Course: Comtemporary Algorithms T.II/2019-20

Lecture 14: Linear Programming I

13 January 2020

Lecturer: Dr. Kanat Tangwongsan *Scribe: Kanokpon & Kanokpon*

1 Algebraic view

1.1 Introduction to linear programming

The aim is to minimizing costs of various systems while meeting different constraints.

Linear program is consist of

- variables: $x_1, x_2, \ldots x_n$
- linear constrain: eg. $2x_2 - 5x_7$
- linear objective: function that we aim to maximize/minimize

Example

Objective minimize $3x_1 + 2x_2$

subj to: $\boxed{\begin{aligned} x_1 &\geq 0 \\ x_1 + x_2 &\leq 2 \\ x_1 - x_2 &\geq 1 \end{aligned}}$ Feasible(setting of x_1, x_2 that satisfy the constraints)

1.2 Diet problem

How to spend the least money while getting enough nutrient.

- n food
- m nutrients
- a_{ij} -amount of nutrient i in a unit of food j
- b_i -minimize need of nutrient i
- c_j unit cost of food j
- x_j-amount of food j we consume

Nutrient i:
$\forall_i a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + \dots a_{in}x_n \geq b_i -$ (enough nutrient i)
$\vdots$
$x_j \geq 0$

minimize $c_i x_1 + c_2 x_2 + \dots c_n x_n$
minimize $\boxed{\sum_{j=1}^{n} c_j x_j} = c^T x$
subj to:
$\forall_i : \sum_{j=1}^{n} a_{ij} \geq b_i$
$\boxed{\forall_j : x_j \geq 0} \rightarrow \vec{x} \geq \vec{0}$

In term of vector

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad A\vec{x} = \begin{bmatrix} a_{11} & a_{12} \dots & a_{1n} \\ a_{21} & a_{22} \dots & a_{2n} \\ & \ddots & a_{m,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$\vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \qquad = \begin{bmatrix} \sum_j a_{1j}x_j \\ \sum_j a_{2j}x_j \\ \sum_j a_{ij}x_j \end{bmatrix} \geq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

$$\vec{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

General form		Standard form
Minimize $\vec{c}^T \vec{x}$ subj to $A\vec{x} \geq \vec{b}$	$\underset{\text{stack, } x^+ x^- \text{ variable}}{\overset{\text{change} = \rightarrow \geq \& \leq}{\leftrightarrows}}$	Minimize $c^T x$ subj to $A'\vec{x} = \vec{b}'$ $\vec{x} \geq 0$

$$\begin{bmatrix} \frac{A}{I} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} A_x \\ \dots \\ x \end{bmatrix} \geq \begin{bmatrix} \frac{\vec{b}}{\vec{0}} \end{bmatrix}$$

Tricks

1. minimize $c^T x$ = maximize $(-c)^T x$
2. Upper bound constraints can be recast to be equivalent to lower bound constraints
   $\sum s_i x_i \geq b \iff \sum(-a_i)x_i \leq -b$
3. Inequality constraints can get to an equality constraints
   $\sum a_i x = b \iff \sum a_i x_i \geq b$ & $\sum a_i x_i \leq b$
4. x is unconstrained(can be positive or negative) $\iff x^+ - x^- \; x^+ \geq 0, x^- \geq 0$

$$\alpha \geq \beta \qquad\qquad \sum a_i x_i \geq b_j$$

$$\beta + \underbrace{t}_{\geq 0} = \alpha \qquad\qquad \sum a_i x_j - \underbrace{s_j}_{\text{slack variable}} = b_j$$

Remarks

- n vars + m constants $\rightarrow O(n+m)$ vars & $O(n+m)$ constraints
- Feasible sol^1 : $x \in \mathbb{R}^n$ satisfying all constrains
- Optimal sol^1 : $x \in \mathbb{R}^n$: feasible sol^n & minimize/maximize the obj function

