Agenda

We've done

- Greedy Method
- Divide and Conquer
- Dynamic Programming
- Network Flows & Applications
- $\mathbf{NP}$-completeness

Now

- Linear Programming and the Simplex Method

Linear Programming Motivation: The Diet Problem

Setting

- n foods (beef, apple, potato chips, pho, bún bò, etc.)
- m nutritional elements (vitamins, calories, etc.)
- each gram of jth food contains a_{ij} units of nutritional element i
- a good meal needs b_i units of nutritional element i
- each gram of jth food costs c_j

Objective

- design the most economical meal yet dietarily sufficient
- (Halliburton must solve this problem!)

The Diet Problem as a Linear Program

Let x_j be the weight of food j in a dietarily sufficient meal.

$$
\begin{array}{ll}
\min & c_1x_1 + c_2x_2 + \cdots + c_nx_n \\
\text{subject to} & \begin{array}{ccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & \geq & b_1 \\
a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & \geq & b_2 \\
\vdots & & \vdots & & \dots & & \vdots & \vdots & \vdots \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & \geq & b_m \\
 & & & & & & \multicolumn{3}{r}{x_j \geq 0, \forall j = 1, \dots, n,}
\end{array}
\end{array}
$$

Linear Programming Motivation: The Max-Flow Problem

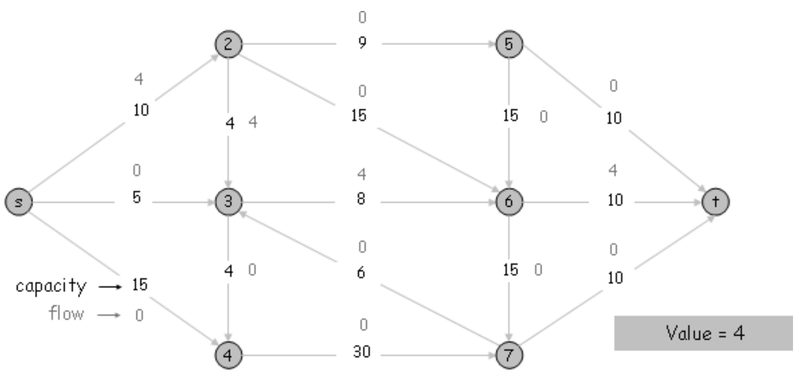
Maximize the value of f:

$$\text{val}(f) = \sum_{e=(s,v)\in E} f_e$$

Subject to

$$0 \leq f_e \leq c_e, \quad \forall e \in E$$

$$\sum_{e=(u,v)\in E} f_e - \sum_{e=(v,w)\in E} f_e = 0, \quad \forall v \neq s, t$$



Formalizing the Linear Programming Problem

Linear objective function

$$\max \quad \text{or} \quad \min -\frac{8}{3}x_1 + 2x_2 + x_3 - 6x_4 + x_5$$

Linear constraints, can take many forms

- Inequality constraints

$$\begin{aligned} 3x_1 + 4x_5 - 2x_6 &\geq 3 \\ 2x_1 + 2x_2 + x_3 &\leq 0 \end{aligned}$$

- Equality constraints

$$-x_2 - x_4 + x_3 = -3$$

- Non-negativity constraints (special case of inequality)

$$x_1, x_5, x_7 \geq 0$$

Some notational conventions

All vectors are column vectors

$$\mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \dots & \dots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}.$$

Linear Program: Standard Form

$$
\begin{array}{ll}
\min / \max & c_1x_1 + c_2x_2 + \cdots + c_nx_n \\
\text{subject to} & \begin{array}{ccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & = & b_1 \\
a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & = & b_2 \\
\vdots & & \vdots & & \dots & & \vdots & = & \vdots \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & = & b_m
\end{array} \\
& \hfill x_j \geq 0, \forall j = 1, \dots, n,
\end{array}
$$

or, in matrix notations,

$$\min / \max \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}$$

Linear Program: Canonical Form – $\min$ Version

$$
\begin{array}{ll}
\min & c_1x_1 + c_2x_2 + \cdots + c_nx_n \\
\text{subject to} & \begin{array}{ccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & \geq & b_1 \\
a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & \geq & b_2 \\
\vdots & & \vdots & & \dots & & \vdots & \geq & \vdots \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & \geq & b_m
\end{array} \\
& \hfill x_j \geq 0, \forall j = 1, \dots, n,
\end{array}
$$

or, in matrix notations,

$$\min \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}$$

Linear Program: Canonical Form – $\max$ Version

$$
\begin{array}{lccccccccc}
\max & c_1x_1 + c_2x_2 + \cdots + c_nx_n & & & & & & & & \\
\text{subject to} & a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & \leq & b_1 \\
 & a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & \leq & b_2 \\
 & \vdots & & \vdots & & \dots & & \vdots & \leq & \vdots \\
 & a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & \leq & b_m \\
 & & & & & & & \multicolumn{3}{r}{x_j \geq 0, \forall j = 1, \dots, n,}
\end{array}
$$

or, in matrix notations,

$$\max \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}$$

Conversions Between Forms of Linear Programs

- $\max \mathbf{c}^T\mathbf{x} = \min(-\mathbf{c})^T\mathbf{x}$
- $\sum_j a_{ij}x_j = b_i$ is equivalent to $\sum_j a_{ij}x_j \leq b_i$ and $\sum_j a_{ij}x_j \geq b_i$.
- $\sum_j a_{ij}x_j \leq b_i$ is equivalent to $-\sum_j a_{ij}x_j \geq -b_i$
- $\sum_j a_{ij}x_j \leq b_i$ is equivalent to $\sum_j a_{ij}x_j + s_i = b_i, s_i \geq 0$. The variable s_i is called a *slack variable*.
- When $x_j \leq 0$, replace all occurrences of x_j by $-x'_j$, and replace $x_j \leq 0$ by $x'_j \geq 0$.
- When x_j is not restricted in sign, replace it by $(u_j - v_j)$, and $u_j, v_j \geq 0$.

