $A_{m \times n}$, then there are at most (C_n^m) BS
$C(n,m)$

→ The # of iterations $\leq C_n^m$

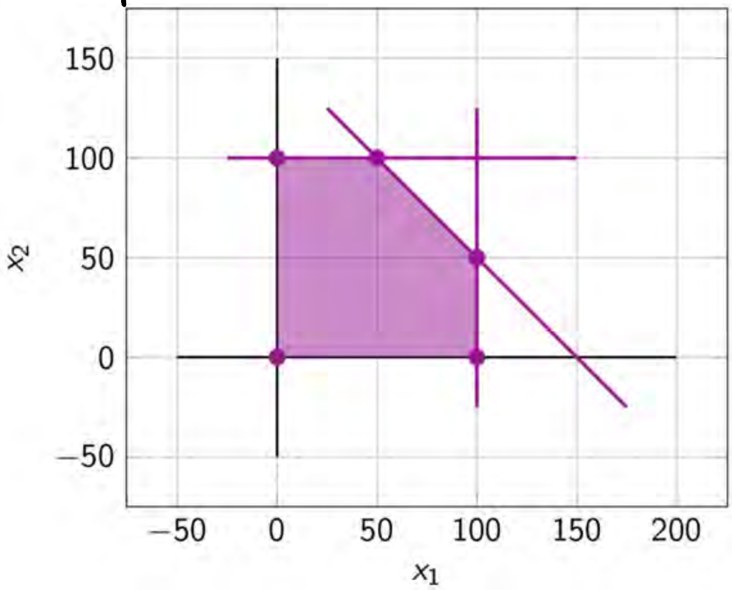
3. BFS

① Definition

BFS : BS that $x \geq 0$

② Graphic meaning

Extreme points



③ Procedure to find a BFS

Step 1 : Find all BSs.

Step 2 : Select BSs as BFSs that $x \geq 0$

Example

$$\max_{x} \quad x_1 + x_2$$

$$\text{s.t.} \quad x_1 \le 100$$

$$2x_2 \le 200$$

$$x_1 + x_2 \le 150$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

Step 1: Find BFSs

Step 1.1: Convert it into standard form

$$\min_{x,s} \quad -x_1 - x_2$$

$$\text{s.t.} \quad x_1 + s_1 = 100$$

$$s_1 \geq 0$$

$$2x_2 + s_2 = 200$$

$$s_2 \geq 0$$

$$x_1 + x_2 + s_3 = 150$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

i.e.
$$\begin{aligned} \min_{x,s} \quad & -x_1 - x_2 \\ \text{s.t.} \quad & x_1 + s_1 = 100 \\ & 2x_2 + s_2 = 200 \\ & x_1 + x_2 + s_3 = 150 \\ & x_1, x_2, s_1, s_2, s_3 \geq 0 \end{aligned}$$

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix} \quad b = \begin{bmatrix} 100 \\ 200 \\ 150 \end{bmatrix}$$

Step 1.2. Find ind. cols

e.g. index = $\{1, 2, 3\}$.

Step 1.3. Supplement $x_i = 0$ when $i \notin$ index

$$x = \begin{bmatrix} x_1 \\ x_2 \\ s_1 \\ 0 \\ 0 \end{bmatrix}$$

Step 1.4. Solve $A_B x_B = b$.

$$x_B = A_B^{-1} b$$

$$= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 100 \\ 200 \\ 150 \end{bmatrix}$$

$$= \frac{1}{\det(A_B)} \operatorname{adj}(A_B) \begin{bmatrix} 100 \\ 200 \\ 150 \end{bmatrix}$$

$$= \frac{1}{-2} \begin{bmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{bmatrix} \begin{bmatrix} 100 \\ 200 \\ 150 \end{bmatrix}$$

$$= \frac{1}{-2} \begin{bmatrix} 0 & 1 & -2 \\ 0 & -1 & 0 \\ -2 & -1 & 2 \end{bmatrix} \begin{bmatrix} 100 \\ 200 \\ 150 \end{bmatrix}$$

$$= \begin{bmatrix} 50 \\ 100 \\ 50 \end{bmatrix}$$

$$\Rightarrow x = \begin{bmatrix} 50 \\ 150 \\ 50 \\ 0 \\ 0 \end{bmatrix}$$

... (Get more BFSs)

$\Downarrow$

Indices	{1, 2, 3}	{1, 2, 4}	{1, 2, 5}
Solution	(50, 100, 50, 0, 0)	(100, 50, 0, 100, 0)	(100, 100, 0, 0, -50)
Objective	-250	-200	infeasible
Indices	{1, 3, 4}	{1, 4, 5}	{2, 3, 4}
Solution	(150, 0, -50, 200, 0)	(100, 0, 0, 200, 50)	(0, 150, 100, -100, 0)
Objective	infeasible	-100	infeasible
Indices	{2, 3, 5}	{3, 4, 5}	
Solution	(0, 100, 100, 0, 50)	(0, 0, 100, 200, 150)	
Objective	-200	0	

Step 2.1 (Finished)

Step 2.2 Select as BFSs.

Indices	{1, 2, 3}	{1, 2, 4}	{1, 2, 5}
Solution	✓ (50, 100, 50, 0, 0)	✓ (100, 50, 0, 100, 0)	✗ (100, 100, 0, 0, -50)
Objective	-250	-200	infeasible
Indices	{1, 3, 4}	{1, 4, 5}	{2, 3, 4}
Solution	✗ (150, 0, -50, 200, 0)	✓ (100, 0, 0, 200, 50)	✗ (0, 150, 100, -100, 0)
Objective	infeasible	-100	infeasible
Indices	{2, 3, 5}	{3, 4, 5}	
Solution	✓ (0, 100, 100, 0, 50)	✓ (0, 0, 100, 200, 150)	
Objective	-200	0	

4. Fundamental LP Theorem

不满足则 Fundamental LP Theorem 不成立

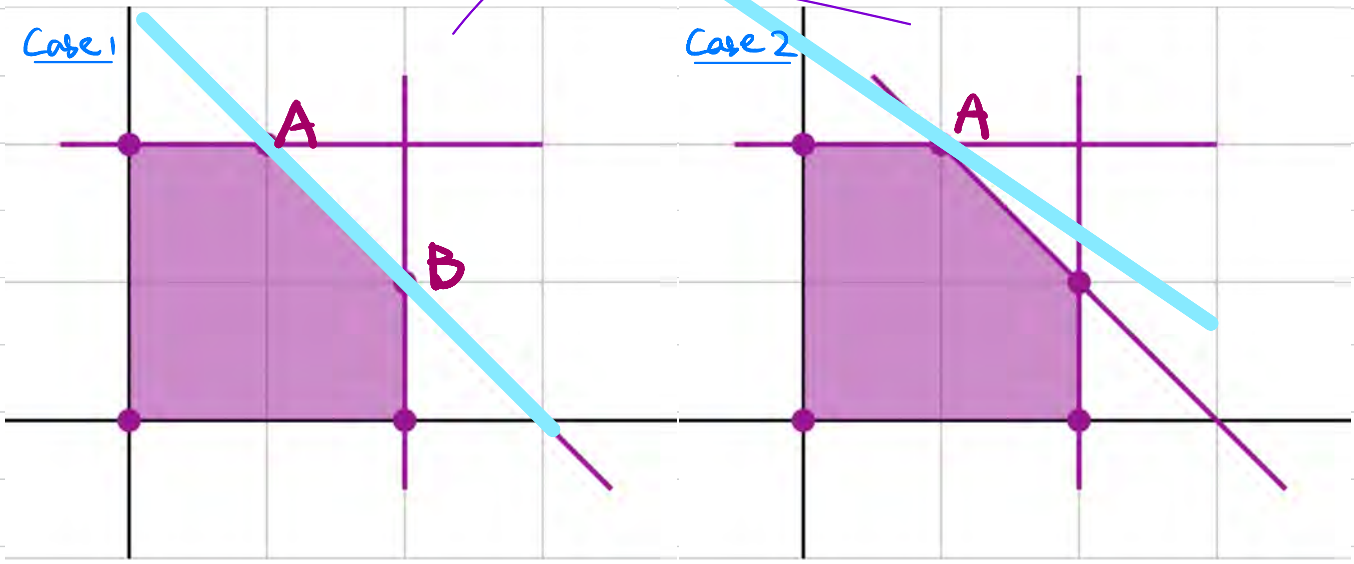
If (LP is in standard form
   A is of full rank m

(即：满足了上述求BFS前的三个assumptions)

↓ 否则 statement 1 可能不成立, e.g. $\{x \mid x_2 \geq x_1\}$: (No BFS)

then (Statement 1: If the feasible set is no empty, there is a BFS (Polyhedron 必是凸的，因此必有顶点)
   Statement 2: If there is an optimal solution, there is an optimal solution that is also a BFS.

→ 当有 opt. sol. 时，opt. sol (BFS + 非 BFS
   BFS



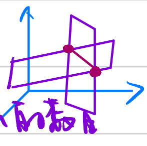
即：我们可通过求 BFS 来求 opt. sol.

Exercises: (Assumptions are satisfied)

✓ a. If $n = m+1$, then there are at most 2 BFS

(m: # of equality constraints
  n: # of variables

→ $m = n-1$ → e.g.



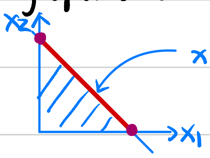
(m constraint 的交点
  n constraint 的个数

(注意，(2D 空间下，equality constraint 是条线
   3D 空间下，equality constraint 是个平面)

✗ b. The set of optimal solution is bounded

✗ c. At every optimal solution, no more than m variables can be positive

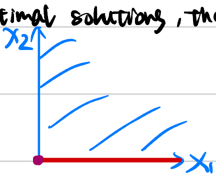
反例：



$x_1 > 0$, $x_2 > 0$ → $2 > 1$ (m)

✗ d. If there are several optimal solutions, then there are at least 2 BFS

反例： min x_2
   s.t. $x_1, x_2 \geq 0$



, 此时只有一个 BFS

L7-8 Simplex method: Detailed Process

1. Motivation

BFSs → An Optimal solution

By Fundamental LP Theorem, We can find one optimal solution in BFSs

只需要找到一个就够了

- 全局遍历: ×
- 局部有针对性的遍历: √ 从一个BFS出发

2. Neighboring BFS

① Definition

BFSs are neighboring iff they differ by exactly one index

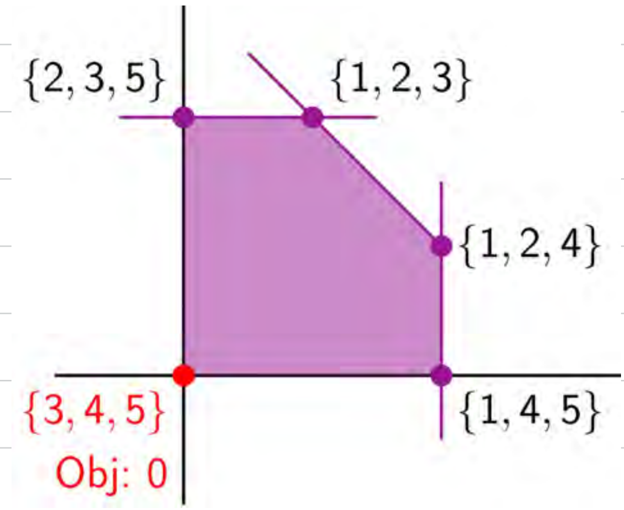
但 x_i 可能变多个

e.g.
- $\{1,2,3\} \to \{1,2,4\}$ √
- $\{1,2,3\} \to \{1,4,5\}$ ×

(注意: BFS移动后必是BFS)

② Graphic explanation

Neighboring BFSs are Graphically neighboring



3. Basic direction

换index → 换x

Moving to a neighboring BFS $\triangleq$ Moving a nondegenerate BFS x to $(x + \theta^* d)$, where $\theta \geq 0$ 前进

where $d = \begin{bmatrix} d_B \\ d_N \end{bmatrix}$, $\begin{cases} d_B = -A_B^{-1} A_j \\ d_N = [0, \cdots, 1, \cdots, 0], \text{ where } d_j = 1 \end{cases}$ At this time, d is called j^{th} basic direction.

proof: (For statement 1)

Assume the index change is: $\{B(1), \cdots, B(l), \cdots, B(m)\} \to \{B(1), \cdots, B(j), \cdots, B(m)\}$

and let $y = x + \theta^* d$

(i) $y_{B(l)} = 0$ (即：已成功把原index移走)

当有多个B(l)满足条件时，使用smallest index rule

$y_{B(l)} = x_{B(l)} + \theta^* d_{B(l)} = x_{B(l)} + \boxed{\underbrace{\min_{\{i: d_i < 0\}} \left(-\frac{x_i}{d_i}\right) d_{B(l)}}_{\text{此时 } i = B(l)}} = 0$

(ii) $y_{B(j)} \geq 0$ (即：已成功把新index加入)

不仅不等于0，还必定positive / = 0

proof: (For statement 2)

If x is degenerate, i.e. $x_{B(i)} = 0$ for some i,

then $\theta^* = \min_{\{i: d_i < 0\}} \left(-\frac{x_i}{d_i}\right) = 0$

$\to \Delta$ in Objective function $= 0$

$\to$ We probably do not move x to $(x + \theta^* d)$

proof: (For statement 3)

