LPMS Note

1 General LP

General LP Problem: **Decision Variables**: x_i **parameters**: a_{ij}, b_i, c_i **Objective Function**: f **Constraints**: subject to 后边的部分

LP problem can be written as:

$$\begin{aligned} \text{maximize} \quad & f = c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{subject to} \quad & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1 \\ & a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2 \\ & \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le b_m \\ & x_1 \ge 0, x_2 \ge 0, \cdots, x_n \ge 0 \end{aligned}$$

We can also write the LP problem in matrix form:

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}, \mathbf{c} = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix}, \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

$$\begin{aligned} \text{maximize} \quad & f = \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \le \mathbf{b} \quad \mathbf{x} \ge 0 \end{aligned}$$

Feasible Solution: If $\mathbf{x}$ satisfies all constraints (i.e. $A\mathbf{x} \le \mathbf{b}$), then $\mathbf{x}$ is a feasible solution. (可行解) **Optimal Sol**: (最优解) (可多个)

Find Optimal Solution: **Graphical Method**: 略. **Vertex Enumeration**: 所有顶点检查, 找出最优解. **Simplex Method**: 后面详述.

Slack Variables: For each inequality constraint, we introduce a slack variable x_i $(i > n)$ to convert it to an equation. (松弛变量)

LP problem can be written as: ps: $x_i \ge 0$ $(i > n)$.

$$\begin{aligned} \text{maximize} \quad & f = c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{subject to} \quad & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n + x_{n+1} = b_1 \\ & a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n + x_{n+2} = b_2 \\ & \vdots \\ & a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n + x_{n+m} = b_m \\ & x_1 \ge 0, x_2 \ge 0, \cdots, x_{n+m} \ge 0 \end{aligned}$$

We can also write the LP problem in matrix form:

$$\overline{A} = [\, A \ \ I_m \,], \mathbf{b} = \mathbf{b}, \overline{\mathbf{c}} = \begin{pmatrix} \mathbf{c} \\ \mathbf{0}_{m\times 1} \end{pmatrix}, \mathbf{x} = \begin{pmatrix} \mathbf{x} \\ \vdots \\ x_{n+m} \end{pmatrix}$$

$$\begin{aligned} \text{maximize} \quad & f = \overline{\mathbf{c}}^T\mathbf{x} \\ \text{subject to} \quad & \overline{A}\mathbf{x} = \mathbf{b} \quad \mathbf{x} \ge 0 \end{aligned}$$

Solving: $\mathbf{x} = \mathbf{x}_b + \mathbf{v}$ where $\overline{A}\mathbf{v} = 0$ and $\mathbf{x}_b = [\mathbf{0}\ \mathbf{b}]^T$

Feasible Region: It's the set K of the solutions to $\overline{A}\mathbf{x} = \mathbf{b}$ **Convex Set**: The set K is convex if $\forall \mathbf{x}, \mathbf{x}' \in K, \forall \theta \in [0,1], \mathbf{x}_\theta = (1-\theta)\mathbf{x} + \theta\mathbf{x}' \in K$.

Vertex on Convex Set: A vertex of the convex set K is a point $\mathbf{x} \in K$ which doesn't lie strictly inside any line segment connecting two points in K.

· **Theorem**: If LP has a unique optimal solution is a vertex. **Theorem**: If LP has a non-unique solution, ∃ optimal solution at vertex

2 Simplex Method

2.1 Simplex Algorithm

Solve LP Problem: Assume $f = \overline{\mathbf{c}}^T\mathbf{x}$ with $\overline{A}\mathbf{x} = \mathbf{b}$ and $\mathbf{x} \ge 0$.

