CME 307 / MS&E 311 / OIT 676: Optimization

LP geometry, modeling and solution techniques

Professor Udell

Management Science and Engineering
Stanford

November 18, 2024

Course survey

you're interested in:

- modeling real-world problems, from political science and economics to energy and desalination!
- robustness and modeling under uncertainty
- understanding core optimization concepts like duality and KKT conditions
- ...

questions:

- recommended resource for linear algebra?
- how to ask questions in class?

requests:

- slower on proofs, please!

Outline

LP standard form

LP inequality form

What kinds of points can be optimal?

Solving LPs

Modeling

Linear programming: standard form

standard form linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

optimal value $p^\star$, solution $x^\star$ (if it exists)

- any x with $Ax = b$ and $x \geq 0$ is called a **feasible point**
- if problem is infeasible, we say $p^\star = \infty$
- $p^\star$ can be finite or $\pm\infty$

Linear programming: standard form

standard form linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

optimal value $p^\star$, solution $x^\star$ (if it exists)

- any x with $Ax = b$ and $x \geq 0$ is called a **feasible point**
- if problem is infeasible, we say $p^\star = \infty$
- $p^\star$ can be finite or $\pm\infty$

Q: if $p^\star = -\infty$, does a solution exist?

Linear programming: standard form

standard form linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

optimal value $p^\star$, solution $x^\star$ (if it exists)

- any x with $Ax = b$ and $x \geq 0$ is called a **feasible point**
- if problem is infeasible, we say $p^\star = \infty$
- $p^\star$ can be finite or $\pm\infty$

Q: if $p^\star = -\infty$, does a solution exist?
Q: if $p^\star = \infty$, does a solution exist?

Linear programming: standard form

standard form linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

optimal value $p^\star$, solution $x^\star$ (if it exists)

- any x with $Ax = b$ and $x \geq 0$ is called a **feasible point**
- if problem is infeasible, we say $p^\star = \infty$
- $p^\star$ can be finite or $\pm\infty$

Q: if $p^\star = -\infty$, does a solution exist?
Q: if $p^\star = \infty$, does a solution exist?
henceforth assume $A \in \mathbf{R}^{m \times n}$ has full row rank m
Q: why? how to check?

Linear programming: standard form

standard form linear program (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

optimal value $p^\star$, solution $x^\star$ (if it exists)

- any x with $Ax = b$ and $x \geq 0$ is called a **feasible point**
- if problem is infeasible, we say $p^\star = \infty$
- $p^\star$ can be finite or $\pm\infty$

Q: if $p^\star = -\infty$, does a solution exist?
Q: if $p^\star = \infty$, does a solution exist?
henceforth assume $A \in \mathbf{R}^{m \times n}$ has full row rank m
Q: why? how to check?
A: otherwise infeasible or redundant rows; use gaussian elimination to check and remove

LP example: diet problem

- x_j servings of food j, $j = 1, \ldots, n$
- c_j cost per serving
- a_{ij} amount of nutrient i in food j
- b_i required amount of nutrient i, $i = 1, \ldots, m$

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

LP example: diet problem

- x_j servings of food j, $j = 1, \ldots, n$
- c_j cost per serving
- a_{ij} amount of nutrient i in food j
- b_i required amount of nutrient i, $i = 1, \ldots, m$

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Ax = b \\
 & x \geq 0
\end{array}
$$

extensions:

- foods come from recipes? $x = By$

LP example: diet problem

- x_j servings of food j, $j = 1, \ldots, n$
- c_j cost per serving
- a_{ij} amount of nutrient i in food j
- b_i required amount of nutrient i, $i = 1, \ldots, m$

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

extensions:

- foods come from recipes? $x = By$
- ensure diversity in diet? $y \leq u$

LP example: diet problem

- x_j servings of food j, $j = 1, \ldots, n$
- c_j cost per serving
- a_{ij} amount of nutrient i in food j
- b_i required amount of nutrient i, $i = 1, \ldots, m$

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

extensions:

- foods come from recipes? $x = By$
- ensure diversity in diet? $y \leq u$
- ranges of nutrients? $Ax + s = b,\ l \leq s \leq u$

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.
interpretation: conic hull

- define the **cone** generated by $A = [a_1, \ldots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \textbf{cone}(a_1, \ldots, a_n)$$

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.
interpretation: conic hull

- define the **cone** generated by $A = [a_1, \dots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \mathbf{cone}(a_1, \dots, a_n)$$

- LP is feasible if $b \in \mathbf{cone}(a_1, \dots, a_n)$

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.
interpretation: conic hull

- define the **cone** generated by $A = [a_1, \dots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \mathbf{cone}(a_1, \dots, a_n)$$

- LP is feasible if $b \in \mathbf{cone}(a_1, \dots, a_n)$

interpretation: intersection of hyperplane and halfspaces

- define a **hyperplane** $\{x \mid Ax = b\}$

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.

interpretation: conic hull

- define the **cone** generated by $A = [a_1, \ldots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \textbf{cone}(a_1, \ldots, a_n)$$

- LP is feasible if $b \in \textbf{cone}(a_1, \ldots, a_n)$

interpretation: intersection of hyperplane and halfspaces

- define a **hyperplane** $\{x \mid Ax = b\}$ (dimension?)

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.

interpretation: conic hull

- define the **cone** generated by $A = [a_1, \dots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \mathbf{cone}(a_1, \dots, a_n)$$

- LP is feasible if $b \in \mathbf{cone}(a_1, \dots, a_n)$

interpretation: intersection of hyperplane and halfspaces

- define a **hyperplane** $\{x \mid Ax = b\}$ (dimension?)
- define a **halfspace** $\{x \mid a^T x \geq b\}$

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.
interpretation: conic hull

- define the **cone** generated by $A = [a_1, \ldots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \textbf{cone}(a_1, \ldots, a_n)$$

- LP is feasible if $b \in \textbf{cone}(a_1, \ldots, a_n)$

interpretation: intersection of hyperplane and halfspaces

- define a **hyperplane** $\{x \mid Ax = b\}$ (dimension?)
- define a **halfspace** $\{x \mid a^T x \geq b\}$
- the **positive orthant** $x \geq 0$ is an intersection of halfspaces