New BFS should also satisfy the first constraint "$A(x + \theta d) = b$",

$A(x + \theta d) = Ax + \theta A d = b + \theta A d = b$

$\Rightarrow Ad = 0 \Rightarrow [A_B \ A_N] \begin{bmatrix} d_B \\ d_N \end{bmatrix} = A_B d_B + A_N d_N = A_B d_B + A_j = 0 \Rightarrow d_B = -A_B^{-1} A_j$ (A_B is invertible)

4. Reduced cost

① Definition

Δ in Objective function $= c^T(x+\theta d) - c^T x = \theta c^T d$

Then we define $\bar{c}_j = c^T d = [c_B^T \ c_N^T] \begin{bmatrix} d_B \\ d_N \end{bmatrix} = c_B^T d_B + c_j = -c_B^T A_B^{-1} A_j + c_j = c_j - c_B^T A_B^{-1} A_j$

(表示当我们换向 j^{th} Basic direction 时的 Reduced cost) (注意：Δ in Objective function 和 Reduced cost 之间相差了 θ)

② Property

Positive reduced cost : ✗

Negative reduced cost : ✓

③ Extreme case

If $j = B(i)$, (i.e. 此时是 Basic variable 的 Reduced cost) 自己换自己 ⟶ 没有换 ⟶ $\bar{c}_j = 0$

then $\bar{c}_{B(i)} = 0$

proof:

$\bar{c}_{B(i)} = c_{B(i)} - c_B^T A_B^{-1} A_{B(i)}$

$= c_{B(i)} - c_B^T (A_B^{-1} A_{B(i)})$

$= c_{B(i)} - c_B^T e_i$ $\quad e_i$: i^{th} unit vector with 1 at i^{th} entry and 0, otherwise — $\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$

$= c_{B(i)} - c_{B(i)}$

$= 0$

④ Stopping criterion

If $\bar{c} \geq 0$, then at this time, x is the optimal solution (因为我们只需遍历顶点) $\qquad \bar{c} \geq 0 \longrightarrow \Delta \geq 0$

If $\bar{c} > 0$, x is the unique optimal solution

If x is the unique optimal solution and nondegenerate, then $\bar{c} > 0$ $\qquad \Delta \geq 0 \not\longrightarrow \bar{c} \geq 0$

在这里强调"Nondegenerate"是因为：当 x^* is degenerate, $\theta^* = 0$, 无论 $\bar{c}$ 的取值范围如何, Δ in Objective function value $= 0$

proof:

Let y be an arbitrary feasible solution and we define $d = y - x$

Since $Ax = Ay = b$, then $Ad = A(y-x) = 0$

So $A_B d_B + \sum_{i \in N} A_i d_i = 0 \Rightarrow d_B = -\sum_{i \in N} A_B^{-1} A_i d_i$

Δ in Objective function $= c^T(y-x) = c^T d = c_B d_B + \sum_{i \in N} c_i d_i = \sum_{i \in N} (c_i - c_B A_B^{-1} A_i) = \sum_{i \in N} \bar{c}_i d_i$

Since $\bar{c}_i \geq 0$, $d_i \geq 0$, then Δ in Objective function $= \sum_{i \in N} \bar{c}_i d_i \geq 0$

$\Rightarrow x$ is optimal

5. Stepsize

θ is the stepsize. (Constant)

After we compute the jth Basic direction d and figure out that $\bar{c}_j < 0$,

we want to go in that direction as far as possible, i.e. maximize θ

then $\theta^* = \max\{\theta \geq 0 : x + \theta d \geq 0\}$

因为Basic direction的定义里已经包含了1st constraint，所以在这里我们只需考虑2nd constraint.

$$\begin{cases} d \geq 0 \longrightarrow \theta^* = +\infty \rightarrow LP \text{ is unbounded} \\ d_i < 0 \text{ for some } i \longrightarrow \theta^* = \min_{\{i : d_i < 0\}} -\frac{x_i}{d_i} \end{cases}$$

proof:

$\theta^* = \max\{\theta \geq 0 : x + \theta d \geq 0\}$

$$\begin{cases} \text{for } d_i \geq 0 : x_i + \theta d_i \geq 0 \quad (\text{一直成立}) \\ \text{for } d_i < 0 : \text{Let } x_i + \theta^* d_i \geq 0, \text{ then } \theta^* \leq -\frac{x_i}{d_i}, \ \forall i \text{ where } d_i < 0 \end{cases} \Rightarrow \theta^* = \min_{\{i : d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$$

注意要变号(因为$d_i<0$)

6. An iteration

Step 1: Compute Reduced cost $\bar{c}_j$ $\begin{cases} \bar{c}_j \geq 0 \text{ for all } j \rightarrow \text{Stop} \\ \bar{c}_j < 0 \rightarrow \text{Step 2} \end{cases}$

Step 2: Compute the relative jth Basic direction d $\begin{cases} d \geq 0 \rightarrow LP \text{ is unbounded} \quad (\theta^* = \infty) \\ d_i < 0 \text{ for some } i \longrightarrow \text{Step 3} \end{cases}$

Step 3: Compute maximal stepsize θ^*, and set $y = x + \theta^* d$

$\longrightarrow$ Repeat step 1 to step 3 until we reach a BFS where $\bar{c}_j > 0$ for all j

If the constraint is not empty and every BFS is nondegenerate

then the # of iterations are finite

$\leq C(n, m)$

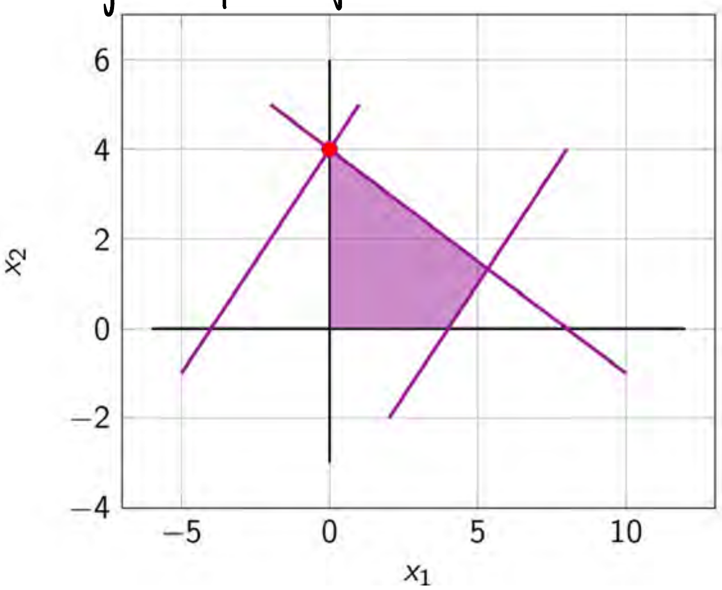
7. Degeneracy

① Definition

A BFS is degenerate iff $x_{B(i)} = 0$ for some i.

② Graphic explanation

At a degenerate BFS, # of intersectional constraints > Dimension



8. Bland's rule

① Motivation

Assume we come to a degenerate BFS,

and after our computation, $\bar{c}_j < 0$ & $\underline{\theta^* = 0}$,

(在Degenerate BFS时，θ^*必须等于0：$\theta^* = \min_{\{i: d_i < 0\}} \left(-\frac{x_i}{d_i}\right)$，因为 $\begin{pmatrix} x_i > 0 \\ x_i = 0 \end{pmatrix}$，所以$\theta^* = 0$)

we need to set rules of finding Entering index & Exiting index to avoid any cycle

② Definition

If we use the <u>smallest index rule</u> both for entering index (basis) and exiting index (basis),

then there is no cycle. ⟶ 我们能够离开Degenerate BFS ⟶ <u>在有限步下完成iteration</u> (Convergence theorem)

<u>Smallest index rule</u>: Choose the smallest index j with $\bar{c}_j < 0$

9. Finding an initial BFS

① Motivation

If we just test BS whether BS ≥ 0 one by one,
then it takes lots of time

(因为求BS需要那三个步骤，且其中的 $x_B = A_B^{-1} b$ 求逆矩阵非常耗时间)

$\longrightarrow$ We want some methods that can help us check the feasibility before officially computing BFS

Upside: Easily initiate a BFS
Downside: Use simplex method twice.

② Two-phase method

a. Phase 1 (用于判断Original Problem是否feasible / 是否有BFS)

△ Introduce an auxiliary LP with $\boxed{b \geq 0}$ (如果 $b<0$，就两边同乘 -1) → 使得 $(0, y)$ is feasible

$$\min_x \; c^T x \qquad\qquad \min_{x,y} \; \mathbf{1}^T y$$
$$\text{s.t. } Ax = b \quad \longrightarrow \quad \text{s.t. } Ax + y = b$$
$$x \geq 0 \qquad\qquad x \geq 0$$
$$y \geq 0$$

(注意此时 y 也是个vector) $\longrightarrow$ $\begin{cases} x: n \times 1 \\ y: m \times 1 \end{cases}$

本质上，在这以处 $\begin{cases} x = x_N \\ y = x_B \end{cases}$, $\begin{cases} A_N = A \\ A_B = I \end{cases}$

△ Solve the auxiliary LP with simplex method (Start with $(0, y)$)

我们从这个 x 开始

△ Check feasibility
- if the optimal value of the auxiliary LP is 0 $\longrightarrow$ Feasible $\longrightarrow$ This x is a BFS
- if the optimal value of the auxiliary LP is positive $\longrightarrow$ Infeasible $\longrightarrow$ 不用解了 (无BFS)

b. Phase 2

Check degeneracy
- if x^* (对于auxiliary LP) is a Nondegenerate BFS: Start with this BFS
- if x^* is a Degenerate BFS: 在 x 的index里找index加入Basic index，使得 A_B 是 independent

提示：我们不一定一直要用Two-Phase Method，
- 出现简单的 A_B 形如 $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ 时，直接写出一个initial BFS，并证明这是一个BFS
- 否则，使用Two-Phase Method

L9 Simplex method: Simplex Tableau

1. Structure

$c^T - c_B^T A_B^{-1} A$	$-c_B^T A_B^{-1} b$
$A_B^{-1} A$	$A_B^{-1} b$

e.g.

B	-1	-2	0	0	0	0
3	1	0	1	0	0	100
4	0	2	0	1	0	200
5	1	1	0	0	1	150

①左上 (行向量)

Reduced cost vector:

$\bar{c}^T = c^T - c_B^T A_B^{-1} A$

②右上 (常数)

Negative of objective function value:

$-c_B^T A_B^{-1} b = -c_B^T x_B$

③左下 (矩阵)

Negative of Basic direction matrix

每一列代表对应方向的Basic direction

④右下 (列向量)

x_B:

$A_B^{-1} b = x_B$

2. Pivoting (In order of index)

Iteration of Simplex Method $\triangleq$ Pivoting in Simplex Tableau

① Choosing the incoming basis (Pick pivot col)

Look at the reduced cost vector:

$$\begin{cases} \text{if } \bar{c} \geq 0: \text{ Stop (Now the BFS is optimal)} \\ \text{Otherwise: Choose } j^{th} \text{ col with } \bar{c}_j < 0 \text{ under Bland's rule.} \end{cases}$$

② Choosing the outgoing basis (Pick pivot row)

Look at the basic direction matrix:

- If $\forall i, \bar{A}_{ij} \leq 0$ (即: $\forall i, d_i \geq 0$): The LP is unbounded (特别小心这一步!)
  j^{th} 那一列的 $\bar{A}_{ij}$ 都 ≤ 0
- Otherwise: We do Minimal Ratio Test (MRT), pick i^{th} row with minimal θ^*

Minimal Ratio Test (MRT): $\theta^* = \min\limits_{i: d_i < 0} \left(-\frac{x_i}{d_i}\right) \longrightarrow \theta^* = \min\limits_i \left\{ \frac{\bar{b}_i}{\bar{A}_{ij}} : \bar{A}_{ij} > 0 \right\}$, where

$$\begin{cases} \bar{b}: \text{右下部分的 } x_B \\ \bar{A}: \text{左下部分的 Basic direction matrix} \end{cases}$$

proof:

We just need to prove $\bar{A}_{ij} = -d_i$: $\bar{A}_{ij} = (A_B^{-1}A)_{ij} = (A_B^{-1}$ 的第 i 列$) \cdot A_j = -d_i$

③ Update the tableau

Gaussian elimination & Change the Basic index

a. Divide the pivot row by the pivot element (使得 pivot element = 1)

b. Make the elements in the pivot column to be 0 by adding multiples of pivot row

本质: 使得在这个 j^{th} direction 中, d_N 是一个 $[0, \cdots, 1, \cdots, 0]$ 向量

$$\begin{cases} \bar{A}_{1j}, \cdots, \bar{A}_{i-1,j} = 0 \\ \bar{A}_{ij} = 1 \quad \text{(只有 Pivot element = 1)} \\ \bar{A}_{i+1,j}, \cdots, \bar{A}_{mj} = 0 \end{cases} \longrightarrow \begin{cases} d_1, \cdots, d_{j-1} = 0 \\ d_j = 1 \\ d_{j+1}, \cdots, d_{n-m} = 0 \end{cases}$$