1.3 Fourier-Motzkin elimination

1. Introduce a new var
2. Reduce dimensions

For variable x_1 arrange constraints into three groups

1. has positive coefficients
2. has negative coefficients
3. do not involve x_1 at all

$$\begin{aligned} &\text{min } c^T x \\ &\text{subj to} Ax \geq b \end{aligned} \iff \begin{aligned} \text{min} \qquad & x_n + 1 \\ \text{subj to} \qquad & Ax \geq b \\ & c^T x \leq x_{n+1} \end{aligned}$$

Example

$$\begin{aligned} -2x_1 \quad -x_2 \quad +3x_3 &\geq 5 && (1) \\ -x_1 \quad +2x_2 \quad -4x_3 &\geq 1 && (2) \\ 5x_2 \quad +7x_3 &\geq 4 && (3) \\ x_1 \qquad\qquad -7x_3 &\geq 1 && (4) \end{aligned}$$

How to eliminate x_1?

$$
\begin{aligned}
p &= \text{constraints where the coeff of } x_1 > 0 \\
N &= \text{constraints where the coeff of } x_1 < 0 \\
Z &= \text{constraints where the coeff of } x_1 = 0 \\
&= [M] \backslash (P \cup N)
\end{aligned}
\qquad
\begin{aligned}
p &= \{1, 4\} \\
N &= \{2\} \\
Z &= \{3\}
\end{aligned}
$$

On P constraints

$$i \in [P] : \underbrace{\left(\frac{a_{i1}x_1}{a_{i1}}\right)}_{1} + \frac{a_{i2}x_2}{a_{i1}} + \cdots + \frac{a_{in}x_n}{a_{i1}} \geq \frac{b_i}{a_{i1}}$$

$$\iff$$

$$x_1 \geq \frac{b_i}{a_{i1}} - \left[\sum_{j=2}^{n} \left(\frac{a_{ij}}{a_{i1}}\right) xj\right] - (\text{Lower bound } x_1 \geq blah_i)$$

On N constraints

$$i \in [N] : \underbrace{\left(\frac{a_{i1}x_1}{a_{i1}}\right)}_{1} + \frac{a_{i2}x_2}{a_{i1}} + \cdots + \frac{a_{in}x_n}{a_{i1}} \leq \frac{b_i}{a_{i1}} - (\text{Upper bound } x_1 \leq blah_i)$$

$$\iff$$

$$x_1 \leq \frac{b_i}{a_{i1}} - \left[\sum_{j=2}^{n} \left(\frac{a_{ij}}{a_{i1}}\right) xj\right]$$

We will get new constraints as follows: for each $i \in P$ and $i' \in N$, we get $blah_i \leq x_i$ and $x_{i'} \leq blah_{i'}$ so we will get $blah_i \leq blah_{i'}$

We took $|P| + |N|$ constraints and change it to $|P| \cdot |N|$ which is $\leq m^2$, so for each n variables we will end up with m^{2^n} constraints

1.3.1 Gaussian Elimination

take the first $\sum_j a_{1j}x_j = b_1$ and rewriting it as $x_1 = a_{11}^{-1}(b_1 - \sum_{j=2}^{n} a_{1j}x_j)$ and substituting this into the other constraints (This can be done in polynomial time)

- #operations done in polynomial in n and m
- The size of the number in the intermediate stages of the algorithm are $poly(n, m, \log |a_{ij}|)$

1.4 Equational Form Solving

Assume our LP is in this form

$$min\{c^T x | Ax = b, x \geq 0\}$$

Assume $A_x = b$ has ≥ 1 sol^n

Without loss of generality the rows of A are linearly independent ($rank(A) \geq m$)

Given a subset $B \subseteq [n]$ we define A_B to be the concatenation of the B columns of A and s_B to be the column vector consisting of the variables $\{x_i | i \in B\}$ If A_B is linearly independent then A_B has full rank & invertible($x_B = A_B^{-1} b$) so

$$A_B x_B = b$$