Lecture 2: Intro to LP, Linear algebra review.

Logistics

- Complete the section time survey today
  [https://forms.gle/NPtYQUHv9FN6ndBo6](https://forms.gle/NPtYQUHv9FN6ndBo6)
- Complete assignment 0 today
- Office hours will be posted on course website today (always check the page before you go)
- Assignment 1 will be posted today (due Tue. 9/17)
Lecture 2: Lesson Plan

- What is an LP?
- Graphical and algebraic correspondence
- Problems in canonical form
- LP in matrix form. Matrix review.

Jensen & Bard: 2.1-2.3, 2.5, 3.1 (can ignore the two definitions for now), 3.2

Available in Cabot Science Library.

Linear Programming

- Maximizing (or minimizing) a linear function subject to a finite number of linear constraints

\[
\begin{align*}
\text{max} & \quad \sum_{j=1}^{n} c_j x_j & \quad \text{objective function} \\
\text{s.t.} & \quad \sum_{j=1}^{n} a_{ij} x_j \leq b_i & \quad (i = 1, \ldots, m) \\
& \quad x_j \geq 0 & \quad (j = 1, \ldots, n) \\
\end{align*}
\]

Decision variables: \(x_j\)
Parameters: \(c_j, a_{ij}\)
Standard Inequality Form

$$\begin{align*}
\text{max} & \quad \mathbf{c}^T \mathbf{x} \\
\text{s.t.} & \quad \mathbf{A} \mathbf{x} \leq \mathbf{b} \\
& \quad \mathbf{x} \geq 0
\end{align*}$$

$$\mathbf{c}^T = (c_1, \ldots, c_n)$$

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

Standard Equality Form

$$\begin{align*}
\text{max} & \quad \mathbf{c}^T \mathbf{x} \\
\text{s.t.} & \quad \mathbf{A} \mathbf{x} = \mathbf{b} \\
& \quad \mathbf{x} \geq 0
\end{align*}$$

$$\mathbf{c}^T = (c_1, \ldots, c_n)$$

$$\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$$
A Little History

• The field of linear programming started in 1947 when George Dantzig designed the “simplex method” for solving U.S. Air Force planning problems.
• Dantzig was deciding how to use the limited resources of the Air Force.
• “planning” == “programming”
  – “program” was a military term that referred to plans or proposed schedules for training, logistical supply, or deployment of combat units.
  – this naming sometimes called “Dantzig’s great mistake”

Terminology for Solutions of LP

• A feasible solution
  – A solution that satisfies all constraints
• An infeasible solution
  – A solution that violates at least one constraint
• Feasible region
  – The region of all feasible solutions
• An optimal solution
  – A feasible solution that has the most favorable value of the objective function
Example: Marketing Campaign

• Ad on news page—get 7m high-income women, 2m high-income men. $50,000
• Ad on sports page—get 2m high-income women and 12m high-income men. $100,000
• Goal: 28m women, 24m men; min cost. How many of each ad to buy? (Can buy fractions!)

\[
\begin{align*}
\text{min } & \quad z = 50x_1 + 100x_2 \\
\text{s.t. } & \quad 7x_1 + 2x_2 \geq 28 \\
& \quad 2x_1 + 12x_2 \geq 24 \\
& \quad x_1, x_2 \geq 0
\end{align*}
\]

Graphical version of problem
(solution is \(x_1=3.6\), \(x_2=1.4\), value 320)

\[
\begin{align*}
\text{min } & \quad z = 50x_1 + 100x_2 \\
\text{s.t. } & \quad 7x_1 + 2x_2 \geq 28 \\
& \quad 2x_1 + 12x_2 \geq 24 \\
& \quad x_1, x_2 \geq 0
\end{align*}
\]

Solution is at an extreme point of feasible region!
Example: Multiple Opt. Solutions

\[ \text{max } 3x_1 - x_2 \]
\[ \text{s.t. } 15x_1 - 5x_2 \leq 30 \]
\[ 10x_1 + 30x_2 \leq 120 \]
\[ x_1, x_2 \geq 0 \]