3. Process (Solve LP with Simplex Tableau)

① Transform a LP into a standard form LP

② Find an initial BFS
- Easy case
- Complex case ⟶ Two-phase method with Simplex Tableau (满足 $b \geq 0$)

③ Pivoting

L10 Complexity

1. Definition

① Textual definition

The # of operations (+/−/×/÷/Comparison …) in an algorithm

② O( )

An algorithm's ($g(n)$) complexity is $O(f(n))$

iff $\exists\ c_1, c_2$, s.t. $\forall\ n \geq c_2$, $|g(n)| \leq c_1 f(n)$

当n足够大时

2. Distinction between theoretical analysis & practical analysis

The criteria are different

- Complexity (Theoretically): Worst case
- Applicability (Practically): Under some specific conditions & Average case

Simplex method
- Worst case: Exponential
- Average case: Polynomial

Interior point method
- Worst case: Polynomial
- Average case: Polynomial

3. Complexity of a problem

- P : $\exists$ a polynomial-time algorithm to solve this problem
- LP : $\nexists$ a polynomial-time algorithm to solve this problem

L11-13 Duality Theory

1. Motivation

We try to convert a standard form LP into an equivalent expression:

- 形式上: Forms are symmetric
- Value上: 满足 Strong duality theorem: $c^Tx = b^Ty$
- Solution上: 满足 Complementarity condition theorem: $x_i(c_i - A_i^Ty) = 0, \ \forall i$

① Add penalty term

$$\begin{aligned}\min_{x}\ & c^Tx \\ \text{s.t. } & Ax=b \\ & x\geq 0\end{aligned} \iff \begin{aligned}\min_{x}\ & c^Tx + \max_{y\in R^n} y^T(b-Ax) \\ \text{s.t. } & x\geq 0\end{aligned}$$

把 equality constraints 放到 Objective function 里。

proof:

If $Ax^* \neq b$, then $\max_{y\in R^n} y^T(b-Ax^*) = \infty$ (因为 y is free), Infeasible $\Rightarrow$ Optimal value of minimization problem $= +\infty$

Then at this time, x^* is not optimal, so $Ax^* = b$

② Swap max & min

$$\min_{x\geq 0} c^Tx + \max_{y} y^T(b-Ax) \iff \max_{y} b^Ty + \min_{x\geq 0} x^T(c-A^Ty)$$

注意: 此时我们已经把 inequality constraint 放到 min 下方

表达式已被 Transposed.

proof:

③ Drop penalty term

$$\max_{y} b^Ty + \min_{x\geq 0} x^T(c-A^Ty) \iff \begin{aligned}\max_{y}\ & b^Ty \\ \text{s.t. } & A^Ty\leq c\end{aligned}$$

proof:

If $A^Ty^* \nleq c$, i.e. $\exists i$, s.t. $A^Ty_i^* > c_i$, Then $\min_{x\geq 0} x^T(c-A^Ty) = \infty$

Then at this time, y^* is not optimal, so $A^Ty^* \leq c$

2. Definition

Primal problem	Dual problem
$\min_x \; c^T x$	$\max_y \; b^T y$
s.t. $a_i^T x \geq b_i, \; i \in M_1,$	s.t. $y_i \geq 0, \; i \in M_1,$
$a_i^T x \leq b_i, \; i \in M_2,$	$y_i \leq 0, \; i \in M_2,$
$a_i^T x = b_i, \; i \in M_3,$	y_i free, $i \in M_3,$
$x_j \geq 0, \; j \in N_1,$	$A_j^T y \leq c_j, \; j \in N_1,$
$x_j \leq 0, \; j \in N_2,$	$A_j^T y \geq c_j, \; j \in N_2,$
x_j free, $j \in N_3$	$A_j^T y = c_j, \; j \in N_3$

x is primal variable — y is dual variable

3. Transformation (min→max) (max→min)

① Process

a. Objective function

(min→max):

- $\min \Longleftrightarrow \max$
- $x \Longleftrightarrow y$
- $c^T \Longleftrightarrow b^T$

(max→min):

- $\min \Longleftrightarrow \max$
- $x \Longleftrightarrow y$
- $c^T \Longleftrightarrow b^T$

b. Linear constraints

(min→max): Linear constraint $\longrightarrow$ Simple constraint

- $\geq \longrightarrow \geq$; $\leq \longrightarrow \leq$, $= \longrightarrow$ free
- $b_i \longrightarrow 0$

(max→min): Linear constraint $\longrightarrow$ Simple constraint

- $\geq \longrightarrow \leq$; $\leq \longrightarrow \geq$, $= \longrightarrow$ free
- $b_i \longrightarrow 0$

c. Simple constraint

(min→max): Simple constraint $\longrightarrow$ Linear constraint

- $\geq \longrightarrow \leq$; $\leq \longrightarrow \geq$; free $\longrightarrow =$
- $0 \longrightarrow c_j$
- A_j^T

(max→min): Simple constraint $\longrightarrow$ Linear constraint

- $\geq \longrightarrow \geq$; $\leq \longrightarrow \leq$; free $\longrightarrow =$
- $0 \longrightarrow c_j$
- A_j^T

② Examples

a. Example 1

$$\begin{aligned}\min_{x}\quad & x_1+2x_2\\ \text{s.t.}\quad & x_1+x_2\geq 5\\ & x_1-x_2\leq 3\\ & x_1\geq 0\\ & x_2\in\mathbb{R}\end{aligned}\quad\longleftrightarrow\quad\begin{aligned}\max_{y}\quad & 5y_1+3y_2\\ \text{s.t.}\quad & y_1\geq 0\\ & y_2\leq 0\\ & y_1+y_2\leq 1\\ & y_1-y_2=2\end{aligned}$$

b. Example 2

$$\begin{aligned}\min_{x}\quad & x_1+2x_2+3x_3\\ \text{s.t.}\quad & -x_1+3x_2=5\\ & 2x_1-x_2+3x_3\geq 6\\ & x_3\leq 4\\ & x_1\geq 0\\ & x_2\leq 0\\ & x_3\ \text{free}\end{aligned}\quad\longleftrightarrow\quad\begin{aligned}\max_{y}\quad & 5y_1+6y_2+4y_3\\ \text{s.t.}\quad & y_1\ \text{free}\\ & y_2\geq 0\\ & y_3\leq 0\\ & -y_1+2y_2\leq 1\\ & 3y_1-y_2\geq 2\\ & 3y_2+y_3=3\end{aligned}$$

c. Example 3 当自变量有多个时，拆开写dual constraints

$$\begin{aligned}\min_{w,b,t}\quad & \sum_{i=1}^{m} t_i\\ \text{s.t.}\quad & y_i(x_i^T w+b)+t_i\geq 1\quad \forall i=1,\cdots,m\\ & t_i\geq 0\quad \forall i=1,\cdots,m\end{aligned}$$

当表达式较复杂时，我们用矩阵表示：

$$\begin{aligned}\min_{w,b,t}\quad & \mathbf{1}^T t\\ \text{s.t.}\quad & (\operatorname{diag}(y)X^T\ \ y\ \ I)\begin{pmatrix}w\\ b\\ t\end{pmatrix}\geq \mathbf{1}\\ & w\ \text{free}\\ & b\ \text{free}\\ & t\geq 0\end{aligned}\quad\longleftrightarrow\quad\begin{aligned}\max_{u}\quad & \mathbf{1}^T u\\ \text{s.t.}\quad & u\geq 0\\ w\quad & X\operatorname{diag}(y)\,u=0\\ b\quad & y^T u=0\\ t\quad & u\leq 1\end{aligned}$$

把w.b.t对应部分拆开

自变量的个数从(m+n+1)变成n ⇒ Dual problem变简单了
($w\in\mathbb{R}^n$, $b\in\mathbb{R}$, $t\in\mathbb{R}^m$)

d. Example 4 (Tutorial 7)

Consider the problem

$$\begin{aligned} \min \quad & c^T x \\ s.t. \quad & Ax \geq b \end{aligned}$$

1. Derive the dual problem of
   - the original problem.
   - the standard form of the original problem, i.e. adding slack variables.
   - the original problem with x replaced by x^+ and x^-.
2. Show that these three dual problems are equal.

△ Add slack variables

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax + s = b \\ & x \text{ free} \\ & s \geq 0 \end{aligned} \quad \longleftrightarrow \quad \begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & y \text{ free} \\ & \underline{A^T y = c} \\ & y \leq 0 \end{aligned}$$

$\tilde{A} = [A, I] \longrightarrow \tilde{A}^T = \begin{bmatrix} A^T \\ I^T \end{bmatrix}$

The same.

△ Decompose x into x^+ & x^-

$$\begin{aligned} \min \quad & c^T (x^+ - x^-) \\ \text{s.t.} \quad & A(x^+ - x^-) + s = b \\ & x^+ \geq 0 \\ & x^- \geq 0 \\ & s \geq 0 \end{aligned} \quad \longleftrightarrow \quad \begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & y \text{ free} \\ & A^T y \leq c \\ & -A^T y \leq -c \longrightarrow A^T y \geq c \\ & y \leq 0 \end{aligned}$$

$\longrightarrow \underline{A^T y = c}$

4. Theorems

① Thm 1: Equivalence of dual

if problems $A_1 \triangleq A_2$, then their dual problems $B_1 \triangleq B_2$

② Thm 2: Dual²

The dual of dual is primal

③ Thm 3: Weak duality theorem

if x is feasible in primal problem and y is feasible in dual problem,

then $b^T y \le c^T x$

min $\downarrow c^T x$

max $\uparrow b^T y$

当且仅当Primal problem & Dual problem都达到optimal时才取等

proof.

When LP is in standard form,

$$b^T y = (Ax)^T y = \underbrace{x^T}_{\geq 0} \underbrace{A^T y}_{\le c} \le x^T c = c^T x$$

$x^T c = c^T x$ (因为$x^T c$是常数)

④ Thm 4: Strong duality theorem (Optimal value)

if primal problem has an optimal solution,

then its dual problem also gets an optimal solution and $c^T x = b^T y$ (反之也成立: x is primal feasible, y is dual feasible, $c^T x = d^T y$ ⇒ x & y are optimal)

即: min $\downarrow c^T x$, max $\uparrow b^T y$ 同时达到边界 $\rightarrow\leftarrow$

proof:

When LP is in standard form,

if x^* is the optimal solution of primal problem,

then we have $x^* = [x_B^T, x_N^T] = [A_B^{-1} b, 0^T]$ and $c^T - \underbrace{c_B^T A_B^{-1}}_{y^{*T}} A \geq 0 \Rightarrow A^T y^* \le c$

then we define $y^* = (A_B^{-1})^T c_B$,

then $A^T y^* \le c$, so y^* is feasible.

Primal		Dual	
min	$c^\top x$	max	$b^\top y$
s.t.	$Ax = b,\ x \geq 0$	s.t.	$A^\top y \leq c$

What's more,

$$b^T y^* = (y^*)^T b = c_B^T A_B^{-1} \cdot b = c_B^T x_B = c^T x$$

⑤ Thm 5: Attainability of LP

If an LP has a finite optimal value,

then this optimal value is attainable. Non-linear problem 的反例: $\frac{1}{x}$

⑥ Thm 6: Optimality & Feasibility & Boundedness

P | D	Finite Optimum	Unbounded	Infeasible	
Finite Optimum	✓			$\frac{c^Tx}{b^Ty}$ ↓↑
Unbounded			✓	$\frac{c^Tx = -\infty}{b^Ty}$
Infeasible		✓	✓	$\frac{c^Tx = +\infty}{b^Ty}$

(Infeasible 说明 $\min = +\infty$)

⑦ Thm 7: Complementarity conditions (Optimal solution)

a. Original form

Primal		Dual	
min	$c^\top x$	max	$b^\top y$
s.t.	$Ax = b,\ x \geq 0$	s.t.	$A^\top y \leq c$

x & y are optimal solution

iff $\quad x_i > 0 \longrightarrow A_i^T y = c_i$

$\quad A_i^T y < c_i \longrightarrow x_i = 0$

$\Longrightarrow x_i(c_i - A_i^T y) = 0, \ \forall i$

b. Slackness form ⇒ Complementarity slackness condition

Primal		Dual	
min	$c^\top x$	max	$b^\top y$
s.t.	$Ax = b,\ x \geq 0$	s.t.	$A^\top y + s = c,\ s \geq 0$

x & y are optimal solution

iff $x_i \cdot s_i = 0, \ \forall i$

c. General form

Primal			Dual		
min	$c^\top x$		max	$b^\top y$	
s.t.	$a_i^\top x \geq b_i,$	$i \in M_1,$	s.t.	$y_i \geq 0,$	$i \in M_1$
	$a_i^\top x \leq b_i,$	$i \in M_2,$		$y_i \leq 0,$	$i \in M_2$
	$a_i^\top x = b_i,$	$i \in M_3,$		y_i free,	$i \in M_3$
	$x_j \geq 0,$	$j \in N_1,$		$A_j^\top y \leq c_j,$	$j \in N_1$
	$x_j \leq 0,$	$j \in N_2,$		$A_j^\top y \geq c_j,$	$j \in N_2$
	x_j free,	$j \in N_3,$		$A_j^\top y = c_j,$	$j \in N_3$

x & y are optimal solution

iff $\begin{cases} x_j \cdot (A_j^T y - c_j) = 0, \forall j \\ y_i \cdot (a_i^T x - b_i) = 0, \forall i \end{cases}$ 必有一方取等