1. 修改 x_i/列顺序,A 中换顺序后得到可逆的 $B_{m\times m}$ $\Rightarrow$ $\overline{A}\mathbf{x} = \mathbf{b} \Leftrightarrow B\mathbf{x}_B + N\mathbf{x}_N = \mathbf{b}$ $\overline{\mathbf{c}} \to [\mathbf{c}_B\ \mathbf{c}_N]^T$ $\overline{A} \to [B\ N]$ $\mathbf{x} \to [\mathbf{x}_B\ \mathbf{x}_N]^T$ Let $\widehat{\mathbf{b}} = B^{-1}\mathbf{b}$ $\widehat{N} = B^{-1}N$
2. **Solution**: $\mathbf{x}_B = \widehat{\mathbf{b}} - \widehat{N}\mathbf{x}_N$ If $\mathbf{x}_N = 0 \Rightarrow$ It's a basic solution. (But we need to check whether $\mathbf{x}_B \ge 0$) Using $\mathcal{B}, \mathcal{N}$: Index set of independent/else.
3. **Basic Variables**: $\mathbf{x}_B$ **Nonbasic Variables**: $\mathbf{x}_N$
4. **At Basic Solution**: $\mathbf{x}_B = \widehat{\mathbf{b}}, \mathbf{x}_N = \mathbf{0}$ $\Rightarrow$ If $\widehat{\mathbf{b}} \ge 0$. Corresponding $\mathbf{x}$ is: ^{1} vertex of K; ^{2} **Basic Feasible Solution (BFS)**
5. **Basic Costs**: $\mathbf{c}_B^T$ **Nonbasic Costs**:$\mathbf{c}_N^T$ **Reduced Costs**: $\hat{\mathbf{c}}_N = \mathbf{c}_N - \widehat{N}^T\mathbf{c}_B = \mathbf{c}_N - N^TB^{-T}\mathbf{c}_B$ $\widehat{f} = \mathbf{c}_B^T\widehat{\mathbf{b}}$ $\mathbf{x}_B, \mathbf{x}_N \ge 0$
6. **Objective Value**: $f = \overline{\mathbf{c}}^T\mathbf{x} = \mathbf{c}_B^T\mathbf{x}_B + \mathbf{c}_N^T\mathbf{x}_N = \widehat{f} + \hat{\mathbf{c}}_N^T\mathbf{x}_N$ If $\hat{\mathbf{c}}_N \le 0$, then $f \le \widehat{f}$ $\Rightarrow$ Corresponding $\mathbf{x}$ is optimal.
7. If $\hat{\mathbf{c}}_N \le 0$ doesn't hold, we using **Simplex Algorithm**.

Simplex Algorithm:

1. **Initial Basic Feasible Solution**: Try $\mathcal{B} = \{n+1, \dots, n+m\}$ and $\mathcal{N} = \{1, \dots, n\}$ $\Rightarrow$ $B = I, N = A, \mathbf{x}_B = \widehat{\mathbf{b}} = \mathbf{b}, \mathbf{x}_N = \mathbf{0}, \mathbf{c}_B = \mathbf{0}, \mathbf{c}_N = \mathbf{c}$
2. If $\mathbf{b} \ge 0$. $\Rightarrow$ Basis is feasible + cont. Else: Basis B is not feasible. Go to "Cases If $\mathbf{b} \ngeq 0$"
3. If $\hat{\mathbf{c}}_N \le 0$. $\Rightarrow$ Optimal Solution Else: cont.
4. Let $q' \in \mathcal{N}$ 对应 $\hat{\mathbf{c}}_N$ 中最大 positive 分量的 index 同理, 对应的最大正分量值为 $\hat{c}_q$ 对应的 variable 为 $x_{q'}$ 对应 N 中的 $[q']$ 列为 $\mathbf{a}_q$
5. Let $\widehat{\mathbf{a}_q} = B^{-1}\mathbf{a}_\mathbf{q}$. If $\widehat{\mathbf{a}_q} \le 0 \Rightarrow$ LP is unbounded. Else: LP is bounded + cont.
6. Let $p' \in \mathcal{B}$ be the index corresponding to $p = \arg\min_{i=1,\dots,m\,;\,\hat{a}_{iq}>0} \frac{\hat{b}_i}{\hat{a}_{iq}}$ p 是对应的 index,not value 用 $\overline{\alpha} = \frac{\hat{b}_p}{\hat{a}_{pq}}$ 代表值
   $\hat{b}_i, \hat{a}_{iq}$ 代表 $\widehat{\mathbf{b}}$ 的第 i 个分量, $\widehat{\mathbf{a}_q}$ 的第 i 个分量
   p' 代表能使 $\frac{\hat{b}_i}{\hat{a}_{iq}}$ 的值最小的 index, 前提条件是 $\hat{a}_{iq} > 0$ 对应的 variable 为 $x_{p'}$
7. Exchange p' and q' between $\mathcal{B}$ and $\mathcal{N}$ $\Rightarrow$ Update $\mathcal{B}, \mathcal{N}, B, N, \widehat{\mathbf{b}}, \dots$ ps: New $B := B + (a_q - Be_p)e_p^T$
8. Go to 3.