Note: still extremal optimal solutions

Example: Unbounded Objective

\[ \text{max } -x_1 + x_2 \]
\[ \text{s.t. } -x_1 + 4x_2 \geq 0 \]
\[ x_1 \leq 4 \]
\[ x_1, x_2 \geq 0 \]
Example: Infeasible Problem

\[ \begin{align*}
\text{max} & \quad x_1 + x_2 \\
\text{s.t.} & \quad 3x_1 + x_2 \geq 6 \\
& \quad 3x_1 + x_2 \leq 3 \\
& \quad x_1, x_2 \geq 0
\end{align*} \]

Solving LPs

- Transform to the \textit{canonical form} (note: this is NOT the “standard equality form”)
- Work with \textbf{basic feasible solutions}
- \textbf{Iterate}: solution improvement
  - From one BFS to the next…
Canonical Form

\[
\begin{align*}
\max z &= 0x_1 + 0x_2 - 3x_3 - x_4 + 20 \\
\text{s.t.} \quad & \\
& x_1 - 3x_3 + 3x_4 = 6 \\
& x_2 - 8x_3 + 4x_4 = 4 \\
& x_j \geq 0
\end{align*}
\]

1. **Maximization**
2. **RHS coefficients are non-negative**
3. **All constraints are equalities**
4. **Decision variables all non-negative**
5. One decision variable is “isolated” in each constraint:
   - a +1 coefficient.
   - does not appear in any other constraint
   - zero coefficient in objective

Why might this be useful??

Basic Feasible Solution

\[
\begin{align*}
\max z &= 0x_1 + 0x_2 - 3x_3 - x_4 + 20 \\
\text{s.t.} \quad & \\
& \text{\textcircled{1}} x_1 - 3x_3 + 3x_4 = 6 \\
& \text{\textcircled{2}} x_2 - 8x_3 + 4x_4 = 4 \\
& x_j \geq 0
\end{align*}
\]

Canonical form has an associated basic feasible solution in which the isolated variables (basic vars) are non-zero and the rest (non-basic vars) are zero.
Basic Feasible Solution

\[
\begin{align*}
\text{max } z &= 0x_1 + 0x_2 - 3x_3 - x_4 + 20 \\
\text{s.t.} & \quad -3x_3 + 3x_4 = 6 \\
& \quad -8x_3 + 4x_4 = 4 \\
& \quad x_j \geq 0
\end{align*}
\]

Canonical form has an associated basic feasible solution in which the isolated variables (basic vars) are non-zero and the rest (non-basic vars) are zero.

Here, set \(x_1 = 6, x_2 = 4, x_3 = 0, x_4 = 0\).

Optimal in this example as well. (Why?)
**Solution Improvement**

\[
\max z = 0x_1 + 0x_2 - 3x_3 + x_4 + 20
\]

s.t.

\[
x_1 - 3x_3 + 3x_4 = 6
\]

\[
x_2 - 8x_3 + 4x_4 = 4
\]

\[
x_j \geq 0
\]

Current BFS: \(x_1 = 6, x_2 = 4, x_3 = 0, x_4 = 0\).

Let's increase \(x_4\). Need to decrease \(x_1\) and \(x_2\) (keep \(x_3 = 0\)) to keep feasible.
Solution Improvement

\[
\begin{align*}
\max \ z &= 0x_1 + 0x_2 - 3x_3 + x_4 + 20 \\
\text{s.t.} \quad &x_1 - 3x_3 + 3x_4 = 6 \\
&x_2 - 8x_3 + 4x_4 = 4 \\
&x_j \geq 0
\end{align*}
\]

Current BFS: \(x_1 = 6, x_2 = 4, x_3 = 0, x_4 = 0\).
Let's increase \(x_4\). Need to decrease \(x_1\) and \(x_2\) (keep \(x_3 = 0\)) to keep feasible. Second constraint becomes binding.
Obtain new solution: \(x_1 = 3, x_2 = 0, x_3 = 0, x_4 = 1\). Value 21.