Geometry of LP

the **feasible set** is the set of points $\{x \mid Ax = b,\ x \geq 0\}$ that satisfy all constraints.

interpretation: conic hull

- define the **cone** generated by $A = [a_1, \ldots a_n]$:

$$\{Ax \mid x \geq 0\} = \left\{ \sum_{i=1}^{n} a_i x_i \mid x \geq 0 \right\} = \textbf{cone}(a_1, \ldots, a_n)$$

- LP is feasible if $b \in \textbf{cone}(a_1, \ldots, a_n)$

interpretation: intersection of hyperplane and halfspaces

- define a **hyperplane** $\{x \mid Ax = b\}$ (dimension?)
- define a **halfspace** $\{x \mid a^T x \geq b\}$
- the **positive orthant** $x \geq 0$ is an intersection of halfspaces
- LP is feasible if hyperplane $\{x \mid Ax = b\}$ intersects the positive orthant

Geometry of LP: convexity

- define **convex combination** of $x, y \in \mathbf{R}^n$: $\theta x + (1 - \theta) y$ for $\theta \in [0, 1]$

Geometry of LP: convexity

- define **convex combination** of x, $y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta) y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta) y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

- define **polytope**: the convex hull of a finite set: $\mathbf{conv}(\{x_1, \dots, x_k\})$

some useful convex sets:

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

- define **polytope**: the convex hull of a finite set: $\mathbf{conv}(\{x_1, \ldots, x_k\})$

some useful convex sets:

- a hyperplane is convex

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

- define **polytope**: the convex hull of a finite set: $\mathbf{conv}(\{x_1, \ldots, x_k\})$

some useful convex sets:

- a hyperplane is convex
- a halfspace is convex

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

- define **polytope**: the convex hull of a finite set: $\mathbf{conv}(\{x_1, \ldots, x_k\})$

some useful convex sets:

- a hyperplane is convex
- a halfspace is convex
- the intersection of convex sets is convex

Geometry of LP: convexity

- define **convex combination** of $x,\ y \in \mathbf{R}^n$: $\theta x + (1-\theta)y$ for $\theta \in [0,1]$
- define **convex set**: C is convex if for any $x, y \in C$,

$$\theta x + (1-\theta)y \in C, \qquad \theta \in [0,1]$$

- define the **convex hull** of a set S:

$$\mathbf{conv}(S) = \left\{ \sum_{i=1}^{k} \theta_i x_i \mid x_i \in S,\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1 \right\}$$

- define **polytope**: the convex hull of a finite set: $\mathbf{conv}(\{x_1, \ldots, x_k\})$

some useful convex sets:

- a hyperplane is convex
- a halfspace is convex
- the intersection of convex sets is convex
- the feasible set $\{x : Ax = b,\ x \geq 0\}$ is convex

Outline

LP inequality form

another useful form for LP is **inequality form**

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array}$$

LP inequality form

another useful form for LP is **inequality form**

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array}$$

interpretation: halfspaces

- $a_i^T x \leq b_i$ defines a **halfspace**
- $Ax \leq b$ defines a **polyhedron**: intersection of halfspaces
- LP is feasible if polyhedron $\{x \mid Ax \leq b\}$ is nonempty

LP example: production planning

- x_i units of product i
- c_i cost per unit
- a_{ij} amount of resource j used by product i
- b_j amount of resource j available
- d_i demand for product i

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Ax \leq b \\
& 0 \leq x \leq d
\end{array}
$$

LP example: production planning

- x_i units of product i
- c_i cost per unit
- a_{ij} amount of resource j used by product i
- b_j amount of resource j available
- d_i demand for product i

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Ax \leq b \\
& 0 \leq x \leq d
\end{array}
$$

extensions:

- fixed cost for producing product i at all?

LP example: production planning

- x_i units of product i
- c_i cost per unit
- a_{ij} amount of resource j used by product i
- b_j amount of resource j available
- d_i demand for product i

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & 0 \leq x \leq d \end{array}$$

extensions:

- fixed cost for producing product i at all?
  $c^T x + f^T z$, $z_i \in \{0, 1\}$, $x_i \leq M z_i$ for M large

LP inequality form to standard form

standard form to inequality form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \rightarrow$$

LP inequality form to standard form

standard form to inequality form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \rightarrow \qquad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & Ax \geq b \\ & -x \leq 0 \end{array}$$

LP inequality form to standard form

standard form to inequality form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \quad \rightarrow \quad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & Ax \geq b \\ & -x \leq 0 \end{array}$$

inequality form to standard form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array} \quad \rightarrow$$

LP inequality form to standard form

standard form to inequality form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \quad \rightarrow \quad \begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & Ax \geq b \\ & -x \leq 0 \end{array}$$

inequality form to standard form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array} \quad \rightarrow \quad \begin{array}{ll} \text{minimize} & c^T (x_+ - x_-) \\ \text{subject to} & A(x_+ - x_-) + s = b \\ & s, x_+, x_- \geq 0 \end{array}$$

so both forms have the same expressive power, and feasible sets are polyhedra

Active constraints

for constraint set $Ax \leq b$, an **active constraint** at x is one that holds with equality:

$$a_i^T x = b_i$$

Active constraints

for constraint set $Ax \leq b$, an **active constraint** at x is one that holds with equality:

$$a_i^T x = b_i$$

- the **active set** at x is the set of indices of active constraints $\{i \mid a_i^T x = b_i\}$

Active constraints

for constraint set $Ax \leq b$, an **active constraint** at x is one that holds with equality:

$$a_i^T x = b_i$$

- the **active set** at x is the set of indices of active constraints $\{i \mid a_i^T x = b_i\}$

for nonnegative variable $x \geq 0$, x_i is **active** if $x_i > 0$

example: active slack variables are dual to active constraints

$$\begin{aligned} Ax \leq b \quad &\iff \quad Ax + s = b,\ s \geq 0 \\ a_i^T x = b_i \quad &\iff \quad s_i = 0 \\ \text{constraint } i \text{ is active} \quad &\iff \quad \text{slack variable } s_i \text{ is inactive} \end{aligned}$$

Outline

Extreme points

define **extreme point**: $x \in \mathbf{R}^n$ is extreme in $C \subset \mathbf{R}^n$ if it cannot be written as a convex combination of other points in C: for $\theta \in [0, 1]$,

$$x \in C \quad \text{and} \quad x = \theta y + (1 - \theta) z \quad \Longrightarrow \quad x = y = z$$

Extreme points

define **extreme point**: $x \in \mathbf{R}^n$ is extreme in $C \subset \mathbf{R}^n$ if it cannot be written as a convex combination of other points in C: for $\theta \in [0, 1]$,

$$x \in C \quad \text{and} \quad x = \theta y + (1-\theta) z \qquad \Longrightarrow \qquad x = y = z$$

fact: if $x^\star$ is the unique optimal solution of $\text{minimize}_{x \in S}\ c^T x$, then $x^\star$ is extreme in the set S.