Example of Converting Linear Programs

Write

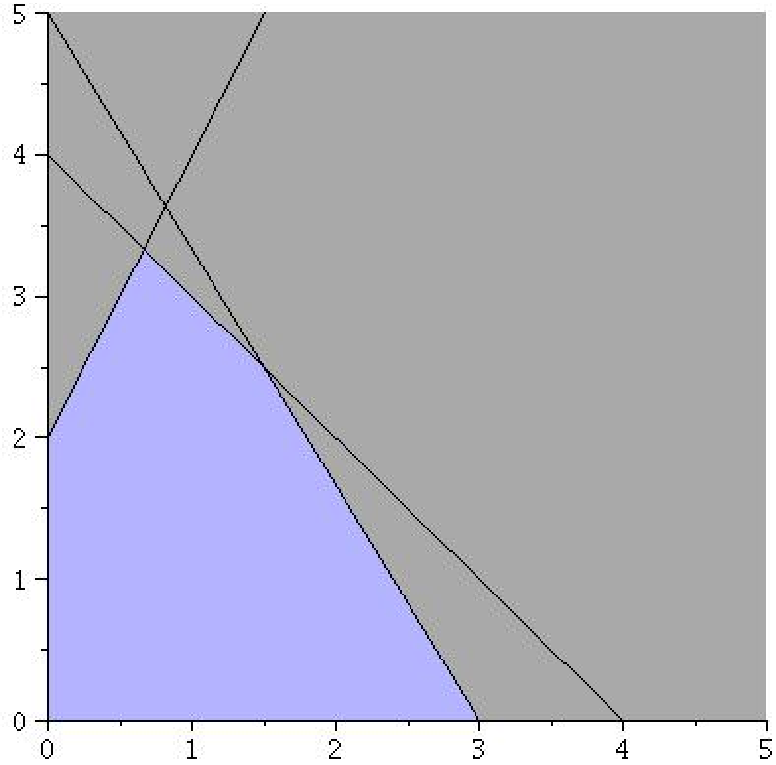
$$
\begin{array}{rrcrcrcrcr}
\min & x_1 & - & x_2 & + & 4x_3 & & & & \\
\text{subject to} & 3x_1 & - & x_2 & & & & & = & 3 \\
 & & - & x_2 & & & + & 2x_4 & \geq & 4 \\
 & x_1 & & & + & x_3 & & & \leq & -3 \\
 & & & & & & & x_1, x_2 & \geq & 0
\end{array}
$$

in standard $(\min / \max)$ form and canonical $(\min / \max)$ form.

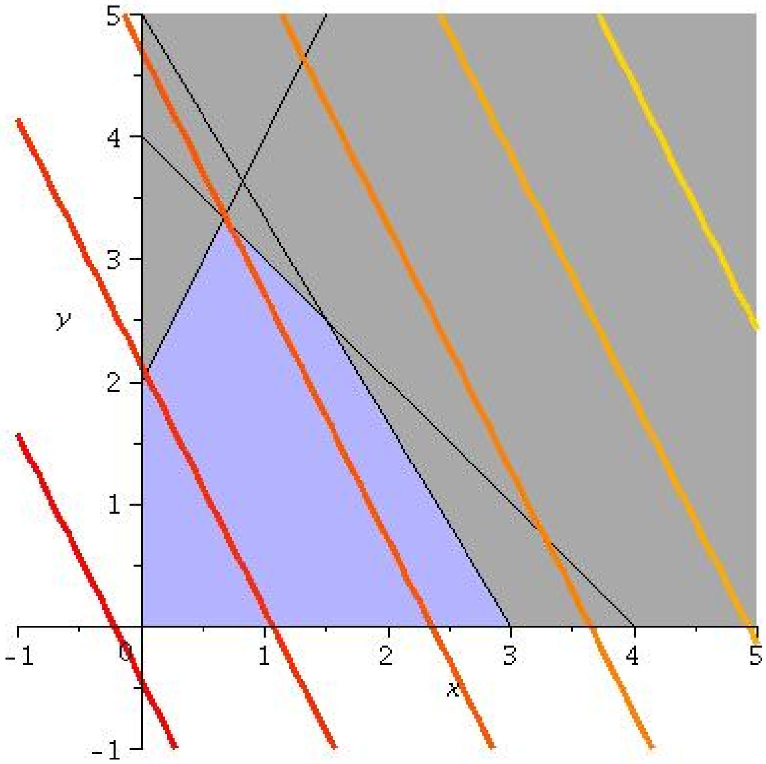
LP Geometry: Example 1

$$
\begin{array}{rrcrcr}
\max & 2x + y & & & & \\
\text{subject to} & -2x & + & y & \leq & 2 \\
 & 5x & + & 3y & \leq & 15 \\
 & x & + & y & \leq & 4 \\
 & & & & & x \geq 0, y \geq 0
\end{array}
$$