$\longrightarrow \forall i \in B$ and $x_i > 0 \Rightarrow y = (A_B^{-1})^T c_B$

proof:

Let x & y are optimal solutions of primal & dual

Then $c^T x = b^T y$

then $0 = c^T x - b^T y = \underbrace{c^T x - x^T A^T y}_{构造 x^T(c - A^T y)} + \underbrace{(Ax - b)^T y}_{构造 y^T(Ax-b)}$

注意这里是$(c-A^Ty)$而不是(A^Ty-c)

$= \sum_{j \in N_1} x_j(c_j - A_j^T y) + \sum_{j \in N_2} x_j(c_j - A_j^T y) + \sum_{j \in N_3} x_j(c_j - A_j^T y)$

$+ \sum_{i \in M_1} y_i(a_i^T x - b_i) + \sum_{i \in M_2} y_i(a_i^T x - b_i) + \sum_{i \in M_3} y_i(a_i^T x - b_i)$

≥ 0

$\Rightarrow \begin{cases} \forall j,\ x_j(A_j^T y - c_j) = 0 \\ \forall i,\ y_i(a_i^T x - b_i) = 0 \end{cases}$

5. Finding dual solution via Simplex tableau (第2种求dual sol的方法)

① Setup

Primal	Standard form of Primal	Compact form of SFP
$\min_x c^T x$	$\min_{x,s} c^T x + 0s$	$\min \tilde{c}^T y$
s.t. $Ax \le b$	s.t. $Ax + Is = b$	s.t. $\tilde{A} y = b$
$x \ge 0$	$x \ge 0$	$y \ge 0$
	$s \ge 0$	

$$y = \begin{bmatrix} x \\ s \end{bmatrix}, \quad \tilde{c} = \begin{bmatrix} c \\ 0 \end{bmatrix}, \quad \tilde{A} = \begin{bmatrix} A \\ I \end{bmatrix}$$

② Process

a. Using simplex tableau to solve primal problem

b. Finding the part of $\bar{c}$ that matches $\tilde{A} = I$, it is exactly the negative of our optimal dual solution.

$$-\bar{\tilde{c}} = y$$

proof:

Suppose after some iterations, our simplex tableau reaches an optimal solution, and our tableau is:

$c^T - c_B^T A_B^{-1} A$	$-c_B^T A_B^{-1} b$
$A_B^{-1} A$	$A_B^{-1} b$

At the very beginning, $\tilde{c} = 0$ matches $\tilde{A} = I$, $\tilde{c}$'s corresponding $\bar{\tilde{c}} = -c_B^T A_B^{-1} = -(A_B^{-1})^T c_B$

Since by complementarity condition, $x_j^* (A_j^T y - c_j) = 0$

Assume for $i \in B$, $x_j^* > 0$, then we have $A_B^T y = c_B \longrightarrow y = (A_B^T)^{-1} c_B = (A_B^{-1})^T c_B = -\bar{\tilde{c}}$

如果 x^* 是degenerate的?

→ 是不是要继续iterate?

→ 在一张simplex tableau里，我们可以同时读取 optimal primal sol / optimal dual sol

6. Application

① Production planning problem (Room)

Primal: Maximize profit

$$
\begin{array}{llll}
\text{maximize} & x_1 & +2x_2 & \\
\text{subject to} & x_1 & & \le 100 \\
 & & 2x_2 & \le 200 \\
 & x_1 & +x_2 & \le 150 \\
 & x_1, & x_2 & \ge 0.
\end{array}
$$

Dual: Minimize cost (向合金公司买原材料)

$$
\begin{array}{lllll}
\text{minimize} & 100p_1 & +200p_2 & +150p_3 & \\
\text{subject to} & p_1 & & +p_3 & \ge 1 \\
 & & 2p_2 & +p_3 & \ge 2 \\
 & p_1, & p_2, & p_3 & \ge 0
\end{array}
$$

$y^* = (p_1^*, p_2^*, p_3^*)$ is called Shadow price.

② Multi-Firm alliance problem

Primal: Maximize profit for an alliance

$$\max_x \; c^T x \longrightarrow v^S$$
$$\text{s.t. } Ax \le b_i$$
$$x \ge 0$$
$$\Downarrow$$

c^T、A不变

b变

$$\max_x \; c^T x \longrightarrow v^*$$
$$\text{s.t. } Ax \le \sum_{i \in S} b_i$$
$$x \ge 0$$

Dual: Allocate rationally

$$\min_y \; \Big(\sum_{i=1}^m b_i\Big)^T y$$
$$\text{s.t. } A^T y \ge c$$
$$y \ge 0$$

For the grand alliance: $\sum_{i=1}^m z_i = v^*$

For the alliances : $\sum_{i \in S} z_i \ge v^S \quad \forall S \subseteq \{1, \cdots, m\}$

在利润最大化时，每个sub-alliance 应比之前的大

③ Alternative systems (To prove infeasibility)

a. We want to consider the existence of solution (feasibility) of: $A^T y \le c$

b. We regard this LP as a dual feasibility problem, and write down the corresponding primal problem:

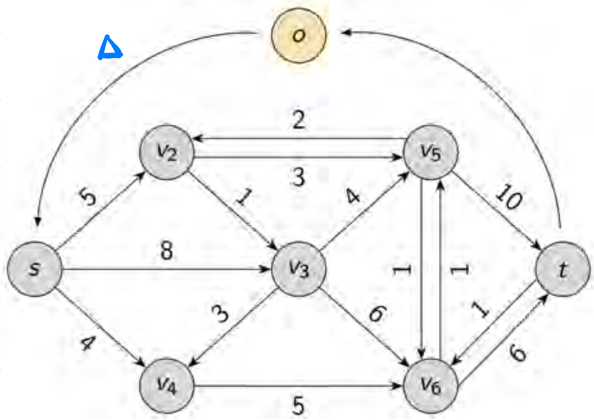
Primal	Dual
$\min \; c^T x$	$\max \; \underline{0}$ (Feasibility problem)
$\text{s.t. } Ax = 0$	$\text{s.t. } A^T y \le c$
$x \ge 0$	

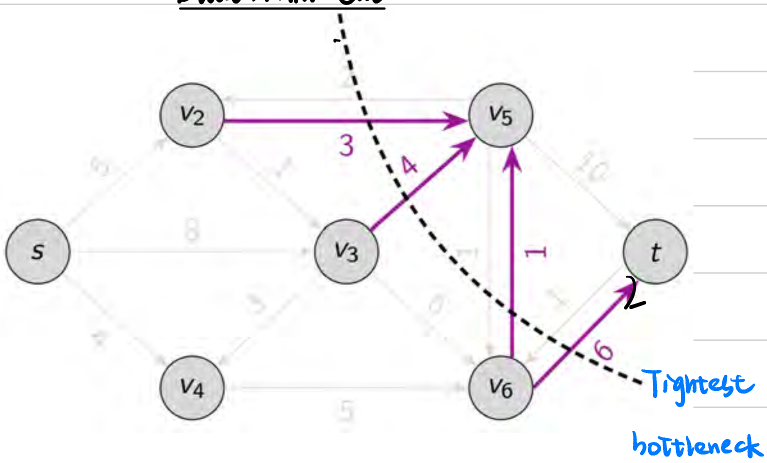
c. We conclude that $A^T y \le c$ has no solution when we can find a x s.t.
$$\begin{cases} Ax = 0 \\ x \ge 0 \\ c^T x < 0 \end{cases}$$

根据weak duality theory, 对 x & y are feasible, then $c^T x \ge b^T y = 0$,
"$c^T x < 0$"则说明此时 y is infeasible, i.e. "$A^T y \le c$" has no solution.

④ Maximum flow

<u>Primal: Max-Flow</u> <u>Dual: Min-Cut</u>





$$\max_{x,\Delta} \quad \Delta$$ 注意：要考虑到倒流的情况

$$\text{s.t.} \quad \sum_{j:(j,s)\in E} x_{js} + \Delta - \sum_{j:(s,j)\in E} x_{sj} = 0$$

$$\sum_{j:(j,t)\in E} x_{jt} - \Delta - \sum_{j:(t,j)\in E} x_{tj} = 0$$

$$\sum_{j:(j,i)\in E} x_{ji} - \sum_{j:(i,j)\in E} x_{ij} = 0 \quad , \forall i \neq s,t$$

$$x_{ij} \geq 0 \quad , \forall (i,j)\in E$$

$$x_{ij} \leq c_{ij} \quad , \forall (i,j)\in E$$

$$\min_{z,y} \quad \sum_{(i,j)\in E} c_{ij} z_{ij}$$

$$\text{s.t.} \quad z_{ij} \geq y_i - y_j$$

$$y_s - y_t = 1$$

$$z_{ij} \geq 0$$

y_i 节点的标签

z_{ij} 信号指标

We further assume that $y_i = 0 \mid 1$

$y_i = 0 \mid 1$

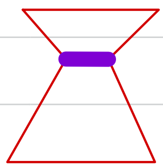
$z_{ij} = 0 \mid 1$

(x_{ij} 是 edge (i,j) 上的实际流量)

当这个network复杂时，x_{ij}数量过多，此时primal LP过于复杂.

只需找Minimal cut (Minimal weighted drop)，

从而找到 Tightest bottleneck (Boundary)



L14 Sensitivity Analysis

1. Motivation

Input $\left(\begin{array}{l} A \\ b \\ c \end{array}\right.$ changes $\longrightarrow$ Optimal sol & Optimal value change

2. Local sensitivity

① Assumption

a. The change of input is small

b. The optimal sol is unique

② Change (V是 primal & dual problem 的 Optimal value)

a. Change of b

$\nabla V(b) = y^*$

$\left\{\begin{array}{l} \text{Input:} \quad b \longrightarrow b + \Delta b_i \\ \text{Optimal val:} \ b^T y^* \longrightarrow b^T y^* + \underline{\Delta b_i y_i^*} \end{array}\right.$

当 Δb_i 发生在 inactive constraint 时，对应的 $y_i^* = 0$，故 $\left\{\begin{array}{l} \text{optimal sol} \\ \text{optimal value} \end{array}\right.$ 都不变.

proof:

If the change of input is small, then our optimal solution does not change

$V(b+\Delta b) - V(b) = (b+\Delta b)^T y^* - b^T y^* = \Delta b^T y^*$

b. Change of c

$\nabla V(c) = x^*$

$\left\{\begin{array}{l} \text{Input:} \quad c \longrightarrow c + \Delta c_i \\ \text{Optimal val:} \ c^T x \longrightarrow c^T x^* + \Delta c_i x_i^* \end{array}\right.$

3. Global sensitivity

① Motivation

We limit $\begin{cases}\Delta b \\ \Delta c\end{cases}$ is small in local sensitivity,

We want to find the range of $\begin{cases}\Delta b \\ \Delta c\end{cases}$ $\longrightarrow$ Find λ

(也就是说，我们在研究Local sensitivity 的assumption)

如果Δb | Δc变化过大
- Δb: $\tilde{x}^*$ is no longer feasible (Not BFS) $\longrightarrow$ Restart
- Δc: $\tilde{x}^*$ is no longer optimal $\longrightarrow$ Continue iteration

$\tilde{x}$ is still an optimal BFS

optimal $\uparrow$ $\bar{c} \geq 0$ BFS $\uparrow$ Feasible

② Criterion

Corresponding Optimal solution stays unchanged. $\longrightarrow$ $\boxed{(\Delta b \mid \Delta c) \subseteq \text{Range of change} \longrightarrow \text{Optimal basis stays unchanged}}$

- $\Delta b \longrightarrow y^*$不变，$x^*$变 (因为opt. value变了)
- $\Delta c \longrightarrow x^*$不变，$y^*$变

Optimal basis stays unchanged: primal & dual 的都不变

③ Optimal solutions

Recall in simplex tableau,

$$\begin{cases} x_B^* = A_B^{-1} b \\ y^* = (A_B^{-1})^T c_B \end{cases}$$

④ Changes (对于primal problem) (注意：要先写成standard form)

a. Change of b

If $b \longrightarrow \tilde{b} = b + \Delta b$,

then $\tilde{x}_B = \underline{A_B^{-1}} \tilde{b} = A_B^{-1} b + A_B^{-1} \Delta b = x_B^* + A_B^{-1} \Delta b$

(b变化时，A_B保持不变)

At this time, $V(\tilde{b}) = c_B^T \tilde{x}_B = c_B^T x_B^* + c_B^T A_B^{-1} \Delta b = V^* + \underline{c_B^T A_B^{-1}} \Delta b = V^* + (\underline{y^*})^T \Delta b$

正好在local sensitivity证明的部分.