Extreme points

define **extreme point**: $x \in \mathbf{R}^n$ is extreme in $C \subset \mathbf{R}^n$ if it cannot be written as a convex combination of other points in C: for $\theta \in [0, 1]$,

$$x \in C \quad \text{and} \quad x = \theta y + (1-\theta) z \quad \implies \quad x = y = z$$

fact: if $x^\star$ is the unique optimal solution of $\text{minimize}_{x \in S}\ c^T x$,
then $x^\star$ is extreme in the set S.
proof: suppose by way of contradiction that $x^\star$ is not extreme in S:

$$\begin{aligned} x^\star &= \theta y + (1-\theta) z \qquad \text{for } y, z \in S, \theta \in (0,1) \\ p^\star := c^T x^\star &= \theta c^T y + (1-\theta) c^T z > \theta p^\star + (1-\theta) p^\star = p^\star \end{aligned}$$

where the inequality follows from the (unique) optimality of $x^\star$. Contradiction!

Extreme points

define **extreme point**: $x \in \mathbf{R}^n$ is extreme in $C \subset \mathbf{R}^n$ if it cannot be written as a convex combination of other points in C: for $\theta \in [0, 1]$,

$$x \in C \quad \text{and} \quad x = \theta y + (1-\theta)z \quad \Longrightarrow \quad x = y = z$$

fact: if $x^\star$ is the unique optimal solution of $\text{minimize}_{x \in S}\ c^T x$,
then $x^\star$ is extreme in the set S.
proof: suppose by way of contradiction that $x^\star$ is not extreme in S:

$$\begin{aligned} x^\star &= \theta y + (1-\theta)z \qquad \text{for } y, z \in S, \theta \in (0,1) \\ p^\star := c^T x^\star &= \theta c^T y + (1-\theta) c^T z > \theta p^\star + (1-\theta) p^\star = p^\star \end{aligned}$$

where the inequality follows from the (unique) optimality of $x^\star$. Contradiction!

Q: Example of a problem with a non-extreme solution?

Extreme points

define **extreme point**: $x \in \mathbf{R}^n$ is extreme in $C \subset \mathbf{R}^n$ if it cannot be written as a convex combination of other points in C: for $\theta \in [0, 1]$,

$$x \in C \quad \text{and} \quad x = \theta y + (1 - \theta) z \quad \Longrightarrow \quad x = y = z$$

fact: if $x^\star$ is the unique optimal solution of $\text{minimize}_{x \in S} \; c^T x$,
then $x^\star$ is extreme in the set S.
proof: suppose by way of contradiction that $x^\star$ is not extreme in S:

$$\begin{aligned} x^\star &= \theta y + (1 - \theta) z \qquad \text{for } y, z \in S, \theta \in (0, 1) \\ p^\star := c^T x^\star &= \theta c^T y + (1 - \theta) c^T z > \theta p^\star + (1 - \theta) p^\star = p^\star \end{aligned}$$

where the inequality follows from the (unique) optimality of $x^\star$. Contradiction!

Q: Example of a problem with a non-extreme solution?
Q: Does there always exist an extreme solution?

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

interpretation: $\{z : c^T z = c^T x\}$ is a hyperplane that intersects S only at x.
we say this hyperplane **supports** S at x

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

interpretation: $\{z : c^T z = c^T x\}$ is a hyperplane that intersects S only at x.
we say this hyperplane **supports** S at x

fact: x is a vertex of $S \implies x$ is an extreme point of S

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

interpretation: $\{z : c^T z = c^T x\}$ is a hyperplane that intersects S only at x.
we say this hyperplane **supports** S at x

fact: x is a vertex of $S \implies x$ is an extreme point of S
proof:

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

interpretation: $\{z : c^T z = c^T x\}$ is a hyperplane that intersects S only at x.
we say this hyperplane **supports** S at x

fact: x is a vertex of $S \implies x$ is an extreme point of S
proof: x is a vertex of S. suppose its defining vector is c and consider the optimization problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x \in S \end{array}$$

Vertices

define **vertex**: $x \in \mathbf{R}^n$ is a vertex of set $S \subset \mathbf{R}^n$ if for some vector $c \in \mathbf{R}^n$,

$$c^T x < c^T y \quad \forall y \in S \setminus \{x\}$$

interpretation: $\{z : c^T z = c^T x\}$ is a hyperplane that intersects S only at x.
we say this hyperplane **supports** S at x

fact: x is a vertex of $S \implies x$ is an extreme point of S
proof: x is a vertex of S. suppose its defining vector is c and consider the optimization problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x \in S \end{array}$$

x is the unique optimum of this problem, so the proof of this statement follows from the previous proof.