Example 1 – Feasible Region



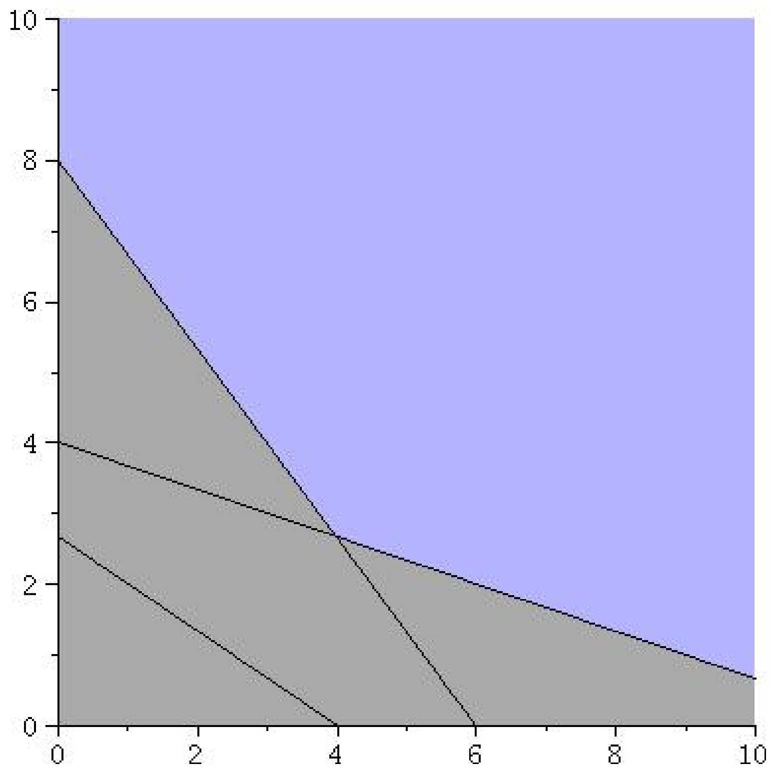
Example 1 – Objective Function



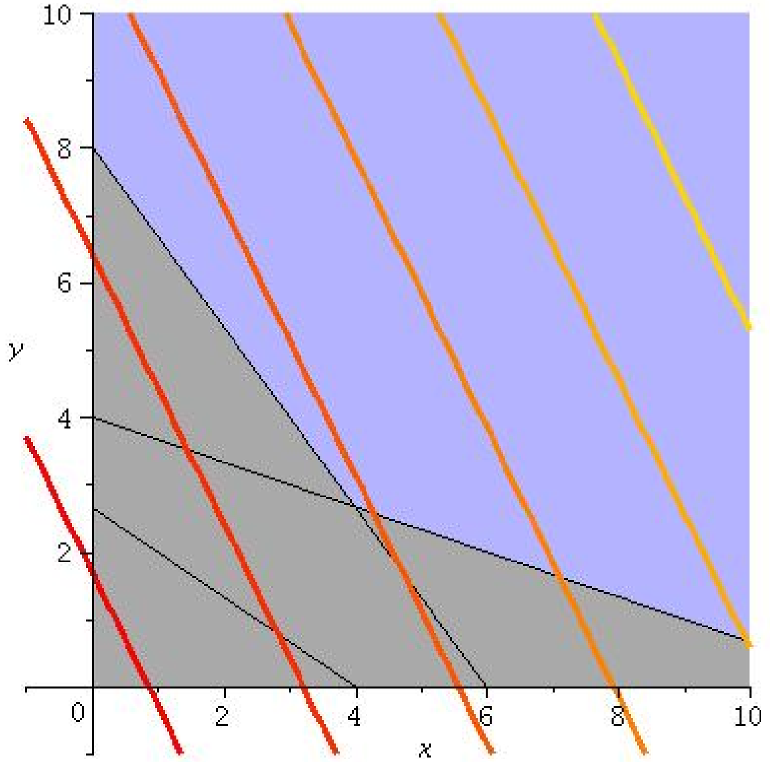
LP Geometry: Example 2

$$
\begin{array}{rcccccc}
\max & 2x + y & & & & \\
\text{subject to} & 2x & + & 3y & \geq & 8 \\
& 8x & + & 3y & \geq & 12 \\
& 4x & + & 3y & \geq & 24
\end{array}
$$

Example 2 – Feasible Region

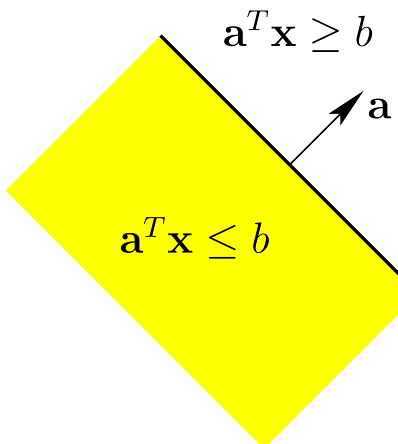


Example 2 – Objective Function



Half Space and Hyperplane

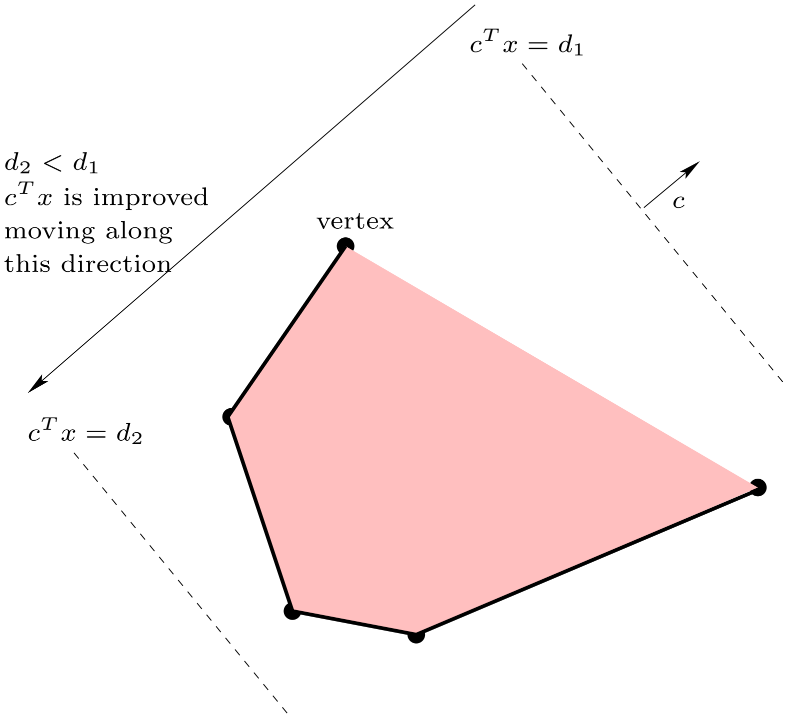
Each inequality $\mathbf{a}^T\mathbf{x} \geq b$ defines a half-space.