To verify how small the Δb should be, we let $\Delta b = \lambda \underline{e_i}$ (e_i是一个向量，在i处为1)

思考一下，当$\tilde{x}_B$满足什么条件时，optimal basis保持不变？ $\longrightarrow$ Feasible

- A_B is invertible ✓
- $A_B \tilde{x}_B = b$ ✓
- $\tilde{x}_B \geq 0$ $\longrightarrow$ 不一定

So, we just need to let $\tilde{x}_B = x_B^* + \lambda A_B^{-1} e_i \geq 0$

$\longrightarrow$ Finally, we get then range of λ.

b. Change of c

If $c \longrightarrow \tilde{c} = c + \Delta c$,

then to verify how small the Δc should be, we let $\Delta c = \lambda e_j$

To ensure $\tilde{x}$ is still an optimal BFS.

then we need to have $\bar{\tilde{c}} = \tilde{c} - \tilde{c}_B^T A_B^{-1} A = \begin{bmatrix} \tilde{c}_B \\ \tilde{c}_N \end{bmatrix} - \begin{bmatrix} \tilde{c}_B^T A_B^{-1} A_B \\ \tilde{c}_B^T A_B^{-1} A_N \end{bmatrix} \geq 0$,

Since $\bar{\tilde{c}}_B^T = 0$, then we just need to have $\bar{\tilde{c}}_N^T \geq 0$

then we just need to have $\bar{\tilde{c}}_N^T = \tilde{c}_N^T - \tilde{c}_B^T A_B^{-1} A_N \geq 0$

因为 $\Delta c = \lambda e_j$，我们不知道要改的是哪一个 c_j，所以我们分类讨论：

△ Case 1: $j \in B$

Then $\bar{\tilde{c}}_N^T = \tilde{c}_N^T - \tilde{c}_B^T A_B^{-1} A_N$

$= c_N^T - (c_B^T + \lambda e_j^T) A_B^{-1} A_N$

$= c_N^T - c_B^T A_B^{-1} A_N - \lambda e_j^T A_B^{-1} A_N$

$= \underline{r_N^T} - \lambda e_j^T A_B^{-1} A_N \geq 0$

$(r_N^T = \bar{c}_N^T)$

△ Case 2: $j \in N$

Then $\bar{\tilde{c}}_N^T = \tilde{c}_N^T - \tilde{c}_B^T A_B^{-1} A_N$

$= (c_N^T + \lambda e_j^T) - \tilde{c}_B^T A_B^{-1} A_N$

$= c_N^T - c_B^T A_B^{-1} A_N + \lambda e_j^T$

$= r_N^T + \lambda e_j^T \geq 0$

c. Change of A

△ Case 1: $j \in B$

No simple way to deal with it.

△ Case 2: $j \in N$

$\longrightarrow$ Since it will not change $\tilde{x}_B$, then the feasibility stays unchanged

$\longrightarrow$ We need to further computer $\bar{c}$, check whether it's ≥ 0
- Yes $\longrightarrow$ BFS stays optimal
- No $\longrightarrow$ Continue iteration

d. Adding a variable

$\longrightarrow$ Feasibility: Still feasible (因为 basic index 的数目取决于m, 故 basis 不变, BFS 不变, 依然 feasible)

$\longrightarrow$ Optimality: Check $\bar{c_j}$
- ≥ 0 Still optimal
- < 0 Continue iteration

因为c_j不一定为0

<u>Example</u>

$$\max_{x} \quad x_1 + 2x_2$$
$$\text{s.t.} \quad x_1 \leq 100$$
$$2x_2 \leq 200$$
$$x_1 + x_2 \leq 150$$
$$x_1, x_2 \geq 0$$

$\xrightarrow{\text{Add } x_3}$

$$\max_{x} \quad x_1 + 2x_2 + 2x_3$$
$$\text{s.t.} \quad x_1 + x_3 \leq 100$$
$$2x_2 + x_3 \leq 200$$
$$x_1 + x_2 + x_3 \leq 150$$
$$x_1, x_2, x_3 \geq 0$$

B						
B	0	0	0	1/2	1	250
1	1	0	0	-1/2	1	50
3	0	0	1	1/2	-1	50
2	0	1	0	1/2	0	100

$\longrightarrow$ $\bar{c}_6 = c_6 - c_B^T A_B^{-1} A_6$ $\longrightarrow$ Continue iteration

$$= -2 - [1,2,0] \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= -2 - [1,2,0] \times \left(-\frac{1}{2}\right) \begin{bmatrix} 0 & 1 & -2 \\ 0 & -1 & 0 \\ -2 & -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= -2 + \frac{3}{2} = -\frac{1}{2}$$

- 方法一：自己算
- 方法二：<u>从Tableau读</u>

(注意此时是B, 所以c_B要按顺序读)

e. Adding a constraint

$\longrightarrow$ Feasibility
- Still feasible: Still a BFS
- Not feasible:
  - ► If not, then the best way to deal with it is to interpret it as adding a dual variable. Then use the simplex tableau for the dual problem to continue calculations.

$\longrightarrow$ Optimality: Still optimal

L15-16 Interior point method

1. Motivation

Simplex method (Feasibility of primal solution / Complementarity conditions → Dual solution

Dual-Simplex method (Feasibility of dual solution / Complementarity conditions → Primal solution

Rotated variant

Interior point method (Primal-dual type) (Feasibility of primal solution / Feasibility of dual solution → Complementarity conditions

→ We want to find x, y and s, s.t.

$Ax = b, \ x \geq 0$ — Primal feasibility

$A^T y + s = c, \ s \geq 0$ — Dual feasibility

$x_i \cdot s_i = 0, \ \forall i$ — Complementarity condition

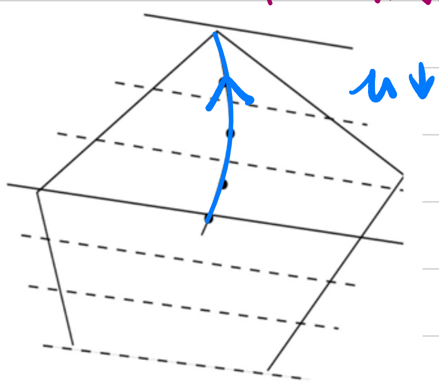
这是一个 Non-linear constraint

↓ 转化

$x_i \cdot s_i \leq u, \ u > 0, \ \forall i$, u is called complementarity gap

idea: 找到一个 solution 与对应的 u

→ 减小 u, 直到找到另外一个 solution



(在到达 optimal 之前, $x > 0$, $s > 0$

通过 interior point method, Optimal solution 不一定是 BFS.

Difference:

- Simplex method: 提供 Low-rank solution ⟶ 简化选择
- Interior point method: 提供 High-rank solution ⟶ 均匀选择

Maximal number of non-zeros

417 Mathematical background

1. Matrix

① Orthogonality

a. Orthogonal

$$\begin{cases} \text{Vectors are orthogonal: } \langle x,y\rangle = 0 \\ \text{Matrix is orthogonal: Its cols are orthogonal to each other} \end{cases}$$

b. Orthonormal

A is orthonormal iff $A^TA = I$

② Trace

$tr(A) = a_{11} + \cdots + a_{nn}$

③ Definiteness

a. Definition

$$A \text{ is } \begin{cases} \text{positive semidefinite} \\ \text{positive definite} \\ \text{negative semidefinite} \\ \text{negative definite} \\ \text{indefinite} \end{cases}, \text{ iff } \begin{cases} \forall x \in R^n,\ x^TAx \geq 0 \\ \forall x \in R^n \setminus 0,\ x^TAx > 0 \\ \forall x \in R^n,\ x^TAx \leq 0 \\ \forall x \in R^n \setminus 0,\ x^TAx < 0 \\ \text{A is neither positive semidefinite nor negative semidefinite} \end{cases}$$

b. Method of judging

△ Eigenvalue method

$$\begin{cases} \prod_{i=1}^{n} \lambda_i = \det(A) \\ \sum_{i=1}^{n} \lambda_i = tr(A) \end{cases}$$

$$\begin{cases} \text{Positive definite} &: \forall i,\ \lambda_i > 0 \\ \text{Positive semidefinite} &: \forall i,\ \lambda_i \geq 0 \\ \text{Negative definite} &: \forall i,\ \lambda_i < 0 \\ \text{Negative semidefinite} &: \forall i,\ \lambda_i \leq 0 \end{cases}$$

△ Principal minor method

Let $\begin{cases} D_k \text{ denote the leading principal minor of order } k \\ \Delta_k \text{ denote the principal minor of order } k \end{cases}$

$Q(x) > 0 \longleftrightarrow D_k > 0$, for $k = 1, 2, 3, \cdots, n$

$Q(x) \geq 0 \longleftrightarrow \Delta_k \geq 0$, for $k = 1, 2, 3, \cdots, n$

$Q(x) < 0 \longleftrightarrow (-1)^k D_k > 0$, for $k = 1, 2, 3, \cdots, n$

$Q(x) \leq 0 \longleftrightarrow (-1)^k \Delta_k \geq 0$, for $k = 1, 2, 3, \cdots, n$

2. Sequence

① Definition

$\{x^k\} = x^1, \cdots, x^k$ 代表第k项

② Convergence / Limit

$\lim_{k\to\infty} x^k = x^*$ 代表极限.

3. Multivariate calculus

① Continuous

各个方向的极限=极限=函数值 $\longrightarrow$ 连续

be continuous $\triangleq$ there is no "jump"

② Differentiability

F is differentiable at x iff:

$\exists$ a matrix DF s.t.

$$\lim_{h\to 0} \frac{\|F(x+h)-F(x)-DF(x)\cdot h\|}{\|h\|} = 0$$

DF is the derivative of F ($\mathbb{R}^n \to \mathbb{R}^{m\times n}$), and it's called Jacobian matrix. ($\mathbb{R}^{m\times n}$)

③ Gradient & Jacobian matrix

Gradient $\mathbb{R}^n \to \mathbb{R}$: $\nabla f(x) = Df(x)$

Jacobian matrix $\mathbb{R}^n \to \mathbb{R}^m$: $DF(x) = \begin{pmatrix} \nabla f_1(x) \\ \vdots \\ \nabla f_m(x) \end{pmatrix}_{m\times n}$

④ Hessian matrix 二次偏导矩阵

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial f}{\partial x_1 \partial x_1} & & \\ & \ddots & \\ & & \frac{\partial f}{\partial x_n \partial x_n} \end{pmatrix}$$

L18 Optimality conditions for unconstrained problems

1. Local optimality

① Necessary condition

$$\begin{cases} \text{FOC: } \nabla f(x^*) = 0 \\ \text{SOC: } \nabla^2 f(x^*) \text{ is positive semidefinite} \end{cases}$$

② Sufficient condition

$$\begin{cases} \text{FOC: } \nabla f(x^*) = 0 \\ \text{SOC: } \nabla^2 f(x^*) \text{ is positive definite} \end{cases}$$

FOC $\longrightarrow$ Stationary point
$$\begin{pmatrix} \text{SOC: PD} & \text{Local minimizer} \\ \text{SOC: ND} & \text{Local maximizer} \\ \text{SOC: Indefinite} & \text{Saddle point} \end{pmatrix}$$
当PSD、NSD时，要进一步判断是其中哪一类

③ Examples

a. Example 1

We consider the two-dimensional optimization problem

$$\min_{x \in \mathbb{R}^2} \; f(x) = x_1^2 x_2 + x_1 x_2^3 - 5x_1 x_2$$

Find all local minimizer, local maximizer, and saddle points of f!