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \text{(LP)}$$

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \text{(LP)}$$

define: $x \in \mathbf{R}^n$ is a **basic feasible solution** (BFS) of (LP) if there is a set $S \subset \{1, \ldots, n\}$ of m columns so that $A_S \in \mathbf{R}^{m \times m}$ is invertible and

$$x_S = A_S^{-1} b, \qquad x_{\bar{S}} = 0, \qquad x \geq 0.$$

- $A_S \in \mathbf{R}^{m \times m}$ is submatrix of A with columns in S

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \text{(LP)}$$

define: $x \in \mathbf{R}^n$ is a **basic feasible solution** (BFS) of (LP) if there is a set $S \subset \{1, \ldots, n\}$ of m columns so that $A_S \in \mathbf{R}^{m \times m}$ is invertible and

$$x_S = A_S^{-1} b, \qquad x_{\bar{S}} = 0, \qquad x \geq 0.$$

- $A_S \in \mathbf{R}^{m \times m}$ is submatrix of A with columns in S
- two BFS with S, S' are neighbors if they share all but one columns: $|S \cap S'| = m - 1$

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \qquad \text{(LP)}$$

define: $x \in \mathbf{R}^n$ is a **basic feasible solution** (BFS) of (LP) if there is a set $S \subset \{1, \ldots, n\}$ of m columns so that $A_S \in \mathbf{R}^{m \times m}$ is invertible and

$$x_S = A_S^{-1} b, \qquad x_{\bar{S}} = 0, \qquad x \geq 0.$$

- $A_S \in \mathbf{R}^{m \times m}$ is submatrix of A with columns in S
- two BFS with S, S' are neighbors if they share all but one columns: $|S \cap S'| = m - 1$

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \tag{LP}$$

define: $x \in \mathbf{R}^n$ is a **basic feasible solution** (BFS) of (LP) if there is a set $S \subset \{1, \ldots, n\}$ of m columns so that $A_S \in \mathbf{R}^{m \times m}$ is invertible and

$$x_S = A_S^{-1} b, \qquad x_{\bar{S}} = 0, \qquad x \geq 0.$$

- $A_S \in \mathbf{R}^{m \times m}$ is submatrix of A with columns in S
- two BFS with S, S' are neighbors if they share all but one columns: $|S \cap S'| = m - 1$

Q: how to find a BFS?

Basic feasible solution

recall the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array} \tag{LP}$$

define: $x \in \mathbf{R}^n$ is a **basic feasible solution** (BFS) of (LP) if there is a set $S \subset \{1, \ldots, n\}$ of m columns so that $A_S \in \mathbf{R}^{m \times m}$ is invertible and

$$x_S = A_S^{-1} b, \qquad x_{\bar{S}} = 0, \qquad x \geq 0.$$

- $A_S \in \mathbf{R}^{m \times m}$ is submatrix of A with columns in S
- two BFS with S, S' are neighbors if they share all but one columns: $|S \cap S'| = m - 1$

Q: how to find a BFS?
A: choose m linearly independent columns of A and set $x = A_S^{-1} b$; check $x \geq 0$.

Extreme point $\iff$ vertex $\iff$ BFS

fact. consider the feasible set $F = \{x \mid Ax = b,\ x \geq 0\}$ in $\mathbf{R}^n$. the following are equivalent:

- x is an extreme point of F
- x is a vertex of F
- x is a BFS of F

Extreme point $\iff$ vertex $\iff$ BFS

fact. consider the feasible set $F = \{x \mid Ax = b,\ x \geq 0\}$ in $\mathbf{R}^n$. the following are equivalent:

- x is an extreme point of F
- x is a vertex of F
- x is a BFS of F

implications: since any polyhedron $Ax \leq b$ can be written as $Ax = b,\ x \geq 0$,

- (BFS $\implies$) a polyhedron has a finite number of extreme points
- (extreme point $\implies$) BFS are independent of the representation of the feasible set

Extreme point $\iff$ vertex $\iff$ BFS

fact. consider the feasible set $F = \{x \mid Ax = b,\ x \geq 0\}$ in $\mathbf{R}^n$. the following are equivalent:

- x is an extreme point of F
- x is a vertex of F
- x is a BFS of F

implications: since any polyhedron $Ax \leq b$ can be written as $Ax = b,\ x \geq 0$,

- (BFS $\Longrightarrow$) a polyhedron has a finite number of extreme points
- (extreme point $\Longrightarrow$) BFS are independent of the representation of the feasible set

we have already shown that vertex $\Longrightarrow$ extreme point. need to show

- extreme point $\Longrightarrow$ BFS
- BFS $\Longrightarrow$ vertex

Extreme point $\implies$ BFS

we will show the contrapositive: x is not a BFS $\implies$ x is not an extreme point

Extreme point $\implies$ BFS

we will show the contrapositive: x is not a BFS $\implies$ x is not an extreme point

suppose that $x^\star \in F$ but is not a BFS:
there is no $S \subseteq [n]$ so that A_S is invertible, $x^\star_S = A_S^{-1} b$, and $x^\star_{\bar{S}} = 0$.

Extreme point $\implies$ BFS

we will show the contrapositive: x is not a BFS $\implies$ x is not an extreme point

suppose that $x^\star \in F$ but is not a BFS:
there is no $S \subseteq [n]$ so that A_S is invertible, $x^\star_S = A_S^{-1} b$, and $x^\star_{\bar{S}} = 0$.
consider $I = \{i : x^\star_i > 0\}$, the active set of variables in $x^\star$.

- if A_I were full rank $|I|$, we could complete A_I to an invertible A_S,
- so there is some $d_I \in \textbf{nullspace}(A_I)$, $d_I \neq 0$.

Extreme point $\implies$ BFS

we will show the contrapositive: x is not a BFS $\implies$ x is not an extreme point

suppose that $x^\star \in F$ but is not a BFS:
there is no $S \subseteq [n]$ so that A_S is invertible, $x^\star_S = A_S^{-1} b$, and $x^\star_{\bar{S}} = 0$.
consider $I = \{i : x^\star_i > 0\}$, the active set of variables in $x^\star$.

- if A_I were full rank $|I|$, we could complete A_I to an invertible A_S,
- so there is some $d_I \in \mathbf{nullspace}(A_I)$, $d_I \neq 0$.

extend this vector to $d \in \mathbf{R}^n$ with $d_{\bar{I}} = 0$, so $Ad = A_I d_I = 0$.
now for $\epsilon \leq \min_i x^\star_i / \max_i |d_i|$, define $x^+, x^- \in \mathbf{R}^n$ as

$$x^+ = x^\star + \epsilon d, \qquad x^- = x^\star - \epsilon d.$$

these are feasible:

- $x^+, x^- \geq 0$ by our choice of ϵ,
- $Ax^+ = Ax^- = b$ since $Ad = 0$.