Corresponds to a new canonical form. Isolated vars: \(x_1\) and \(x_4\).
“pivot on \(x_4\) in the second constraint”
“pick something to enter, something forced to leave”

---

Solution Improvement

\[
\begin{align*}
\max \ z &= 0x_1 + 0x_2 - 3x_3 + x_4 + 20 \\
\text{s.t.} \quad &x_1 - 3x_3 + 3x_4 = 6 \\
&x_2 - 8x_3 + 4x_4 = 4 \\
&x_j \geq 0
\end{align*}
\]

Current BFS: \(x_1 = 6, x_2 = 4, x_3 = 0, x_4 = 0\).
Let's increase \(x_4\). Need to decrease \(x_1\) and \(x_2\) (keep \(x_3 = 0\)) to keep feasible. Second constraint becomes binding.
Obtain new solution: \(x_1 = 3, x_2 = 0, x_3 = 0, x_4 = 1\). Value 21.

Corresponds to a new canonical form. Isolated vars: \(x_1\) and \(x_4\).
New Canonical Form

After linear transformations:

\[
\begin{align*}
\text{max } z &= 0x_1 - \frac{1}{4}x_2 - x_3 + 0x_4 + 21 \\
\text{s.t. } &x_1 - \frac{3}{4}x_2 + 3x_3 = 3 \\
&\frac{1}{4}x_2 - 2x_3 + x_4 = 1 \\
&x_j \geq 0
\end{align*}
\]

New BFS is \(x_1=3, x_2=0, x_3=0, x_4=1\), and optimal.

Geometric Interpretation of Solution Improvement

\[
\begin{align*}
\text{max } z &= 0x_1 + 0x_2 - 3x_3 + x_4 + 20 \\
\text{s.t. } &x_1 - 3x_3 + 3x_4 = 6 \\
&-8x_3 + 4x_4 = 4 \\
&x_j \geq 0
\end{align*}
\]

\(x_1=6, x_2=4, x_3=0, x_4=0\)

\(x_1=3, x_2=0, x_3=0, x_4=1\)
Can any LP be made canonical?

\[
\begin{align*}
\text{max } z &= 0x_1 - \frac{1}{4}x_2 - x_3 + 0x_4 + 21 \\
\text{s.t. } &x_1 - \frac{3}{4}x_2 + 3x_3 = 3 \\
&\frac{1}{4}x_2 - 2x_3 + x_4 = 1 \\
x_j &\geq 0
\end{align*}
\]

(1) maximization, (2) positive RHS, (3) equality constraints, (4) non-negative vars, (5) isolated vars.

\textit{+1 coeff, only in one constraint, not in obj.}

Reduction to canonical form (I)

- “min z” =
- If a RHS value is negative then

- If \(x_1 \leq 0\) then

- If \(x_3\) is “free” (neither \(x_3 \leq 0\) or \(x_3 \geq 0\)) then
Reduction to canonical form (I)

• “min z” = “max −z”
• If a RHS value is negative then

• If \( x_1 \leq 0 \) then

• If \( x_3 \) is “free” (neither \( x_3 \leq 0 \) or \( x_3 \geq 0 \)) then
Reduction to canonical form (I)

• “min z” = “max –z”
• If a RHS value is negative then multiply constraint by -1

• If \( x_1 \leq 0 \) then replace \( x_1 := -x_2 \), with \( x_2 \geq 0 \)

• If \( x_3 \) is “free” (neither \( x_3 \leq 0 \) or \( x_3 \geq 0 \)) then

Reduction to canonical form (I)

• “min z” = “max –z”
• If a RHS value is negative then multiply constraint by -1

• If \( x_1 \leq 0 \) then replace \( x_1 := -x_2 \), with \( x_2 \geq 0 \)

• If \( x_3 \) is “free” (neither \( x_3 \leq 0 \) or \( x_3 \geq 0 \)) then replace \( x_3 := u – v \), with \( u \geq 0 \) and \( v \geq 0 \).
Reduction to canonical form (II)