Case: If $\mathbf{b} \ngeq 0$: Phase I Problem

1. Subtract **artificial variables** $x_{n+m+1}, \dots, x_{n+m+m} \ge 0$ and change objective function f to:

$$\begin{array}{llllllllllll} \max & f = & & & & & & -x_{n+m+1} & \cdots & -x_{n+2m} & & \\ \text{s. t.} & a_{11}x_1 & + & \cdots & + & a_{1n}x_n & +x_{n+1} & -x_{n+m+1} & & & = & b_1 \\ & & \vdots & & \vdots & & \ddots & & \ddots & & \vdots & \\ & a_{m1}x_1 & + & \cdots & + & a_{mn}x_n & +x_{n+m} & & & -x_{n+2m} & = & b_m \\ & x_1 \ge 0, & & \dots, & & x_{n+2m} \ge 0 & & & & & & \end{array}$$

2. Let Basic Variables: $\mathbf{x}_B =$ 第 i 个元素是: $x_{n+i} = b_i$ (if $b_i \geq 0$)   是:$x_{n+m+i} = -b_i$ (if $b_i < 0$) ⋆

   Let Nonbasic Variables: $\mathbf{x}_N =$ 第 i 个元素是: $x_{n+m+i} = 0$ (if $b_i \geq 0$)   是:$x_{n+i} = b_i$ (if $b_i < 0$)

   Let Basic Matrix $B =$ 第 i 列是: $\mathbf{e}_i$ (if $b_i \geq 0$) 否则: $-\mathbf{e}_i$ 其他列是 N 的对应列   Let $\hat{\mathbf{b}} = B^{-1}\mathbf{b} = |\mathbf{b}| \geq 0$

   Let $\mathbf{c}_B = x_B$ 对应的 f 中的系数 $\mathbf{c}_N =$ 同理 $\hat{\mathbf{c}}_N = \mathbf{c}_N - N^T B^{-T} \mathbf{c}_B$

3. **Size of Infeasibility**: is $\sum_{i=1}^{m} x_{n+m+i}$ $f = -\sum_{i=1}^{m} x_{n+m+i} \leq 0$

4. If $f = 0$, $\mathbf{x}$ is a BFS. ⇒ Go to **Phase II Problem**(恢复到之前的 f 并删除 artificial variables)

   Else: Go to **Simplex Algorithm** 得到辅助问题 Phase I 的最优解.   **If** $f < 0$: LP is infeasible.   **If** $f = 0$: LP is feasible. ⇒ **Phase II**

Case: If $\mathbf{b} \ngeq 0$: Phase II Problem

2.2 More thing about Simplex Algorithm

Degeneracy: If $\hat{\mathbf{b}}$ has any zero component, then $\mathbf{x}$ is a **degenerate vertex.**   如果 $\hat{\mathbf{b}}$ 的第 p 个分量 $\hat{\mathbf{b}}_p = 0$, 那么单纯形法可能会陷入循环.

Theorem: If no degenerate situation, then Simplex Algorithm will terminate in a finite number of steps.

Examples: 1.The worst case of Simplex Algorithm. 2. Cyclic case of Simplex Algorithm.

1. **Klee-Minty Problem**: max $f = \sum_{j=1}^{n} 10^{j-1} x_j$; s.t. $x_i + 2\sum_{j=i+1}^{n} 10^{j-i} x_j \leq 100^{n-i}$ for $i = 1, \ldots, n$; $x_i \geq 0$

   has 2^n vertices.   **Worst Case**: $2^n - 1$ iterations.