Extreme point $\implies$ BFS

we will show the contrapositive: x is not a BFS $\implies$ x is not an extreme point

suppose that $x^\star \in F$ but is not a BFS:
there is no $S \subseteq [n]$ so that A_S is invertible, $x^\star_S = A_S^{-1}b$, and $x^\star_{\bar{S}} = 0$.
consider $I = \{i : x^\star_i > 0\}$, the active set of variables in $x^\star$.

- if A_I were full rank $|I|$, we could complete A_I to an invertible A_S,
- so there is some $d_I \in \textbf{nullspace}(A_I)$, $d_I \neq 0$.

extend this vector to $d \in \mathbf{R}^n$ with $d_{\bar{I}} = 0$, so $Ad = A_I d_I = 0$.
now for $\epsilon \leq \min_i x^\star_i / \max_i |d_i|$, define $x^+, x^- \in \mathbf{R}^n$ as

$$x^+ = x^\star + \epsilon d, \qquad x^- = x^\star - \epsilon d.$$

these are feasible:

- $x^+, x^- \geq 0$ by our choice of ϵ,
- $Ax^+ = Ax^- = b$ since $Ad = 0$.

so $x^\star = \frac{1}{2}x^+ + \frac{1}{2}x^-$ is not extreme in F.

BFS $\implies$ vertex

suppose $x^\star$ is a BFS of F with active set S and A_S invertible. define $c \in \mathbf{R}^n$ as

$$c_i = \begin{cases} 0 & \text{if } i \in S \\ 1 & \text{otherwise} \end{cases}$$

so $c^T x^\star = 0$.

BFS $\implies$ vertex

suppose $x^\star$ is a BFS of F with active set S and A_S invertible. define $c \in \mathbf{R}^n$ as

$$c_i = \begin{cases} 0 & \text{if } i \in S \\ 1 & \text{otherwise} \end{cases}$$

so $c^T x^\star = 0$.

- $x^\star$ is the only point in F supported on S, as **nullspace**$(A_S) = 0$,
- so any other feasible point $x \in F$ has a positive objective value $c^T x > 0$.

hence $x^\star$ is a vertex of F with defining vector c.

Outline

Solving LPs

algorithms:

- enumerate all vertices and check
- fourier-motzkin elimination
- simplex method
- ellipsoid method
- interior point methods
- first-order methods
- ...

Solving LPs

algorithms:

- enumerate all vertices and check
- fourier-motzkin elimination
- simplex method
- ellipsoid method
- interior point methods
- first-order methods
- ...

remarks:

- enumeration and elimination are simple but not practical
- simplex was the first practical algorithm; still used today
- ellipsoid method is the first polynomial-time algorithm; not practical
- interior point methods are polynomial-time and practical
- first-order methods are practical and scale to large problems

Example of Fourier-Motzkin elimination

consider the system of inequalities

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Example of Fourier-Motzkin elimination

consider the system of inequalities

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

we can collect inequalities on x_1 into those bounding it above and below:

$$\{0, x_2 - 1\} \leq x_1 \leq 4 - 2x_2$$

Example of Fourier-Motzkin elimination

consider the system of inequalities

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

we can collect inequalities on x_1 into those bounding it above and below:

$$\{0, x_2 - 1\} \leq x_1 \leq 4 - 2x_2$$

by appending all pairwise inequalities to existing inequalities on x_2, we recover the feasible set for x_2:

$$\begin{aligned} 0 &\leq 4 - 2x_2 \\ x_2 - 1 &\leq 4 - 2x_2 \\ x_2 &\geq 0 \end{aligned}$$

$\implies x_2 \in [0, 5/3]$.

Example of Fourier-Motzkin elimination

consider the system of inequalities

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ -x_1 + x_2 &\leq 1 \\ x_1, x_2 &\geq 0 \end{aligned}$$

we can collect inequalities on x_1 into those bounding it above and below:

$$\{0, x_2 - 1\} \leq x_1 \leq 4 - 2x_2$$

by appending all pairwise inequalities to existing inequalities on x_2, we recover the feasible set for x_2:

$$\begin{aligned} 0 &\leq 4 - 2x_2 \\ x_2 - 1 &\leq 4 - 2x_2 \\ x_2 &\geq 0 \end{aligned}$$

$\implies x_2 \in [0, 5/3]$.

elimination method also shows projection of a polyhedron is a (closed) polyhedron

Enumerate vertices of LP

can generate all extreme points of LP: for each $S \subseteq \{1, \ldots, n\}$ with $|S| = m$,

- $A_S \in \mathbf{R}^{m \times m}$, submatrix of A with columns in S, is invertible
- solve $A_S x_S = b$ for x_S and set $x_{\bar{S}} = 0$
- if $x_S \geq 0$, then x is a BFS
- evaluate objective $c^T x$

the best BFS is optimal!

Enumerate vertices of LP

can generate all extreme points of LP: for each $S \subseteq \{1, \ldots, n\}$ with $|S| = m$,

- $A_S \in \mathbf{R}^{m \times m}$, submatrix of A with columns in S, is invertible
- solve $A_S x_S = b$ for x_S and set $x_{\bar{S}} = 0$
- if $x_S \geq 0$, then x is a BFS
- evaluate objective $c^T x$

the best BFS is optimal!

problem: how many BFSs are there?

Enumerate vertices of LP

can generate all extreme points of LP: for each $S \subseteq \{1, \ldots, n\}$ with $|S| = m$,

- $A_S \in \mathbf{R}^{m \times m}$, submatrix of A with columns in S, is invertible
- solve $A_S x_S = b$ for x_S and set $x_{\bar{S}} = 0$
- if $x_S \geq 0$, then x is a BFS
- evaluate objective $c^T x$

the best BFS is optimal!

problem: how many BFSs are there?
n choose m is $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ ("exponentially many")

Simplex algorithm

basic idea: local search on the vertices of the feasible set

- start at BFS x and evaluate objective $c^T x$
- move to a neighboring BFS x' with better objective $c^T x'$
- repeat until no improvement possible

Simplex algorithm

basic idea: local search on the vertices of the feasible set

- start at BFS x and evaluate objective $c^T x$
- move to a neighboring BFS x' with better objective $c^T x'$
- repeat until no improvement possible

discuss in groups:

- how to find an initial BFS?
- how to find a neighboring BFS with better objective?
- how to prove optimality?