- Inequality constraints

\[
\begin{align*}
40x_1 + 10x_2 + 6x_3 & \leq 55 \\
40x_1 + 10x_2 + 6x_3 & \geq 33
\end{align*}
\]

\[x_4 \geq 0, \ x_5 \geq 0\]

slack variable

surplus variable
Reduction to canonical form (III)

- Need isolated variables
- A constraint with slack var already good!

\[ 40x_1 + 10x_2 + 6x_3 + x_4 = 55 \]

- Other constraints, e.g. with surplus vars not good:

\[ 40x_1 + 10x_2 + 6x_3 - x_5 = 33 \]

doesn’t work
Reduction to canonical form (III)

- Need isolated variables
- A constraint with slack var already good!

\[ 40x_1 + 10x_2 + 6x_3 + x_4 = 55 \]

- Other constraints, e.g. with surplus vars not good:
  \[ 40x_1 + 10x_2 + 6x_3 - x_5 = 33 \]

Introduce a new artificial variable (we’ll insist that \( x_6 = 0 \) in any solution)

\[ 40x_1 + 10x_2 + 6x_3 - x_5 + x_6 = 33 \]

Standard Inequality Form

\[
\begin{align*}
\text{max} & \quad c^T x \\
\text{s.t.} & \quad Ax \leq b \\
& \quad x \geq 0 \\
\end{align*}
\]

\( c^T = (c_1, \ldots, c_n) \)

\( A = \begin{pmatrix}
  a_{11} & \cdots & a_{1n} \\
  \vdots & \ddots & \vdots \\
  a_{m1} & \cdots & a_{mn}
\end{pmatrix} \quad x = \begin{pmatrix}
  x_1 \\
  \vdots \\
  x_n
\end{pmatrix} \quad b = \begin{pmatrix}
  b_1 \\
  \vdots \\
  b_m
\end{pmatrix} \)
Review: Matrices (1/4)

• Matrix: rectangular array of numbers \([a_{ij}]\)
  
  – dimension: \(m \times n\) (\(m\) rows, \(n\) columns)
  
  – \(k \times 1\): column vector; \(1 \times k\): row vector

• \(B = \alpha A = A\alpha\), scalar \(\alpha\): \(\alpha a_{ij} = b_{ij}\)

\[
A \cdot B = C \\
\begin{pmatrix} (m \times p) \\ (p \times n) \end{pmatrix} \begin{pmatrix} (m \times n) \end{pmatrix} = \begin{pmatrix} -7 & -17 \\ -1 & -10 \end{pmatrix}
\]
Review: Matrices (2/4)

- $\mathbf{A}^T$ transpose: $a_{ij}^{T} = a_{ji}$

  $$A = \begin{pmatrix} 2 & 4 & -1 \\ -3 & 0 & 44 \end{pmatrix}, \quad A^T = \begin{pmatrix} 2 & -3 \\ 4 & 0 \\ -1 & 4 \end{pmatrix}$$

- $\mathbf{c}^T \cdot \mathbf{x} = \sum_{j=1}^{n} c_j x_j$ “inner product”

- Partitions

  $$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} = \begin{pmatrix} A_1 & \cdots & A_n \end{pmatrix}$$

  $$Ax = A_1 x_1 + \ldots + A_n x_n$$
Review: Matrices (3/4)

- **Square** matrix: \( m \) by \( m \)
- **Identity** matrix: square matrix w/ diagonal elements all 1 and all non-diagonal are 0.
- \( I_2, I_3, \ldots \)

\[
I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}
\]

- \( m \) by \( m \) square \( A \), inverse: \( A^{-1} = B \Rightarrow BA = AB = I_m \)

\[
\begin{pmatrix} 2 & 0 & -1 \\ 3 & 1 & 2 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 \\ -5 & 1 & -7 \\ 1 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}
\]

Review: Matrices (4/4)

- Given \( Ax = b \) (with square matrix \( A \))
- Can write:

\[
A^{-1}(Ax) = A^{-1}b
\]

Equivalently:

\[
x = A^{-1}b
\]

- Can find a unique solution to a square linear system if \( A \) is invertible.
Next Time

• Applications, Examples, Exercises.