FOC: $\begin{cases} 2x_1x_2 + x_2^3 - 5x_2 = 0 \\ x_1^2 + 3x_1x_2^2 - 5x_1 = 0 \end{cases}$

To solve FOC, we discuss the following 4 case:

Case 1: $x_1^* = x_2^* = 0 \longrightarrow$ Stationary pt $= (0,0)$

Case 2: $x_1^* = 0,\ x_2^* \neq 0 \longrightarrow$ Stationary pt $= (0,\sqrt{5}), (0,-\sqrt{5})$

Case 3: $x_1^* \neq 0,\ x_2^* = 0 \longrightarrow$ Stationary pt $= (5,0)$

Case 4: $x_1^* \neq 0,\ x_2^* \neq 0 \longrightarrow \begin{cases} 2x_1 + x_2^2 - 5 = 0 \\ x_1 + 3x_2^2 - 5 = 0 \end{cases} \longrightarrow$ Stationary pt $= (2,1), (2,-1)$

SOC

$\nabla^2 f(x^*)$	λ_1, λ_2	Definiteness
$\begin{pmatrix} 0 & -5 \\ -5 & 0 \end{pmatrix}$	$5, -5$	ID
$\begin{pmatrix} 0 & 5 \\ 5 & 0 \end{pmatrix}$	$5, -5$	ID
$\begin{pmatrix} \sqrt{5} & 10 \\ 10 & 0 \end{pmatrix}$	$5\sqrt{5}, -4\sqrt{5}$	ID
$\begin{pmatrix} -\sqrt{5} & 10 \\ 10 & 0 \end{pmatrix}$	$4\sqrt{5}, -5\sqrt{5}$	ID
$\begin{pmatrix} 1 & 2 \\ 2 & 12 \end{pmatrix}$	$12.4, 0.7$	PD $\longrightarrow$ Local min
$\begin{pmatrix} -1 & 2 \\ 2 & -12 \end{pmatrix}$	$-0.7, -12.4$	ND $\longrightarrow$ Local max

b. Example 2 : OLS

$\min \|X\beta - y\|^2 = \beta^T X^T X \beta + y^T y - 2\beta^T X^T y$

$$\begin{cases} \nabla f(\beta) = 2X^T X\beta - 2X^T y \\ \nabla^2 f(\beta) = 2X^T X \end{cases}$$

对 $f(x) = \frac{1}{2}x^T A x + b^T x + c$, then $\begin{cases} \nabla f(x) = Ax + b \\ \nabla^2 f(x) = A \end{cases}$

2. Global optimality

Local optimality → Global optimality

Check ∞

(因为对于 Unconstrained problem, 没有 Boundary point, 故只需 check ∞)

L19-21 Optimality conditions for constrained problems

1. General form of NLP

① Form

$$\min_{x} \quad f(x)$$

$$\text{s.t.} \quad g(x) \le 0$$

$$h(x) = 0$$

② Concepts

For inequality constraint $g(x)$:

$$\begin{cases} g(x) < 0: & \text{inactive constraint} \\ g(x) = 0: & \text{active constraint} \end{cases}$$

2. Feasible direction

① Definition

Given $x \in \Omega$, we call d a feasible direction at x

if $\exists \bar{t} > 0$, s.t. $x + td \in \Omega$ for all $t \in [0, \bar{t}]$

即：一个可行feasible point $\xrightarrow{\text{Feasible direction}}$ Another feasible point

② Example

a. Equality linear constraint

$\Omega = \{x \mid Ax = b\} \longrightarrow D = \{d \mid Ad = 0\}$

proof:

$A(x + td) = b \rightarrow tAd = 0 \longrightarrow Ad = 0$

b. Inequality linear constraint

$\Omega = \{x \mid Ax \le b\} \longrightarrow D = \{d \mid$

proof:

$A(x + td) \le b$

Since $Ax \le b$,

then $\begin{cases} \text{for } i: a_i^T x = b: \; t a_i^T d \le 0 \longrightarrow a_i^T d \le 0 \\ \text{for } i: a_i^T x < b: \; \text{there always exists } t \to 0, \text{ s.t. } a_i^T(x + td) < b \longrightarrow d \text{ free} \end{cases}$

当feasible point是开区间的一个interior point时，它在全部域内移动后，仍是开区间的一个interior point

In all, $D = \{d \mid a_i^T d \le 0 \text{ if } a_i^T x = b_i\}$

3. Local optimality

① Necessary condition

a. General constraints

FOC: $\forall d, \nabla f(x^*)d \geq 0$ For unconstrainted problem, since all directions are feasible, thus we must have $\nabla f(x^*)=0$

SOC: 略 ($\nabla^2 f(x)$ of Largragian)

b. Linear constraints

△ Linear inequality constraint: $Ax \leq b$

FOC: $\exists\, y \geq 0$, s.t. $\begin{cases} \nabla f(x^*) + A^T y = 0 \\ y_i(a_i^T x^* - b_i) = 0, \ \forall i \end{cases}$

proof:

Let x^* be a local solution of our problem, then: $\forall d, \nabla f(x^*)^T d \geq 0$

So this solution means: $\nabla f(x^*) d \geq 0$, for $d \in D = \{d \mid a_i^T d \leq 0 \text{ if } a_i^T x^* = b_i\}$

Then, we construct an optimization problem:

$$\min_d \ \nabla f(x^*) d$$
$$\text{s.t. } a_i^T d \leq 0, \ \forall i \in A(x^*) \triangleq \text{if } a_i^T x^* = 0$$

For x^*, it's feasible and $\nabla f(x^*) d$ is lower bounded by 0

We denote $C = \begin{bmatrix} c_1^T \\ \vdots \\ c_m^T \end{bmatrix}$ with $c_i^T = \begin{cases} a_i^T & \text{if } i \in A(x^*) \\ 0 & \text{if } i \notin A(x^*) \end{cases}$.

then the optimization problem can be reduced to:

$$\min_d \ \nabla f(x^*) d$$
$$\text{s.t. } Cd \leq 0$$

Next, we consider the dual of this optimization problem:

$$\max_y \ 0 \quad \text{s.t. } y \leq 0, \ C^T y = \nabla f(x^*) \quad \xrightarrow{\text{Let } y = -y} \quad \max_y \ 0 \quad \text{s.t. } y \geq 0, \ \nabla f(x^*) + C^T y = 0$$

By optimality & feasibility in duality: Since x^* is feasible, then y also should be feasible.

i.e. $y \geq 0$ and $\nabla f(x^*) + C^T y = \nabla f(x^*) + \sum_{i=1}^m c_i y_i = \nabla f(x^*) + \sum_{i \in A(x^*)} a_i y_i = 0$

For y_i with $i \in I(x^*)$, we can just set $y_i = 0$

Together, y now should satisfies the conditions: $\begin{cases} y \geq 0 \\ \nabla f(x^*) + A^T y = 0 \\ y_i(a_i^T x^* - b_i) = 0, \ \forall i \end{cases}$

△2 Linear equality constraint: $Ax = b$

FOC: $\exists y$, s.t. $\nabla f(x^*) = A^T y$

proof:

FOC for general constraints: $\forall d, \nabla f(x^*)^T d \geq 0$

$\longrightarrow \forall d \in D = \{d \mid Ad = 0\}, \nabla f(x^*)^T d \geq 0$

$$\longrightarrow \min_d \nabla f(x^*) d \quad \text{s.t. } Ad = 0$$

$$\longrightarrow \max_y 0 \quad \text{s.t. } y \text{ free}, \ A^T y = \nabla f(x^*)$$

$\longrightarrow$ FOC: $\exists y: A^T y = \nabla f(x^*)$

4. KKT conditions

① Motivation

We have derived the FOC for
- General constrained problem with feasible direction,
- Linear constrained problem

we want to further derive a more applicable FOC method for General constrained problem.

② Lagrangian

$$L(x, \lambda, u) = \boxed{f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=1}^{p} u_j h_j(x)}$$ 小心Constraint的数目!

- $\lambda \longrightarrow g(x)$ inequality constraint
- $u \longrightarrow h(x)$ equality constraint

③ Definition

a. Primal feasibility: $g(x) \leq 0$ & $h(x) = 0$

b. Main condition: $\nabla L(x, \lambda, u) = \nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x) + \sum_{j=1}^{p} u_j \nabla h_j(x) = 0$

c. Dual feasibility: $\lambda_i \geq 0$, u_i free

d. Complementarity condition: $\lambda_i g_i(x) = 0$, $\forall i = 1, \cdots, m$

5. Convexity

① Convex function

a. Definition

使得 $\lambda x_1+(1-\lambda)x_2$ 有意义

$\forall x_1, x_2 \in \Omega$ & $0 \le \lambda \le 1$, $f(\lambda x_1+(1-\lambda)x_2) \le \lambda f(x_1)+(1-\lambda)f(x_2)$

b. Judgement method

f is convex iff $\nabla^2 f(x)$ is PSD

c. Common example

△ Linear function

△ $\|x\|^2$ ($= x^T x$)

△ Sum of convex function: If $a_i \ge 0$ & f_i convex, then $\sum a_i f_i$ convex

△ Composition with linear function: If $f(x)$ convex, then $f(Ax+b)$ convex

proof:

Take $x, y \in \Omega$, Let $g(x) = f(Ax+b)$,

then $g(\lambda x+(1-\lambda)y) = f(A(\lambda x+(1-\lambda)y)+b) = f(\lambda Ax+Ay-\lambda Ay+b) = f(\lambda(Ax+b)+(1-\lambda)(Ay+b))$

$\le \lambda f(Ax+b)+(1-\lambda)f(Ay+b)$

$= \lambda g(x)+(1-\lambda)g(y) \longrightarrow f(Ax+b)$ is convex

△ Max of convex function

d. Transformation

Non-convex / Non-concave function $\longrightarrow$ Convex / Concave function $\longrightarrow$ Convex optimization

- 单调变换 (e.g. $xyz \longrightarrow \log x + \log y + \log z$)
- 换元 (e.g. $x^3 \le 1 \longrightarrow x \le 1$)

② Convex constraint

a. Judgement method

If f convex, then $L = \{x : f(x) \le c\}$ is a convex set
concave: $f(x) \ge c$

这只是充分条件,非必要条件,
也就是说,不满足此形式的constraint 也有可能为convex constraint

(Linear constraints are always convex constraints)

所以 $\ge$ | $\le$ 都可以

6. Summary of procedure

① Step 1: Convexity & EVT

② Step 2: KKT conditions

③ Step 3: Boundary points

④ Step 4: CQ violated case (下面这两个向量都不能dependent)

⑤ Step 5: Conclude

L22 Algorithms for unconstrained optimization

1. Bisection method

① Motivation

We want to find an alternative way to solve FOC (解方程太困难)

② Fundamental

Intermediate value theorem

③ Procedure

a. Step 1: Find x_l and x_r s.t. $g(x_l)<0$ and $g(x_r)>0$

b. Step 2: Find $x_m = \frac{x_l+x_R}{2}$, and compute $g(x_m)$ $\begin{cases} =0, & \text{output } x_m \\ >0, & \text{let } x_r = x_m \\ <0, & \text{let } x_l = x_m \end{cases}$

c. Step 3: $\begin{cases} \text{If } x_r - x_l < \varepsilon, \text{ stop and output } \frac{x_l+x_r}{2} \\ \text{Otherwise, continue iteration} \end{cases}$

2. Golden section method

① Motivation

Bisection method is a good way to deal with FOC, however, what if f' is not available?

② Fundamental: Unimodality

a. Definition

f is unimodal if it only has one stationary point

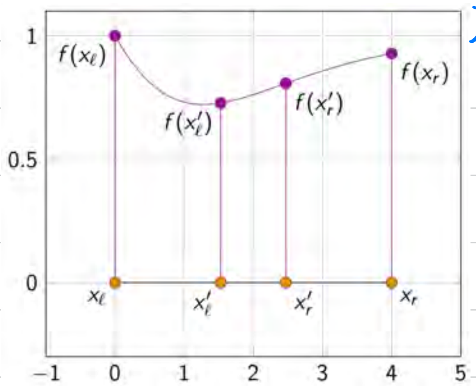
b. Property

$\left.\begin{array}{l} f \text{ is unimodal} \\ x^* \text{ is a local minimizer} \end{array}\right\} \longrightarrow x^*$ is a global minimizer

③ Procedure

a. Step 1: Focus on one interval $[x_l, x_r]$ that f in that interval is unimodal

b. Step 2: Set $\begin{cases} x_l' = (1-\phi)x_l + \phi x_r \\ x_r' = (1-\phi)x_r + \phi x_l \end{cases}$, $0 < \phi < 0.5$.



若此时 set $x_r' = x_r$，为了使得下一个iteration的x_r'与这个iteration的x_l'重合，我们令$\phi = \frac{3-\sqrt{5}}{2}$，即 $1-\phi = \frac{\sqrt{5}-1}{2} = 0.618$

黄金分割数

compute $f(x_l')$ & $f(x_r')$ $\begin{cases} \text{if } f(x_l') < f(x_r'), \text{ then the minimizer lies in } [x_l, x_r'], \text{ so let } x_r = x_r' \\ \text{if } f(x_l') > f(x_r'), \text{ then then minimizer lies in } [x_l', x_r], \text{ so let } x_l = x_l' \end{cases}$

c. Step 3: $\begin{cases} \text{If } x_r - x_l < \varepsilon, \text{ stop and output } \frac{x_l + x_r}{2} \\ \text{Otherwise, continue iteration} \end{cases}$

L2 Gradient descent method

1. Motivation

Bisection method & Golden section method can be just applied in one-dimensional problems

We need to find some methods to deal with high-dimensional problems (Unconstrained problem)

2. Procedure

① Find an initial point

② Judge whether stop
- If $\|\nabla f(x^k)\| \leq \varepsilon$, then stop and output x^k
- otherwise, continue to do iteration

③ Along the descent direction $[-\nabla f(x^k)]$, find the stepsize by Exact line search / Backtracking

④ Set $x^{k+1} = x^k + \alpha_k d^k = x^k - \alpha_k \nabla f(x^k)$

3. Descent direction

① Definition

A vector $d \in \mathbb{R}^n$ is a descent direction of f at x if $\nabla f(x)^T d < 0$ (strictly)

② Property

$f(x^k + \alpha d^k) < f(x^k)$, $\forall \alpha \in [0, \varepsilon]$, $\varepsilon > 0$

4. Step size

① Exact line search

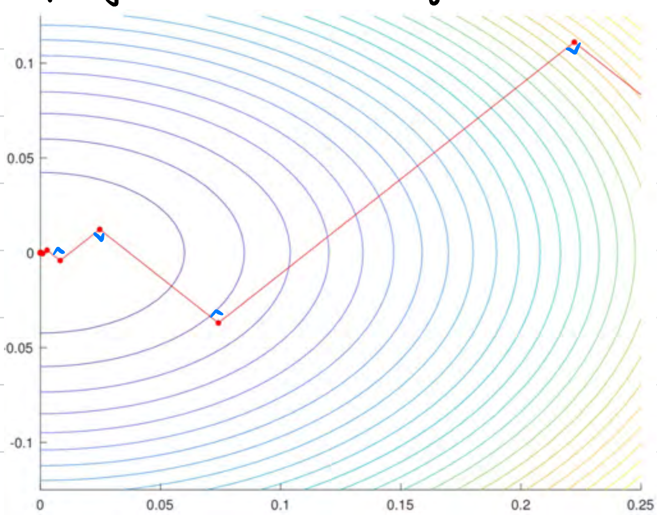
a. Definition

$$\alpha_k = \underset{\alpha \geq 0}{\operatorname{argmin}}\, f(x^k + \alpha d^k)$$

此时，求α_k转化成关于$f(x^k+\alpha d^k)$的一维优化问题

b. Graphic implied meaning

Zig-zag (Directions are orthogonal with each other)



proof:

$$(d^{k+1})^T d^k = \nabla f(x^{k+1}) \cdot \nabla f(x^k)$$
$$= \nabla f(x^k + \alpha_k d^k) d^k$$

Since α_k is the minimizer of $\phi(\alpha) = f(x^k + \alpha_k d^k)$, then $\phi'(\alpha_k) = \nabla f(x^k + \alpha_k d^k) \cdot d^k = 0$

$\longrightarrow$ Perpendicular!