Finding an initial BFS

solve an auxiliary problem for which a BFS is known:

Finding an initial BFS

solve an auxiliary problem for which a BFS is known:

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} z_i \\ \text{subject to} & Ax + Dz = b \\ & x, z \geq 0 \end{array}$$

where $D \in \mathbf{R}^{m \times m}$ is a diagonal matrix with $D_{ii} = \mathbf{sign}(b_i)$ for $i = 1, \ldots, m$.

Finding an initial BFS

solve an auxiliary problem for which a BFS is known:

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m z_i \\ \text{subject to} & Ax + Dz = b \\ & x, z \geq 0 \end{array}$$

where $D \in \mathbf{R}^{m \times m}$ is a diagonal matrix with $D_{ii} = \mathbf{sign}(b_i)$ for $i = 1, \ldots, m$.

- $x = 0$, $z = |b|$ is a BFS of this problem

Finding an initial BFS

solve an auxiliary problem for which a BFS is known:

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} z_i \\ \text{subject to} & Ax + Dz = b \\ & x, z \geq 0 \end{array}$$

where $D \in \mathbf{R}^{m \times m}$ is a diagonal matrix with $D_{ii} = \mathbf{sign}(b_i)$ for $i = 1, \ldots, m$.

- $x = 0$, $z = |b|$ is a BFS of this problem
- $(x, z) = (x, 0)$ is a BFS of this problem $\iff$ x is a BFS of the original problem

Find a better neighboring BFS

start with BFS x with active set S, $x_S > 0$. (called a **non-degenerate** BFS.)
construct the j**th basic direction** d^j by turning on variable $j \notin S$

$$x^+ \leftarrow x + \theta d^j, \qquad \theta > 0$$

where $d^j_j = 1$ and $d^j_i = 0$ for $i \notin S \cup \{j\}$. need to solve for d^j_S.

Find a better neighboring BFS

start with BFS x with active set S, $x_S > 0$. (called a **non-degenerate** BFS.)
construct the j**th basic direction** d^j by turning on variable $j \notin S$

$$x^+ \leftarrow x + \theta d^j, \qquad \theta > 0$$

where $d^j_j = 1$ and $d^j_i = 0$ for $i \notin S \cup \{j\}$. need to solve for d^j_S.

- need to stay feasible wrt equality constraints, so need

$$0 = Ad^j = A_S d^j_S + a_j \implies d^j_S = -A_S^{-1} a_j$$

Find a better neighboring BFS

start with BFS x with active set S, $x_S > 0$. (called a **non-degenerate** BFS.)
construct the j**th basic direction** d^j by turning on variable $j \notin S$

$$x^+ \leftarrow x + \theta d^j, \qquad \theta > 0$$

where $d^j_j = 1$ and $d^j_i = 0$ for $i \notin S \cup \{j\}$. need to solve for d^j_S.

- need to stay feasible wrt equality constraints, so need

$$0 = Ad^j = A_S d^j_S + a_j \implies d^j_S = -A_S^{-1} a_j$$

- as $x_S > 0$ is non-degenerate, $\exists \theta > 0$ st $x^+ \geq 0$

Find a better neighboring BFS

start with BFS x with active set S, $x_S > 0$. (called a **non-degenerate** BFS.)
construct the j**th basic direction** d^j by turning on variable $j \notin S$

$$x^+ \leftarrow x + \theta d^j, \qquad \theta > 0$$

where $d^j_j = 1$ and $d^j_i = 0$ for $i \notin S \cup \{j\}$. need to solve for d^j_S.

- need to stay feasible wrt equality constraints, so need

$$0 = Ad^j = A_S d^j_S + a_j \implies d^j_S = -A_S^{-1} a_j$$

- as $x_S > 0$ is non-degenerate, $\exists \theta > 0$ st $x^+ \geq 0$
- how does objective change if we move to $x^+ = x + \theta d^j$?

$$c^T x^+ - c^T x = \theta c^T d^j = \theta c_j - \theta c_S^T A_S^{-1} a_j$$

Reduced cost

define **reduced cost** $\bar{c}_j = c_j - c_S^T A_S^{-1} a_j$, $j \notin S$

Reduced cost

define **reduced cost** $\bar{c}_j = c_j - c_S^T A_S^{-1} a_j$, $j \notin S$

fact:

- if $\bar{c} \geq 0$, x is optimal
- if x is optimal and nondegenerate ($x_S > 0$), then $\bar{c} \geq 0$

why might x be degenerate? why might that pose a problem?

if $\bar{c} \geq 0$, x is optimal

three steps to the proof:

- every feasible direction at x is contained in **cone**$(\{d_j \mid j \notin S\})$

if $\bar{c} \geq 0$, x is optimal

three steps to the proof:

- every feasible direction at x is contained in $\textbf{cone}(\{d_j \mid j \notin S\})$
  feasible directions d must satisfy, for some $\theta \geq 0$,

$$A(x + \theta d) = b, \quad x + \theta d \geq 0$$

  - nonnegativity requires $d_j \geq 0$ for $j \notin S$
  - feasibility requires $0 = Ad = A(d_S + \sum_{j \notin S} \alpha_j e_j)$ for some $\alpha \geq 0$
  - solve: $d_S = -A_S^{-1} \sum_{j \notin S} \alpha_j A_j = \sum_{j \notin S} \alpha_j (-A_S^{-1} A_j) = \sum_{j \notin S} \alpha_j d_S^j$
  - so $d = \sum_{j \notin S} \alpha_j (d_S^j + e_j) = \sum_{j \notin S} \alpha_j d^j$

if $\bar{c} \geq 0$, x is optimal

three steps to the proof:

- every feasible direction at x is contained in $\textbf{cone}(\{d_j \mid j \notin S\})$
  feasible directions d must satisfy, for some $\theta \geq 0$,
  $$A(x + \theta d) = b, \quad x + \theta d \geq 0$$
  - nonnegativity requires $d_j \geq 0$ for $j \notin S$
  - feasibility requires $0 = Ad = A(d_S + \sum_{j \notin S} \alpha_j e_j)$ for some $\alpha \geq 0$
  - solve: $d_S = -A_S^{-1} \sum_{j \notin S} \alpha_j A_j = \sum_{j \notin S} \alpha_j (-A_S^{-1} A_j) = \sum_{j \notin S} \alpha_j d_S^j$
  - so $d = \sum_{j \notin S} \alpha_j (d_S^j + e_j) = \sum_{j \notin S} \alpha_j d^j$
- the feasible set $F = \{x \mid Ax = b, \ x \geq 0\} \subseteq x + \textbf{cone}(\{d_j \mid j \notin S\})$

if $\bar{c} \geq 0$, x is optimal

three steps to the proof:

- every feasible direction at x is contained in $\mathbf{cone}(\{d_j \mid j \notin S\})$
  feasible directions d must satisfy, for some $\theta \geq 0$,

$$A(x + \theta d) = b, \quad x + \theta d \geq 0$$

  - nonnegativity requires $d_j \geq 0$ for $j \notin S$
  - feasibility requires $0 = Ad = A(d_S + \sum_{j \notin S} \alpha_j e_j)$ for some $\alpha \geq 0$
  - solve: $d_S = -A_S^{-1} \sum_{j \notin S} \alpha_j A_j = \sum_{j \notin S} \alpha_j (-A_S^{-1} A_j) = \sum_{j \notin S} \alpha_j d_S^j$
  - so $d = \sum_{j \notin S} \alpha_j (d_S^j + e_j) = \sum_{j \notin S} \alpha_j d^j$
- the feasible set $F = \{x \mid Ax = b,\ x \geq 0\} \subseteq x + \mathbf{cone}(\{d_j \mid j \notin S\})$ by convexity

if $\bar{c} \geq 0$, x is optimal

three steps to the proof:

- every feasible direction at x is contained in $\textbf{cone}(\{d_j \mid j \notin S\})$
  feasible directions d must satisfy, for some $\theta \geq 0$,

$$A(x + \theta d) = b, \quad x + \theta d \geq 0$$

  - nonnegativity requires $d_j \geq 0$ for $j \notin S$
  - feasibility requires $0 = Ad = A(d_S + \sum_{j \notin S} \alpha_j e_j)$ for some $\alpha \geq 0$
  - solve: $d_S = -A_S^{-1} \sum_{j \notin S} \alpha_j A_j = \sum_{j \notin S} \alpha_j (-A_S^{-1} A_j) = \sum_{j \notin S} \alpha_j d_S^j$
  - so $d = \sum_{j \notin S} \alpha_j (d_S^j + e_j) = \sum_{j \notin S} \alpha_j d^j$
- the feasible set $F = \{x \mid Ax = b,\ x \geq 0\} \subseteq x + \textbf{cone}(\{d_j \mid j \notin S\})$ by convexity
- so

$$\begin{aligned} p^\star = \min_{x' \in F} c^T x' &\geq \min_{\alpha \geq 0} c^T (x + \sum_{j \notin S} \alpha_j d_j) \\ &= c^T x + \min_{\alpha \geq 0} \sum_{j \notin S} \alpha_j \bar{c}_j = c^T x \end{aligned}$$

Outline

Let's do some modeling!

practical solvers for MILP:

- Gurobi and COPT are state-of-the-art commercial solvers
- GLPK and SCIP are free solvers that are not as fast

Let's do some modeling!

practical solvers for MILP:

- Gurobi and COPT are state-of-the-art commercial solvers
- GLPK and SCIP are free solvers that are not as fast
- gurobipy is a python interface to Gurobi
- CVX* (including CVXPY in python) are modeling languages that call solvers with good support for convex problems

Let's do some modeling!

practical solvers for MILP:

- Gurobi and COPT are state-of-the-art commercial solvers
- GLPK and SCIP are free solvers that are not as fast
- gurobipy is a python interface to Gurobi
- CVX* (including CVXPY in python) are modeling languages that call solvers with good support for convex problems
- OptiMUS is a LLM-based modeling tool for MILP that produces gurobipy code https://optimus-solver.com/dashboard

Let's do some modeling!

practical solvers for MILP:

- Gurobi and COPT are state-of-the-art commercial solvers
- GLPK and SCIP are free solvers that are not as fast
- gurobipy is a python interface to Gurobi
- CVX* (including CVXPY in python) are modeling languages that call solvers with good support for convex problems
- OptiMUS is a LLM-based modeling tool for MILP that produces gurobipy code https://optimus-solver.com/dashboard
- JuliaOpt/JuMP is a modeling language in Julia that calls solvers and is super speedy for MILP applications

Let's do some modeling!

practical solvers for MILP:

- Gurobi and COPT are state-of-the-art commercial solvers
- GLPK and SCIP are free solvers that are not as fast
- gurobipy is a python interface to Gurobi
- CVX* (including CVXPY in python) are modeling languages that call solvers with good support for convex problems
- OptiMUS is a LLM-based modeling tool for MILP that produces gurobipy code https://optimus-solver.com/dashboard
- JuliaOpt/JuMP is a modeling language in Julia that calls solvers and is super speedy for MILP applications demos:
  - power systems https://jump.dev/JuMP.jl/stable/tutorials/applications/power_systems/
  - multicast routing https://colab.research.google.com/drive/1iOn1T1Muh51KaA7mf7UlQOdhSFZhZyry?usp=sharing

Oro Verde case + tutorial

https://github.com/stanford-cme-307/demos/tree/main/gurobipy

Modeling challenges

model the following as standard form LPs:

1. **inequality constraints.** $Ax \leq b$
2. **free variable.** $x \in \mathbf{R}$
3. **absolute value.** constraint $|x| \leq 10$
4. **piecewise linear.** objective $\max(x_1, x_2)$
5. **assignment.** *e.g.*, every class is assigned exactly one classroom
6. **logic.** *e.g.*, class enrollment $\leq$ capacity of assigned room
7. **(big-M).** $Ax \leq b$ if $x \geq 10$
8. **flow.** *e.g.*, the least cost way to ship an item from s to t

Modeling challenges

model the following as standard form LPs:

1. **inequality constraints.** $Ax \leq b$
2. **free variable.** $x \in \mathbf{R}$
3. **absolute value.** constraint $|x| \leq 10$
4. **piecewise linear.** objective $\max(x_1, x_2)$
5. **assignment.** *e.g.*, every class is assigned exactly one classroom
6. **logic.** *e.g.*, class enrollment $\leq$ capacity of assigned room
7. **(big-M).** $Ax \leq b$ if $x \geq 10$
8. **flow.** *e.g.*, the least cost way to ship an item from s to t

(see chapter 1 of Bertsimas and Tsitsiklis for more details on 1–6. see https://github.com/stanford-cme-307/demos/blob/main/Mullticast_Routing_Demonstration.ipynb for a detailed treatment of a flow problem.)