Each equality $\mathbf{a}^T\mathbf{x} = b$ defines a hyperplane.

Polyhedron, Vertices, Direction of Optimization

A polyhedron is the set of points $\mathbf{x}$ satisfying $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ (or equivalently $\mathbf{A}'\mathbf{x} \geq \mathbf{b}'$)



Linear Programming Duality: A Motivating Example

$$
\begin{array}{lrcrcrcl}
\max & 3x_1 & + & 2x_2 & + & 4x_3 & & \\
\text{subject to} & x_1 & + & x_2 & + & 2x_3 & \leq & 4 \\
 & 2x_1 & & & + & 3x_3 & \leq & 5 \\
 & 4x_1 & + & x_2 & + & 3x_3 & \leq & 7 \\
 & & & & & x_1, x_2, x_3 & \geq & 0.
\end{array}
$$

Someone claims $\mathbf{x}^T = \begin{bmatrix} 1 & 3 & 0 \end{bmatrix}$ is optimal with cost 9.
Feasibility is easy. How could we verify optimality?
We ask the person for a proof!

Her Proof of the Optimality of $\mathbf{x}$

$$
\begin{array}{llllllllll}
 & \frac{5}{3}\cdot(& x_1 & + & x_2 & + & 2x_3 &) & \leq & \frac{5}{3}\cdot 4 \\
 & 0\cdot(& 2x_1 & & & + & 3x_3 &) & \leq & 0\cdot 5 \\
 & \frac{1}{3}\cdot(& 4x_1 & + & x_2 & + & 3x_3 &) & \leq & \frac{1}{3}\cdot 7 \\
\ldots & & & & & & & & & \\
Ma & ia & hii & & Ma & ia & huu & & & \\
Ma & ia & hoo & & Ma & ia & haha & & & \\
\ldots & & & & & & & & & \\
\Rightarrow & & 3x_1 & + & 2x_2 & + & 4x_3 & & \leq & 9
\end{array}
$$

Done!

OK, How Did ... I do that?

Beside listening to Numa Numa, find non-negative multipliers

$$
\begin{array}{lllllllll}
y_1\cdot(& x_1 & + & x_2 & + & 2x_3 &) & \leq & 4y_1 \\
y_2\cdot(& 2x_1 & & & + & 3x_3 &) & \leq & 5y_2 \\
y_3\cdot(& 4x_1 & + & x_2 & + & 3x_3 &) & \leq & 7y_3
\end{array}
$$

$\Rightarrow (y_1+2y_2+4y_3)x_1+(y_1+y_3)x_2+(2y_1+3y_2+3y_3)x_3 \leq (4y_1+5y_2+7y_3)$

Want LHS to be like $3x_1+2x_2+4x_3$. Thus, as long as

$$
\begin{array}{lllllll}
y_1 & + & 2y_2 & + & 4y_3 & \geq & 3 \\
y_1 & & & + & y_3 & \geq & 2 \\
2y_1 & + & 3y_2 & + & 3y_3 & \geq & 4 \\
 & & & & y_1,y_2,y_3 & \geq & 0.
\end{array}
$$

we have

$$3x_1+2x_2+4x_3 \leq 4y_1+5y_2+7y_3$$

How to Get the Best Multipliers

Answer: minimize the upper bound.

$$\begin{array}{rrcrcrcr} \min & 4y_1 & + & 5y_2 & + & 7y_3 & & \\ & y_1 & + & 2y_2 & + & 4y_3 & \geq & 3 \\ & y_1 & & & + & y_3 & \geq & 2 \\ & 2y_1 & + & 3y_2 & + & 3y_3 & \geq & 4 \\ & & & & & y_1, y_2, y_3 & \geq & 0. \end{array}$$

What We Have Just Shown

If $\mathbf{x}$ is feasible for the Primal Program

$$\begin{array}{rrcrcrcr} \max & 3x_1 & + & 2x_2 & + & 4x_3 & & \\ \text{subject to} & x_1 & + & x_2 & + & 2x_3 & \leq & 4 \\ & 2x_1 & & & + & 3x_3 & \leq & 5 \\ & 4x_1 & + & x_2 & + & 3x_3 & \leq & 7 \\ & & & & & x_1, x_2, x_3 & \geq & 0. \end{array}$$

and $\mathbf{y}$ is feasible for the Dual Program

$$\begin{array}{rrcrcrcr} \min & 4y_1 & + & 5y_2 & + & 7y_3 & & \\ & y_1 & + & 2y_2 & + & 4y_3 & \geq & 3 \\ & y_1 & & & + & y_3 & \geq & 2 \\ & 2y_1 & + & 3y_2 & + & 3y_3 & \geq & 4 \\ & & & & & y_1, y_2, y_3 & \geq & 0. \end{array}$$

then

$$3x_1 + 2x_2 + 4x_3 \leq 4y_1 + 5y_2 + 7y_3$$

Primal-Dual Pairs - Canonical Form

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} & \text{(primal/dual program)} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \geq \mathbf{b} & \\ & \mathbf{x} \geq \mathbf{0} & \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbf{b}^T\mathbf{y} & \text{(dual/primal program)} \\ \text{subject to} \quad & \mathbf{A}^T\mathbf{y} \leq \mathbf{c} & \\ & \mathbf{y} \geq \mathbf{0}. & \end{aligned}$$

Note

The dual of the dual is the primal!