2. **Hall-McKinnon Problem**: max $f = x_1 - 5.5x_2 + 0.75x_3 - 5.75x_4$; s.t. $2.5x - 19.5x_2 - 3.5x_3 + 19.5x_4 + x_5 = 0$; $0.5x_1 - 3.5x_2 - 0.5x_3 + 3.5x_4 + x_6 = 0$; $x_i \geq 0$

2.3 Sparsity LP Problem

Implementation: 计算/编程中的计算化简   Consider: $\hat{\mathbf{b}} = B^{-1}\mathbf{b}$ $\hat{\mathbf{c}}_N = \mathbf{c}_N - N^T B^{-T} \mathbf{c}_B$ $\hat{\mathbf{a}}_q = B^{-1}\mathbf{a}_q$

1. Solve: $B\hat{b} = b$ to get: $\hat{b} = B^{-1}b$
2. Solve: $B^T \pi = c_B$ to get: $\pi = B^{-T} c_B$
3. Solve: $\hat{c}_N = c_N - N^T \pi$ to get: $\hat{c}_N = c_N - N^T B^{-T} c_B$
4. Solve: $B^T \hat{a}_q = a_q$ to get: $\hat{a}_q = B^{-T} a_q$
5. Matrix B is **sparse** and **invertible**. And in Simplex Algorithm, the next matrix B obtained by $B := B + (a_q - Be_p)e_p^T$

Sparse LP Problem: For LP problem: max $f = \mathbf{c}^T\mathbf{x}$ s.t. $A\mathbf{x} = \mathbf{b}$ $\mathbf{x} \geq 0$

It is **sparse** if the matrix A is sparse. (i.e. Most of the elements of A are zero)

Find Inverse of B: Use **Guassian Elimination** to decomposition B into LU where L is lower triangular and U is upper triangular.

Then we can solve $B\mathbf{x} = \mathbf{b}$ by solving ^{1} $L\mathbf{y} = \mathbf{b}$ and ^{2} $U\mathbf{x} = \mathbf{y}$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21}u_{11} & l_{21}u_{12} + u_{22} & l_{21}u_{13} + u_{23} & l_{21}u_{14} + u_{24} \\ l_{31}u_{11} & l_{31}u_{12} + l_{32}u_{22} & l_{31}u_{13} + l_{32}u_{23} + u_{33} & l_{31}u_{14} + l_{32}u_{24} + u_{34} \\ l_{41}u_{11} & l_{41}u_{12} + l_{42}u_{22} & l_{41}u_{13} + l_{42}u_{23} + l_{43}u_{33} & l_{41}u_{14} + l_{42}u_{24} + l_{43}u_{34} + u_{44} \end{bmatrix}$$

By Solving ■ → ■ → ■ → ■ → ■ → ■ → ■

3 Example of format written in LP

Example| 构建 LP 问题模板:

Defining decision variables:

Let x_1 be the number of 1kg packets of Breakfast Blend made each day.

Let x_2 be the number of 1kg packets of Dinner Blend made each day.

Total income is: $f_I = 1.16x_1 + 1.42x_2$.

Total cost is: $f_C = 0.18x_1 + 0.36x_2 + 0.28x_1 + 0.16x_2 = 0.46x_1 + 0.52x_2$.

Thus, total profit is:$f_I - f_C = 1.16x_1 + 1.42x_2 - 0.46x_1 - 0.52x_2 = 0.7x_1 + 0.9x_2$.

The objective is to maximize total profit $f = 0.7x_1 + 0.9x_2$

Constraints:

1. The number of kilos of arabica used, $0.3x_1 + 0.6x_2$, must not exceed the supply of 1200kg.
2. The number of kilos of robusta used, $0.7x_1 + 0.4x_2$, must not exceed the supply of 1500kg.
3. The total number of kilos of coffee made, $x_1 + x_2$, must not exceed the capacity of 2400kg.

Thus, the LP Problem to be solved is:

$$\begin{aligned} \text{maximize} \quad & f = 0.7x_1 + 0.9x_2, \\ \text{subject to} \quad & 0.3x_1 + 0.6x_2 \leq 1200, \\ & 0.7x_1 + 0.4x_2 \leq 1500, \\ & x_1 + x_2 \leq 2400, \\ & x_1, x_2 \geq 0. \end{aligned}$$

Introducing Slack Variables:

$$\begin{aligned} \text{maximize} \quad & f = 0.7x_1 + 0.9x_2, \\ \text{subject to} \quad & 0.3x_1 + 0.6x_2 + x_3 = 1200, \\ & 0.7x_1 + 0.4x_2 + x_4 = 1500, \\ & x_1 + x_2 + x_5 = 2400, \\ & x_1, x_2, x_3, x_4, x_5 \geq 0. \end{aligned}$$