② Backtracking / Armijo line search

a. Definition

△ Original definition

Choose $\sigma, \gamma \in (0,1)$,

and choose α_k as the largest element in $\{1, \sigma, \sigma^2, \sigma^3, \cdots\}$ s.t. $\underbrace{f(x^k+\alpha_k d^k) - f(x^k) \le \gamma \alpha_k \cdot \nabla f(x^k)^T d^k}_{\text{Armijo condition}}$

α_k can be determined after finitely many steps if d^k is a descent direction (一定能找到stepsize)

proof:

By 1st order Taylor expansion,

$f(x^k+\alpha d^k) \approx f(x^k) + \alpha \underbrace{\nabla f(x^k)^T d^k}_{\nabla f(x^k)^T d^k = -\|\nabla f(x^k)\|^2 < 0} < f(x^k) + \gamma \alpha \nabla f(x^k)^T d^k$

△ Functional-form definition

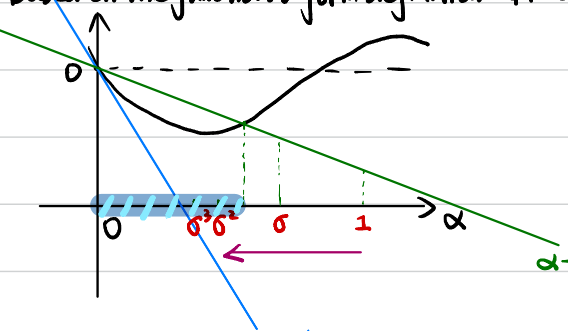
We can rewrite Armijo condition into functional form of α:

Define $\phi_k(\alpha) = f(x^k+\alpha d^k) - f(x^k)$,

then Armijo condition $\triangleq$ $\phi_k(\alpha) \le \gamma \alpha \cdot \phi_k'(0)$

b. Visualization

Based on the functional-form definition: $\phi_k(\alpha) \le \gamma \alpha \cdot \phi_k'(0)$



$\alpha \to \alpha \cdot \gamma \cdot \phi_k'(0)$ (因为$\gamma \in (0,1)$，所以曲线更平了)

$\alpha \to \alpha \cdot \phi_k'(0)$

③ Diminishing step size

a. Definition

Choose $\{\alpha_k\}$ with $\alpha_k \to 0$ and $\sum_{k=0}^{\infty} \alpha_k = \infty$

b. Example

Choose $\alpha_k = \frac{1}{k+1}$

④ Constant stepsize

If $c \in C_L^{1,1}$, then $\bar{\alpha} \in (0, \frac{2}{L})$ is okay.

5. Global convergence

① Definition

No matter which initial point is chosen, we can always find stationary points with gradient descent method.

为什么不是local minimizers?：因为local minimizers要求所有方向梯度下降

Gradient descent method $\longrightarrow$ Accumulation points $\longrightarrow$ Stationary points

② Accumulation point / Cluster point

a. Definition

A point x is an accumulation point of $\{x^k\}_k$ iff

$\forall \varepsilon > 0$, there are infinitely many numbers k with $x^k \in B_\varepsilon(x)$

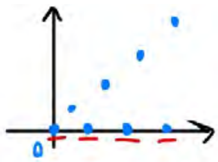
Accumulation point是弱化版的Limit point：All $\longrightarrow$ Infinitely many

Limit point是Accumulation point的子集

b. Example

- The sequence

$$a_k := \begin{cases} k & k \text{ is odd}, \\ 0 & k \text{ is even}, \end{cases}$$



is not bounded. However, it has the accumulation point $a = 0$.

6. Local convergence

① Definition

In each step, there is a rate of convergence

② A concept: Spectral norm

a. Definition

Spectral norm of A is the maximum norm of A's eigenvalue

$\boxed{\|A\| = \sqrt{\lambda_{max}(A^TA)} = \max_{1\le i\le n} |\lambda_i|} = \max_{\|d\|=1} \|Ad\|$

proof:

Let $A = P\Lambda P^T$, where $\begin{cases} P \text{ is an orthogonal matrix} \\ \Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_n), \text{ with } \lambda_i \text{ being the } i^{th} \text{ eigenvalue of } A. \end{cases}$

Then $\|A\| = \sqrt{\lambda_{max}(A^TA)} = \sqrt{\lambda_{max}(P\Lambda P^T P\Lambda P^T)} = \sqrt{\lambda_{max}(P\,\mathrm{diag}(\lambda_1^2, \cdots, \lambda_n^2)P^T)} = \sqrt{\max_{1\le i\le n} \lambda_i^2} = \max|\lambda_i|$

③ Assumption: Lipschitz continuity

a. Definition

$\nabla f(x)$ is Lipschitz continuous iff:

$\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|, \forall x, y \in R^n$

$L \geqslant 0$ is called Lipschitz constant

The class of functions with Lipschitz gradient with constant L is denoted by $C_L^{1,1}$

不等号左侧一阶导 不等号右侧一次模

b. Theorem

$\nabla f(x)$ is Lipschitz continuous iff:

$\forall x, \|\nabla^2 f(x)\| \le L$

Boundness of the Hessian

c. Example

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

then $\|\nabla f(x) - \nabla f(y)\| = \|A(x-y)\| \le \|A\|\|x-y\| \longrightarrow f \in C_L^{1,1}$ with $L = \|A\|$

$C_L^{1,1}$

$\|\nabla^2 f(x)\| \le L, \forall x$

④ Linear convergence

a. Definition

We say $\{x^k\}_k$ converges linearly with rate $\eta \in (0,1)$ to x^* if:

$\|x^{k+1} - x^*\| \le \eta \cdot \|x^k - x^*\|$

i.e. $\lim\limits_{k\to\infty} \dfrac{\|x^{k+1}-x^*\|}{\|x^k - x^*\|} \le \eta,\ \eta \in (0,1)$

b. Theorem

$\{x^k\}_k$ converges linearly if:

$\{x^k\}_k$ is generated by the gradient method & $f(x)$ is strongly convex,

$f \in C_L^{1,1}$ & Strongly convex

区分 $\begin{cases} \text{Strictly convex: } \lambda_i > 0,\ \forall i \\ \text{Strongly convex: } \mu\|d\|^2 \le d^T \nabla^2 f(x) d,\ \forall d, \forall x \ \& \ \mu > 0 \end{cases}$

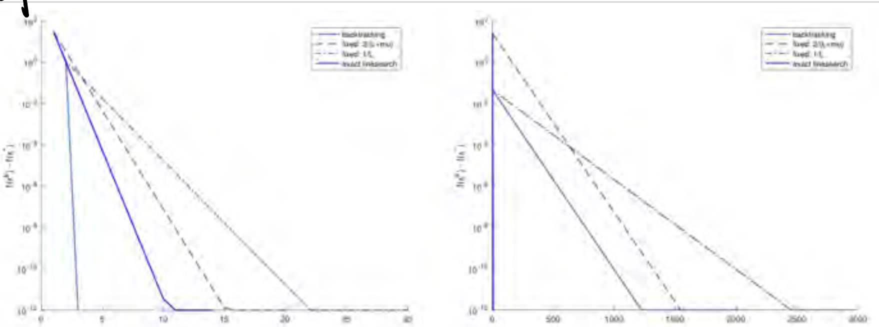
e.g. e^x $\begin{cases} \lambda_i > 0 \longrightarrow \text{Strictly convex} \\ d^T e^x d \ge 0 \longrightarrow \mu = 0 \longrightarrow \text{Not strongly convex} \end{cases}$

i.e. $\exists \mu > 0$, s.t. $\mu\|d\|^2 \le d^T \nabla^2 f(x) d \le L\|d\|^2,\ \forall d, \forall x$ $\begin{cases} \mu \le \lambda_{min}(\nabla^2 f(x)) \\ L \ge \lambda_{max}(\nabla^2 f(x)) \end{cases}$

What's more, the corresponding rate is $\boxed{\eta = 1 - \frac{\mu}{L}}$

为了使η尽量小(下降速度快),我们使 $\begin{cases} \mu\text{尽可能大} \\ L\text{尽可能小} \end{cases}$

e.g.



▸ Comparison of the gradient method with backtracking ($\gamma = \sigma = \frac{1}{2}$) and constant step size for

$$\min_x \frac{1}{2} x^\top A x, \quad A_1 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 2 & 0 \\ 0 & \frac{1}{100} \end{pmatrix}.$$

It holds $L = 2$, $\mu_1 = 1$, $\mu_2 = \frac{1}{100}$. The predicted rates are $\frac{1}{2}$ and $\frac{199}{200} \approx 0.99$.

c. Example

Let $\eta \in (0,1)$, $x^k = \eta^k$, $x^* = 0$, then: $\dfrac{|x^{k+1} - x^*|}{|x^k - x^*|} = \dfrac{\eta^{k+1}}{\eta^k} = \eta$, $\forall k \ge 0$

L24-25 Newton's method

1. Motivation

Compared with gradient descent method:

① Pro

Converges quadratically (Faster)

② Cons

Require second-order information

More sensitive to the initial point   1st order Taylor Expansion 要求

2. Newton's method in R

① Motivation

We want to find stationary points by FOC: $f'(x)=0$, but it's not easy to solve.

Alternatively, we apply first-order Taylor expansion for $g(x)=f'(x)$, then solve x approximately.

② Derivation

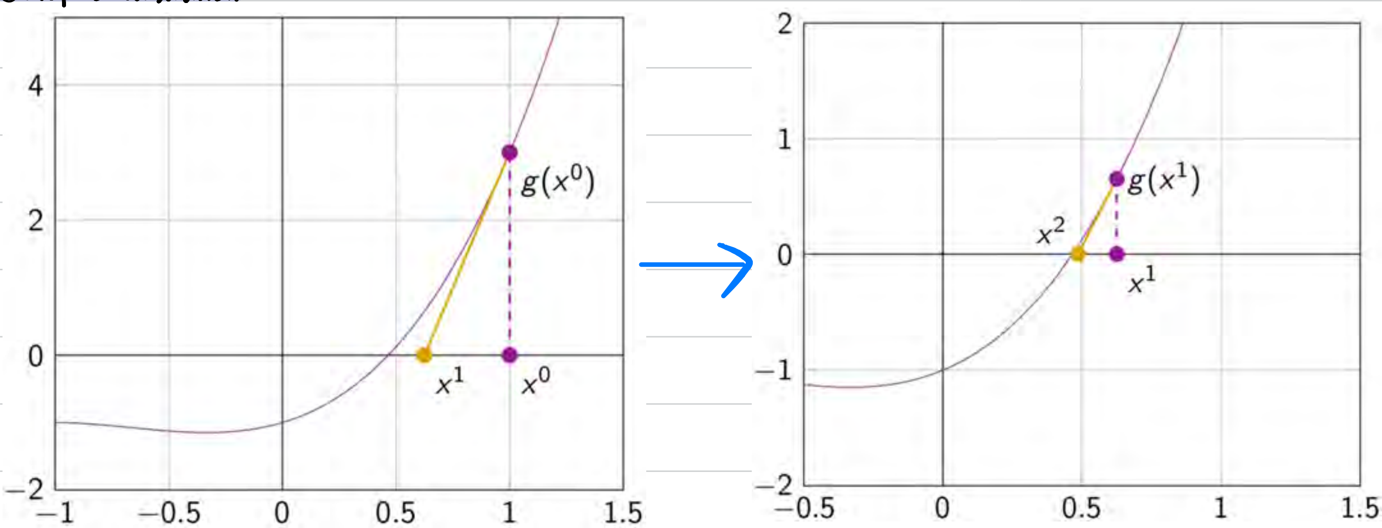
$$g(x^{k+1}) \approx g(x^k)+g'(x^{k+1})(x^{k+1}-x^k)=0$$

$$\longrightarrow g(x^k)+g'(x^k)(x^{k+1}-x^k)=0$$

$$\longrightarrow x^{k+1}=x^k-\frac{g(x^k)}{g'(x^k)}=x^k-\frac{\nabla f(x^k)}{\nabla^2 f(x^k)}$$

$\longrightarrow$ Continuously do iteration

③ Graphic illustration



3. Convergence in R

① Global convergence

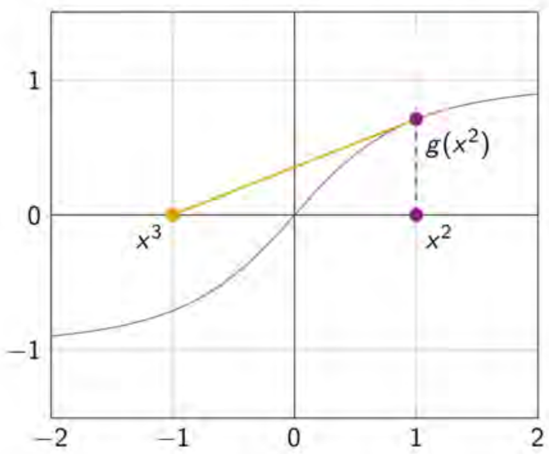
a. Interpretation

Since Newton's method is sensitive to the setup of initial point, then there is no global convergence for Newton's method.

b. Example

Newton's method may not converge for every initial point. (反复横跳，不收敛)

- Consider $g(x) = x/\sqrt{1+x^2}$. It has root $x = 0$.