Primal-Dual Pairs - Standard Form

$$\begin{aligned} \min / \max \quad & \mathbf{c}^T\mathbf{x} & \text{(primal program)} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} & \\ & \mathbf{x} \geq \mathbf{0} & \end{aligned}$$

$$\begin{aligned} \max / \min \quad & \mathbf{b}^T\mathbf{y} & \text{(dual program)} \\ \text{subject to} \quad & \mathbf{A}^T\mathbf{y} \leq \mathbf{c} & \text{no non-negativity restriction!.} \end{aligned}$$

General Rules for Writing Dual Programs

Maximization problem	Minimization problem
Constraints	Variables
ith constraint $\leq$	ith variable ≥ 0
ith constraint $\geq$	ith variable ≤ 0
ith constraint $=$	ith variable unrestricted
Variables	Constraints
jth variable ≥ 0	jth constraint $\geq$
jth variable ≤ 0	jth constraint $\leq$
jth variable unrestricted	jth constraint $=$

Table: Rules for converting between primals and duals.

Note

The dual of the dual is the primal!

Weak Duality – Canonical Form Version

Consider the following primal-dual pair in canonical form

$$\begin{aligned} \text{Primal LP:} \quad & \min\{\mathbf{c}^\mathbf{T}\mathbf{x} \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}, \\ \text{Dual LP:} \quad & \max\{\mathbf{b}^\mathbf{T}\mathbf{y} \mid \mathbf{A}^\mathbf{T}\mathbf{y} \leq \mathbf{c}, \mathbf{y} \geq \mathbf{0}\}. \end{aligned}$$

Theorem (Weak Duality)

Suppose $\mathbf{x}$ is primal feasible, and $\mathbf{y}$ is dual feasible for the LPs defined above, then $\mathbf{c}^T\mathbf{x} \geq \mathbf{b}^T\mathbf{y}$.

Corollary

If $\mathbf{x}^*$ is an primal-optimal and $\mathbf{y}^*$ is an dual-optimal, then $\mathbf{c}^T\mathbf{x}^* \geq \mathbf{b}^T\mathbf{y}^*$.

Corollary

If $\mathbf{x}^*$ is primal-feasible, $\mathbf{y}^*$ is dual-feasible, and $\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*$, then $\mathbf{x}^*$ and $\mathbf{y}^*$ are optimal for their respective programs.

Weak Duality – Standard Form Version

Consider the following primal-dual pair in standard form

$$
\begin{aligned}
\text{Primal LP:} \quad & \min\{\mathbf{c}^{\mathbf{T}}\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}, \\
\text{Dual LP:} \quad & \max\{\mathbf{b}^{\mathbf{T}}\mathbf{y} \mid \mathbf{A}^{\mathbf{T}}\mathbf{y} \leq \mathbf{c}\}.
\end{aligned}
$$

Theorem (Weak Duality)

Suppose $\mathbf{x}$ is primal feasible, and $\mathbf{y}$ is dual feasible for the LPs defined above, then $\mathbf{c}^T\mathbf{x} \geq \mathbf{b}^T\mathbf{y}$.

Corollary

If $\mathbf{x}^*$ is an primal-optimal and $\mathbf{y}^*$ is an dual-optimal, then $\mathbf{c}^T\mathbf{x}^* \geq \mathbf{b}^T\mathbf{y}^*$.

Corollary

If $\mathbf{x}^*$ is primal-feasible, $\mathbf{y}^*$ is dual-feasible, and $\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*$, then $\mathbf{x}^*$ and $\mathbf{y}^*$ are optimal for their respective programs.

Strong Duality

Theorem (Strong Duality)

If the primal LP has an optimal solution $\mathbf{x}^*$, then the dual LP has an optimal solution $\mathbf{y}^*$ such that

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*.$$

			Dual		
			Feasible		Infeasible
			Optimal	Unbounded	
Primal	Feasible	Optimal	X	Nah	Nah
		Unbounded	Nah	Nah	X
	Infeasible		Nah	X	X

The Diet Problem Revisited

The dual program for the diet problem:

$$
\begin{array}{lccccccccc}
\max & b_1y_1 + b_2y_2 + \cdots + b_my_m \\
\text{subject to} & a_{11}y_1 & + & a_{21}y_2 & + & \dots & + & a_{m1}y_m & \geq & c_1 \\
& a_{12}y_1 & + & a_{22}y_2 & + & \dots & + & a_{2n}y_m & \geq & c_2 \\
& \vdots & & \vdots & & \dots & & \vdots & \vdots & \vdots \\
& a_{1n}y_1 & + & a_{2n}y_2 & + & \dots & + & a_{mn}y_m & \geq & c_n \\
& & & & & & & & & y_j \geq 0, \forall j = 1, \dots, m,
\end{array}
$$

(Possible) Interpretation: y_i is the price per unit of nutrient i that a whole-seller sets to "manufacture" different types of foods.

The Max-Flow Problem Revisited

The dual program for the Max-Flow LP Formulation:

$$
\begin{array}{lccl}
\min & \displaystyle\sum_{uv \in E} c_{uv}y_{uv} \\
\text{subject to} & y_{uv} - z_u + z_v & \geq & 0 \quad \forall uv \in E \\
& z_s & = & 1 \\
& z_t & = & 0 \\
& y_{uv} & \geq & 0 \quad \forall uv \in E
\end{array}
$$

Theorem (Max-Flow Min-Cut)

Maximum flow value equal minimum cut capacity.