Use slack variables to represent inequality constraints

to represent the following problem in standard form,

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & Ax \leq b \\
& x \geq 0
\end{array}
$$

Use slack variables to represent inequality constraints

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & x \geq 0 \end{array}$$

introduce slack variable $s \in \mathbf{R}^m$: $Ax + s = b$, $s \geq 0 \iff Ax \leq b$

$$\begin{array}{ll} \text{minimize} & c^T x + 0^T s \\ \text{subject to} & Ax + s = b \\ & x, s \geq 0 \end{array}$$

Split variable into parts to represent free variables

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \end{array}$$

Split variable into parts to represent free variables

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \end{array}$$

introduce positive variables x_+, x_- so $x = x_+ - x_-$:

$$\begin{array}{ll} \text{minimize} & c^T x_+ - c^T x_- \\ \text{subject to} & Ax_+ - Ax_- = b \\ & x_+, x_- \geq 0 \end{array}$$

Use epigraph variables to handle absolute value

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & \|x\|_1 = \sum_{i=1}^{n} |x_i| \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

Use epigraph variables to handle absolute value

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & \|x\|_1 = \sum_{i=1}^n |x_i| \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

introduce epigraph variable $t \in \mathbf{R}^n$ so $|x_i| \leq t_i$:

$$\begin{array}{ll} \text{minimize} & 1^T t \\ \text{subject to} & Ax = b \\ & -t \leq x \leq t \\ & x, t \geq 0 \end{array}$$

verify these constraints ensure $|x_i| \leq t_i$.

Use epigraph variables to handle absolute value

to represent the following problem in standard form,

$$\begin{array}{ll} \text{minimize} & \|x\|_1 = \sum_{i=1}^n |x_i| \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

introduce epigraph variable $t \in \mathbf{R}^n$ so $|x_i| \leq t_i$:

$$\begin{array}{ll} \text{minimize} & 1^T t \\ \text{subject to} & Ax = b \\ & -t \leq x \leq t \\ & x, t \geq 0 \end{array}$$

verify these constraints ensure $|x_i| \leq t_i$.
Q: Why does this work? For what kinds of functions can we use this trick?

Use binary variables to handle assignment

every class is assigned exactly one classroom:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \ldots, n$ and room $j = 1, \ldots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

Use binary variables to handle assignment

every class is assigned exactly one classroom:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \ldots, n$ and room $j = 1, \ldots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

now solve the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij} \\ \text{subject to} & \sum_{j=1}^{m} X_{ij} = 1, \ \forall i \quad \text{(every class assigned one room)} \\ & \sum_{i=1}^{n} X_{ij} \leq 1, \ \forall j \text{(no more than one class per room)} \\ & X_{ij} \in \{0, 1\} \quad \text{(binary variables)} \end{array}$$

where C_{ij} is the cost of assigning class i to room j.

Use binary variables to handle logic

model class enrollment $p_i \leq$ capacity c_j of assigned room:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \ldots, n$ and room $j = 1, \ldots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

Use binary variables to handle logic

model class enrollment $p_i \leq$ capacity c_j of assigned room:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \dots, n$ and room $j = 1, \dots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

solve the problem

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij} & \\ \text{subject to} & \sum_{j=1}^{m} X_{ij} = 1, \ \forall i & \text{(every class assigned one room)} \\ & \sum_{i=1}^{n} X_{ij} \leq 1, \ \forall j & \text{(no more than one class per room)} \\ & \sum_{i=1}^{n} p_i X_{ij} \leq c_j, \ \forall j & \text{(capacity constraint)} \\ & X_{ij} \in \{0, 1\} & \text{(binary variables)} \end{array}$$

where C_{ij} is the cost of assigning class i to room j.

Use binary variables to handle logic

model class enrollment $p_i \leq$ capacity c_j of assigned room:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \dots, n$ and room $j = 1, \dots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

solve the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij} \\ \text{subject to} & \sum_{j=1}^{m} X_{ij} = 1, \ \forall i \quad \text{(every class assigned one room)} \\ & \sum_{i=1}^{n} X_{ij} \leq 1, \ \forall j \text{(no more than one class per room)} \\ & \sum_{i=1}^{n} p_i X_{ij} \leq c_j, \ \forall j \quad \text{(capacity constraint)} \\ & X_{ij} \in \{0, 1\} \quad \text{(binary variables)} \end{array}$$

where C_{ij} is the cost of assigning class i to room j.

what if we want enrollment p to be a variable, too?

. . . or use a big-M relaxation!

model class enrollment $p_i \leq$ capacity c_j of assigned room:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \ldots, n$ and room $j = 1, \ldots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

suppose M is a very large number.

...or use a big-M relaxation!

model class enrollment $p_i \leq$ capacity c_j of assigned room:
define variable $X_{ij} \in \{0, 1\}$ for each class $i = 1, \ldots, n$ and room $j = 1, \ldots, m$

$$X_{ij} = \begin{cases} 1 & \text{class } i \text{ is assigned to room } j \\ 0 & \text{otherwise} \end{cases}$$

suppose M is a very large number. solve the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij} \\ \text{subject to} & \sum_{i=1}^{n} X_{ij} = 1, \ \forall j \quad \text{(every class assigned one room)} \\ & \sum_{j=1}^{m} X_{ij} = 1, \ \forall i \text{(no more than one class per room)} \\ & p_i \leq c_j + (1 - X_{ij}) M, \ \forall i, j \quad \text{(capacity constraint)} \\ & X_{ij} \in \{0, 1\} \quad \text{(binary variables)} \end{array}$$

where C_{ij} is the cost of assigning class i to room j.