② Local convergence

Newton's method is quadratic convergence:

$$\|x^{k+1} - x^*\| \leq C \|x^k - x^*\|^2$$

i.e. $\lim\limits_{k\to\infty} \dfrac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|^2} = C$, $C = \sup\limits_{x} \frac{1}{2}\left[\dfrac{g''(x)}{g'(x)}\right]$

4. Connection with Gradient descent method in $\mathbb{R}$

① Gradient descent method

$$x^{k+1} = x^k - \alpha f'(x^k)$$

② Newton's method

$$x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)}$$

③ Connection

In the 1-D case, Newton's method simply specifies α ($\alpha = \frac{1}{f''(x^k)}$).

- Direction: The same
- Stepsize: Specified

5. Newton's method in $\mathbb{R}^n$

① Derivation

$$g(x^{k+1}) = \nabla f(x^{k+1}) = \nabla f(x^k) + \nabla^2 f(x^k)(x^{k+1} - x^k) = 0$$

$$\longrightarrow x^{k+1} = x^k - (\nabla^2 f(x^k))^{-1} \nabla f(x^k)$$

② Descent direction

$$d = -(\nabla^2 f(x))^{-1} \nabla f(x)$$

Newton's direction is not necessarily a descent direction:

$$\nabla f(x)^T d = -\nabla f(x)^T (\nabla^2 f(x))^{-1} \nabla f(x)$$

- If f is convex, then $\nabla^2 f(x)$ is PSD, then $\nabla f(x)^T d \le 0$
- If $\nabla^2 f(x)$ is PD, then at this time, Newton's direction is a descent direction

What if Newton's direction is not a descent direction?

- if Newton's direction is a good descent direction $\longrightarrow$ Newton's direction with $\alpha^k = 1$ (Pure Newton's Method)
- if Newton's direction is a descent direction $\longrightarrow$ Newton's direction with Backtracking α^k (Newton's Method)
- if Newton's direction is not a descent direction $\longrightarrow$ Gradient descent direction with Backtracking α^k (只有这一步换)

③ Procedure

a. Find an initial point

b. If $\|\nabla f(x^k)\| \le \varepsilon$, then stop and output x^k

Otherwise, continue to do iteration

c. Choose the Newton's direction and judge (不是Descent direction则要换成Gradient descent direction噢!)

d. Determine the stepsize by Backtracking method

6. Convergence in R^n

① Global convergence

No global convergence

② Local convergence

$$\begin{cases} x^0 \text{ is close to } x^* \\ f \text{ is } C_L^{2,1}: \|\nabla^2 f(x)-\nabla^2 f(y)\| \le L\|x-y\| \\ \nabla^2 f(x) \text{ is PD locally} \end{cases}$$

(保证Newton's direction是个Descent direction)

$\longrightarrow$ Quatratic convergence: $\|x^{k+1}-x^*\| \le C\|x^k-x^*\|^2$

(前提很重要！否则Newton's method不一定converges quatratically!)

回忆课堂上的Linear convergence的例子～

7. Connection with Gradient descent method

Gradient descent method: Iteration 多 & 简单

Newton's method: Iteration 少 & 复杂 (更快，要求也更高)

指的是一般来说

L25–26 Projected gradient method

1. Motivation

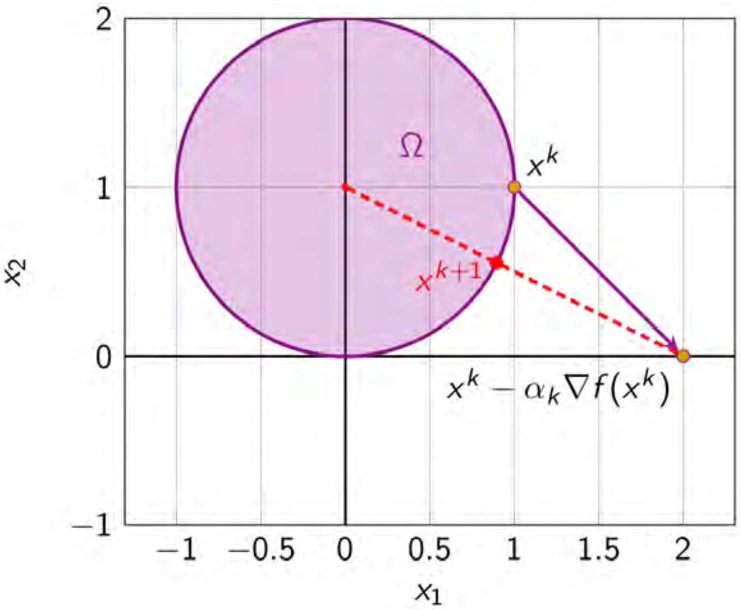
With gradient method:

- For unconstrained problem, x^{k+1} is always feasible
- For constrained problem, x^{k+1} may not be feasible

So, we want to find a way to map x^{k+1} to another feasible & better point

指的是 value descent

The "way" here is: Taking the projected x^{k+1} as x^{k+1}!



2. Projection

① Definition

Let Ω be a non-empty, closed, convex set,

then the projection of x onto Ω is defined as the y^* for:

$$\min_{y} \frac{1}{2}\|x-y\|^2, \quad s.t.\ y \in \Omega$$

At this time, we write $y^* = P_\Omega(x)$

② Example

a. Linear constraint

Given $\begin{cases} \Omega = \{y : Ay = b\} \\ x \in R^n \end{cases} \longrightarrow$ Find $P_\Omega(x)$

$$\boxed{\begin{aligned} &\min_{y} \frac{1}{2}\|y-x\|^2 \\ &s.t.\ Ay = b \end{aligned}}$$

We use KKT-condition to solve it:

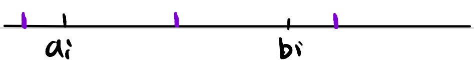
$$\begin{cases} \mathcal{L} = \frac{1}{2}\|y-x\|^2 + \mu(Ay-b) \longrightarrow \nabla \mathcal{L} = y - x + A^T\mu = 0 \\ Ay = b \end{cases}$$

$\longrightarrow P_\Omega(x) = y^* = x - A^T(AA^T)^{-1}[Ax-b]$

b. Box constraint

Given $\begin{cases} \Omega = \{y : y \in [a,b]\} \\ x \in R \end{cases} \longrightarrow$ Find $P_\Omega(x)$

此时我们数形结合:



$$\longrightarrow P_\Omega(x) = \begin{cases} a, & \text{if } x \le a \\ x, & \text{if } a < x < b \\ b, & \text{if } x \ge b \end{cases}$$

③ Projection theorem

Let Ω be a nonempty, closed and convex set, then:

Thm 1: The mapping $P_\Omega: R^n \to R^n$ is Lipschitz continuous with $L=1$

Thm 2: y^* is projection iff $(y^*-x)^T(y-y^*) \geq 0$, $\forall y \in \Omega$ $\triangleq \nabla f(y^*)\cdot(y-y^*) \geq 0$

Thm 3: x^* is a stationary point iff $\forall \lambda > 0$, $x^* - P_\Omega(x^* - \lambda \nabla f(x^*)) = 0$

proof:

$x^* = P_\Omega(x^* - \lambda \nabla f(x^*))$

By definition of projection, we have x^* be a solution of $\min_y \frac{1}{2}\|y-(x^*-\lambda\nabla f(x^*))\|^2$ s.t. $y \in \Omega$

Then by FOC, we have $\forall y \in \Omega$, $(x^*-(x^*-\lambda\nabla f(x^*)))^T(y-x^*) \geq 0$

$\rightarrow \forall y \in \Omega$, $\lambda \nabla f(x^*)^T(y-x^*) \geq 0$

$\rightarrow$ When $\lambda > 0$, $\forall y \in \Omega$, $\nabla f(x^*)^T(y-x^*) \geq 0$

$\rightarrow x^*$ is a stationary point

3. Projected gradient method

① Basic idea

$$\overline{x^{k+1}} = P_\Omega(x^{k+1}) = P_\Omega(x^k - \lambda_k \nabla f(x^k)) = x^k + \underbrace{[P_\Omega(x^k - \lambda_k \nabla f(x^k)) - x^k]}_{d^k}$$

$x^k \longrightarrow x^{k+1} \longrightarrow$ Projected x^{k+1}

② Descent direction

$d := [P_\Omega(x^k - \lambda_k \nabla f(x^k)) - x^k]$ is a descent direction

③ Process

a. Find an initial point

b. Select $\lambda_k > 0$ and then find the descent direction d^k

c. If $\|d^k\| \le \lambda_k \varepsilon$, then stop and output x^k
Otherwise, continue to do iteration

d. Choose a stepsize α_k with Backtracking method

e. Set $x^{k+1} = x^k + \alpha_k d^k$

4. Convergence

If λ_k is small enough: $\lambda_k \le \frac{2(1-\gamma)}{L}$ (有上界，也有下界), then $\alpha_k = 1$ for each iteration
Otherwise, we need to apply Backtracking method to do each iteration

L26-28 Integer Programming

1. Concepts

① Integer linear program (IP)

$$\min_x c^T x$$
$$\text{s.t. } Ax=b$$
$$x \geq 0$$
$$x \in \mathbb{Z}$$

② Mixed integer program (MIP)

Part of the decision variables have integer constraints

③ Binary integer program

Discrete → Integer → Binary

2. Logical constraints

① Motivation

Integer variables can be used to model logical constraints

② Logical constraint

Some of the constraints need to be satisfied.

③ Process

$\left\{\begin{array}{l}\text{Constraints} \\ \text{Binary constraint}\end{array}\right.$ ⟶ Logical constraints

Suppose we have m constraints $g_i(x) \leq 0$, $i=1,\cdots,m$, and at least K of them should be satisfied.

We introduce $y_i \in \{0,1\}$, then

$$\begin{cases} g_1(x) \leq My_1, \cdots, g_m(x) \leq My_m \\ \sum_{i=1}^{m}(1-y_i) \geq K \end{cases}$$

$\begin{cases} y_i=0 & \text{Satisfied} \\ y_i=1 & \text{Can be not satisfied} \end{cases}$ ⟶ $\begin{cases} \text{If } g_i(x) > 0, \text{ choose } y_i = 1 \\ \text{If } g_i(x) \leq 0, \text{ choose } y_i = 0/1 \end{cases}$

3. Formulation

① Exp 1: Warehouse location problem

a. Setup

- There are n warehouses available for use.
- Opening warehouse i has a fixed operating costs f_i.
- There are m customers.
- Customer j has demand d_j. It costs c_{ij} to ship one unit of product from warehouse i to customer j.
- The objective is to satisfy all customers' demands, while minimizing the total costs (operating + shipment).

b. Decision variables

y_i: whether to open warehouse i or not

x_{ij}: the units to ship from warehouse i to customer j.

c. Formulating

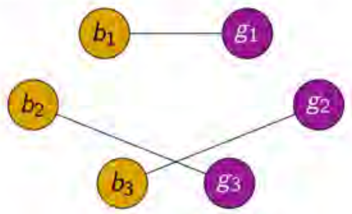
$$
\begin{aligned}
\min_{x,y} \quad & \sum_{i=1}^{n} f_i y_i + \sum_{i,j} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_{ij} \geq d_j, \ \forall j \\
& x_{ij} \leq d_j y_i, \ \forall i, j \\
& x_{ij} \geq 0 \\
& y_i \in \{0,1\}
\end{aligned}
$$

② Exp 2: Matching problem

a. Setup

Setup: Assume there are n girls and n boys' information in a dating website.

- Each girl i has rated boy j with a score v_{ij}.
- The website wants to match one girl with one boy so as to maximize the total score of the matching.



b. Decision variables

x_{ij}: whether there is a match between i & j

c. Formulating

$$\max_{x} \sum_{i=1}^{n}\sum_{j=1}^{n} v_{ij}x_{ij}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \forall i$$

$$\sum_{i=1}^{n} x_{ij} = 1, \forall j$$

$$x_{ij} \in \{0,1\}$$