Proof.

Let $(\mathbf{y}^*, \mathbf{z}^*)$ be optimal to the dual above. Set $W = \{v \mid z_v^* \geq 1\}$, then total flow out of W is equal to cap$(()W, \overline{W})$. □

The Simplex Method: High-Level Overview

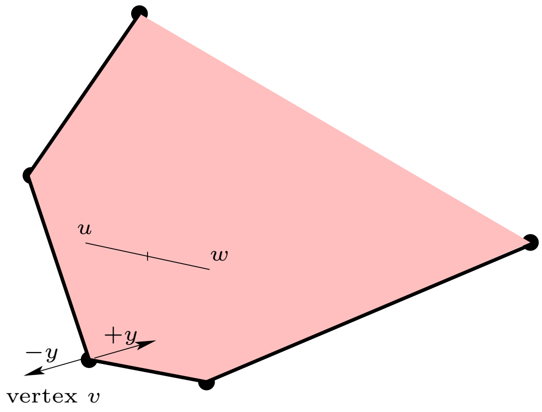
Consider a linear program $\min\{\mathbf{c}^T\mathbf{x} \mid \mathbf{x} \in P\}$, P is a polyhedron

1. Find a vertex of P, if P is not empty (the LP is feasible)
2. Find a neighboring vertex with better cost
   - If found, then repeat step 2
   - Otherwise, either report UNBOUNDED or OPTIMAL SOLUTION

Questions

1. When is P not empty?
2. When does P have a vertex? (i.e. P is pointed)
3. What is a vertex, anyhow?
4. How to find an initial vertex?
5. What if no vertex is optimal?
6. How to find a "better" neighboring vertex
7. Will the algorithm terminate?
8. How long does it take?

3. What is a vertex, anyhow?



Many ways to define a vertex v:

- $v \in P$ a vertex iff $\nexists \mathbf{y} \neq 0$ with $\mathbf{v} + \mathbf{y}, \mathbf{v} - \mathbf{y} \in P$
- $v \in P$ a vertex iff $\nexists \mathbf{u} \neq \mathbf{w}$ such that $\mathbf{v} = (\mathbf{u} + \mathbf{w})/\mathbf{2}$
- $v \in P$ a vertex iff it's the unique intersection of n independent faces

Questions

1. When is P not empty?
2. When does P have a vertex? (i.e. P is pointed)
3. ~~What is a vertex, anyhow?~~
4. How to find an initial vertex?
5. What if no vertex is optimal?
6. How to find a "better" neighboring vertex
7. Will the algorithm terminate?
8. How long does it take?

2. When is P pointed?

Question

Define a polyhedron which has no vertex?

Lemma

P is pointed iff it contains no line

Lemma

$P = \{\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$, *if not empty, always has a vertex.*

Lemma

$\mathbf{v} \in P = \{\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$ *is a vertex iff the columns of* $\mathbf{A}$ *corresponding to non-zero coordinates of* $\mathbf{v}$ *are linearly independent*

Questions

1. When is P not empty?
2. ~~When does P have a vertex? (i.e. P is pointed)~~
3. ~~What is a vertex, anyhow?~~
4. How to find an initial vertex?
5. What if no vertex is optimal?
6. How to find a "better" neighboring vertex
7. Will the algorithm terminate?
8. How long does it take?

5. What if no vertex is optimal?

Lemma

Let $P = \{\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$. If $\min\left\{\mathbf{c}^{\mathbf{T}}\mathbf{x} \mid \mathbf{x} \in P\right\}$ is bounded (i.e. it has an optimal solution), then for all $\mathbf{x} \in P$, there is a vertex $\mathbf{v} \in P$ such that $\mathbf{c}^T\mathbf{v} \leq \mathbf{c}^T\mathbf{x}$.

Theorem

The linear program $\min\{\mathbf{c}^{\mathbf{T}}\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$ either

1. *is infeasible,*
2. *is unbounded, or*
3. *has an optimal solution at a vertex.*

Questions

1. When is P not empty?
2. ~~When does P have a vertex? (i.e. P is pointed)~~
3. ~~What is a vertex, anyhow?~~
4. How to find an initial vertex?
5. ~~What if no vertex is optimal?~~
6. How to find a “better” neighboring vertex
7. Will the algorithm terminate?
8. How long does it take?

6. How to find a "better" neighboring vertex

- The answer is the core of the Simplex method
- This is basically one iteration of the method

Consider a concrete example:

$$\begin{array}{lrcrcrcr} \max & 3x_1 & + & 2x_2 & + & 4x_3 & & \\ \text{subject to} & x_1 & + & x_2 & + & 2x_3 & \leq & 4 \\ & 2x_1 & & & + & 3x_3 & \leq & 5 \\ & 4x_1 & + & x_2 & + & 3x_3 & \leq & 7 \\ & & & & & x_1, x_2, x_3 & \geq & 0. \end{array}$$

Sample execution of the Simplex algorithm

Converting to standard form

$$\begin{array}{lcccccccc} \max & 3x_1 & +2x_2 & +4x_3 & & & & & \\ \text{subject to} & x_1 & +x_2 & +2x_3 & +x_4 & & & = & 4 \\ & 2x_1 & & +3x_3 & & +x_5 & & = & 5 \\ & 4x_1 & +x_2 & +3x_3 & & & +x_6 & = & 7 \\ & & & & & & x_1, x_2, x_3, x_4, x_5, x_6 & \geq & 0. \end{array}$$

- $\mathbf{x} = \begin{bmatrix} 0 & 0 & 0 & 4 & 5 & 7 \end{bmatrix}^T$ is a vertex!
- Define $B = \{4, 5, 6\}$, $N = \{1, 2, 3\}$.
- The variables x_i, $i \in N$ are called free variables.
- The x_i with $i \in B$ are basic variables.
- How does one improve $\mathbf{x}$? Increase x_3 as much as possible! (x_1 or x_2 works too.)

Sample execution of the Simplex algorithm

- x_3 can only be at most $5/3$, forcing $x_4 = 2/3, x_5 = 0, x_5 = 2$
- $\mathbf{x}^T = \begin{bmatrix} 0 & 0 & 5/3 & 2/3 & 0 & 2 \end{bmatrix}$ is the new vertex (why?!!!)
- The new objective value is $20/3$
- x_3 enters the basis B, x_5 leaves the basis
- $B = \{3, 4, 6\}$, $N = \{1, 2, 5\}$

Rewrite the linear program

$$\begin{array}{lrrrrrrcl}
\max & 3x_1 & +2x_2 & +4x_3 & & & & & \\
\text{subject to} & -\frac{1}{3}x_1 & +x_2 & & +\mathbf{x_4} & & & = & \frac{2}{3} \\
& \frac{2}{3}x_1 & & +\mathbf{x_3} & & +\frac{1}{3}x_5 & & = & \frac{5}{3} \\
& 2x_1 & +x_2 & & & & +\mathbf{x_6} & = & 2 \\
& & & & & & x_1, x_2, x_3, x_4, x_5, x_6 & \geq & 0.
\end{array}$$

Sample execution of the Simplex algorithm

We also want the objective function to depend only on the free variables:

$$\begin{aligned}
3x_1 + 2x_2 + 4x_3 &= 3x_1 + 2x_2 + 4\left(\frac{5}{3} - \frac{2}{3}x_1 - \frac{1}{3}x_5\right) \\
&= \frac{1}{3}x_1 + 2x_2 - \frac{4}{3}x_5 + \mathbf{\frac{20}{3}}
\end{aligned}$$

The linear program is thus equivalent to

$$\begin{array}{lrrrrrrcl}
\max & \frac{1}{3}x_1 & +2x_2 & & & -\frac{4}{3}x_5 & & + & \mathbf{\frac{20}{3}} \\
\text{subject to} & -\frac{1}{3}x_1 & +x_2 & & +\mathbf{x_4} & & & = & \frac{2}{3} \\
& \frac{2}{3}x_1 & & +\mathbf{x_3} & & +\frac{1}{3}x_5 & & = & \frac{5}{3} \\
& 2x_1 & +x_2 & & & & +\mathbf{x_6} & = & 2 \\
& & & & & & x_1, x_2, x_3, x_4, x_5, x_6 & \geq & 0.
\end{array}$$

Increase x_2 to $2/3$, so that x_2 enters, x_4 leaves.

Sample execution of the Simplex algorithm

$$\begin{array}{lrrrrrrcl}
\max & x_1 & & & -2x_4 & & & + & \mathbf{8} \\
\text{subject to} & -\frac{1}{3}x_1 & +\mathbf{x_2} & & +x_4 & & & = & \frac{2}{3} \\
& \frac{2}{3}x_1 & & +\mathbf{x_3} & & +\frac{1}{3}x_5 & & = & \frac{5}{3} \\
& \frac{7}{3}x_1 & & & -x_4 & -\frac{1}{3}x_5 & +\mathbf{x_6} & = & \frac{4}{3} \\
& & & & & & x_1,x_2,x_3,x_4,x_5,x_6 & \geq & 0.
\end{array}$$

At this point, only x_1 to increase.

- If all its coefficients are non-positive (like $-1/3$ above), then the LP is UNBOUNDED
- Fortunately, this is not the case here
- Increase x_1 to $4/7$, so that x_1 enters, x_6 leaves.

Sample execution of the Simplex algorithm

$$\begin{array}{lrrrrrrcl}
\max & & & & -\frac{11}{7}x_4 & +\frac{1}{7}x_5 & -\frac{3}{7}x_6 & + & \mathbf{\frac{60}{7}} \\
\text{subject to} & & +\mathbf{x_2} & & +\frac{6}{7}x_4 & -\frac{5}{7}x_5 & +\frac{1}{7}x_6 & = & \frac{6}{7} \\
& & & +\mathbf{x_3} & +\frac{2}{7}x_4 & +\frac{3}{7}x_5 & -\frac{2}{7}x_6 & = & \frac{9}{7} \\
& \mathbf{x_1} & & & -\frac{3}{7}x_4 & -\frac{1}{7}x_5 & +\frac{3}{7}x_6 & = & \frac{4}{7} \\
& & & & & & x_1,x_2,x_3,x_4,x_5,x_6 & \geq & 0.
\end{array}$$

Now, x_5 enters **again**, x_3 leaves.

Sample execution of the Simplex algorithm

$$
\begin{array}{lccccccc}
\max & & & -\frac{1}{3}x_3 & -\frac{34}{21}x_4 & & -\frac{1}{3}x_6 & + & \mathbf{9} \\
\text{subject to} & & +\mathbf{x_2} & +\frac{49}{15}x_3 & +\frac{188}{105}x_4 & & -\frac{1}{3}x_6 & = & 3 \\
 & & & +\frac{7}{3}x_3 & +\frac{2}{3}x_4 & +\mathbf{x_5} & -\frac{2}{3}x_6 & = & 3 \\
 & \mathbf{x_1} & & +\frac{1}{3}x_3 & -\frac{1}{3}x_4 & & +\frac{1}{3}x_6 & = & 1 \\
 & & & & & & x_1, x_2, x_3, x_4, x_5, x_6 & \geq & 0.
\end{array}
$$

Yeah! No more improvement is possible. We have reached the optimal vertex

$$\mathbf{v} = \begin{bmatrix} 1 & 3 & 0 & 0 & 3 & 0 \end{bmatrix}^T.$$

The optimal cost is 9.

Questions

1. When is P not empty?
2. ~~When does P have a vertex? (i.e. P is pointed)~~
3. ~~What is a vertex, anyhow?~~
4. How to find an initial vertex?
5. ~~What if no vertex is optimal?~~
6. ~~How to find a "better" neighboring vertex~~
7. Will the algorithm terminate?
8. How long does it take?

7&8 Termination and Running Time

Termination

- There are finitely many vertices ($\leq \binom{n}{m}$)
- Terminating = non-cycling, i.e. never come back to a vertex
- Many cycling prevention methods: *perturbation method, lexicographic rule, Bland's pivoting rule*, etc.
  - Bland's pivoting rule: pick smallest possible j to leave the basis, then smallest possible i to enter the basis

Running time

- Klee & Minty (1969) showed that Simplex could take exponential time

Summary: Simplex with Bland's Rule

1. Start from a vertex $\mathbf{v}$ of P.
2. Determine B and N; Let $\mathbf{y}_B^T = \mathbf{c}_B^T \mathbf{A}_B^{-1}$.
3. If $\left(c_N^T - \mathbf{y}_B^T \mathbf{a}_j\right) \geq 0$, then vertex $\mathbf{v}$ is optimal. Moreover,
$$\mathbf{c}^T\mathbf{v} = \mathbf{c}_B^T\mathbf{v}_B + \mathbf{c}_N^T\mathbf{v}_N = \mathbf{c}_B^T\left(\mathbf{A}_B^{-1}\mathbf{b} - \mathbf{A}_B^{-1}\mathbf{A}_N\mathbf{v}_N\right) + \mathbf{c}_N^T\mathbf{v}_N$$
4. Else, let
$$j = \min\left\{j' \in N : \left(c_{j'} - \mathbf{y}_B^T\mathbf{a}_{j'}\right) < 0\right\}.$$
5. If $\mathbf{A}_B^{-1}\mathbf{a}_j \leq 0$, then report unbounded LP and STOP!
6. Otherwise, pick smallest $k \in B$ such that $\left(\mathbf{A}_B^{-1}\mathbf{a}_j\right)_k > 0$ and that
$$\frac{(\mathbf{A}_B^{-1}\mathbf{b})_k}{\left(\mathbf{A}_B^{-1}\mathbf{a}_j\right)_k} = \min\left\{\frac{(\mathbf{A}_B^{-1}\mathbf{b})_i}{\left(\mathbf{A}_B^{-1}\mathbf{a}_j\right)_i} \;:\; i \in B,\;\; \left(\mathbf{A}_B^{-1}\mathbf{a}_j\right)_i > 0\right\}.$$
7. x_k leaves, x_j enters: $B = B \cup \{j\} - \{k\}$, $N = N \cup \{k\} - \{j\}$. GO BACK to step 3.

By Product: Strong Duality

Theorem (Strong Duality)

If the primal LP has an optimal solution $\mathbf{x}^*$, *then the dual LP has an optimal solution* $\mathbf{y}^*$ *such that*

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*.$$

Proof.

- Suppose Simplex returns vertex $\mathbf{x}^*$ (at B and N)
- Recall $\mathbf{y}_B^T = \mathbf{c}_B^T\mathbf{A}_B^{-1}$, then $\mathbf{c}^T\mathbf{x}^* = \mathbf{y}_B^T\mathbf{b}$

$$\mathbf{A}^T\mathbf{y}_B = \begin{bmatrix}\mathbf{A}_B^T \\ \mathbf{A}_N^T\end{bmatrix}\mathbf{y}_B = \begin{bmatrix}\mathbf{c}_B \\ \mathbf{A}_N^T\mathbf{y}_B\end{bmatrix} \leq \begin{bmatrix}\mathbf{c}_B \\ \mathbf{c}_N\end{bmatrix} = \mathbf{c}.$$

- Set $\mathbf{y}^* = \mathbf{y}_B^T$. Done!

Questions

1. When is P not empty?
2. ~~When does P have a vertex? (i.e. P is pointed)~~
3. ~~What is a vertex, anyhow?~~
4. How to find an initial vertex?
5. ~~What if no vertex is optimal?~~
6. ~~How to find a "better" neighboring vertex~~
7. ~~Will the algorithm terminate?~~
8. ~~How long does it take?~~

1&4 Feasibility and the Initial Vertex

- In $P = \{\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$, we can assume $\mathbf{b} \geq \mathbf{0}$ (why?).
- Let $\mathbf{A}' = \begin{bmatrix} \mathbf{A} & \mathbf{I} \end{bmatrix}$
- Let $P' = \{\mathbf{z} \mid \mathbf{A}'\mathbf{z} = \mathbf{b}, \mathbf{z} \geq \mathbf{0}\}$.
- A vertex of P' is $\mathbf{z} = [0, \dots, 0, b_1, \dots, b_m]^T$
- P is feasible iff the following LP has optimum value 0

$$\min \left\{ \sum_{i=1}^{m} z_{n+i} \mid \mathbf{z} \in P' \right\}$$

- From an optimal vertex $\mathbf{z}^*$, ignore the last m coordinates to obtain a vertex of P

Questions

1. ~~When is P not empty?~~
2. ~~When does P have a vertex? (i.e. P is pointed)~~
3. ~~What is a vertex, anyhow?~~
4. ~~How to find an initial vertex?~~
5. ~~What if no vertex is optimal?~~
6. ~~How to find a "better" neighboring vertex~~
7. ~~Will the algorithm terminate?~~
8. ~~How long does it take?~~