CS7800: Advanced Algorithms

Class 10: Linear Programming II

- LP Duality
- Minimax Theorem

Jonathan Ullman

October 10, 2025

Linear Programming

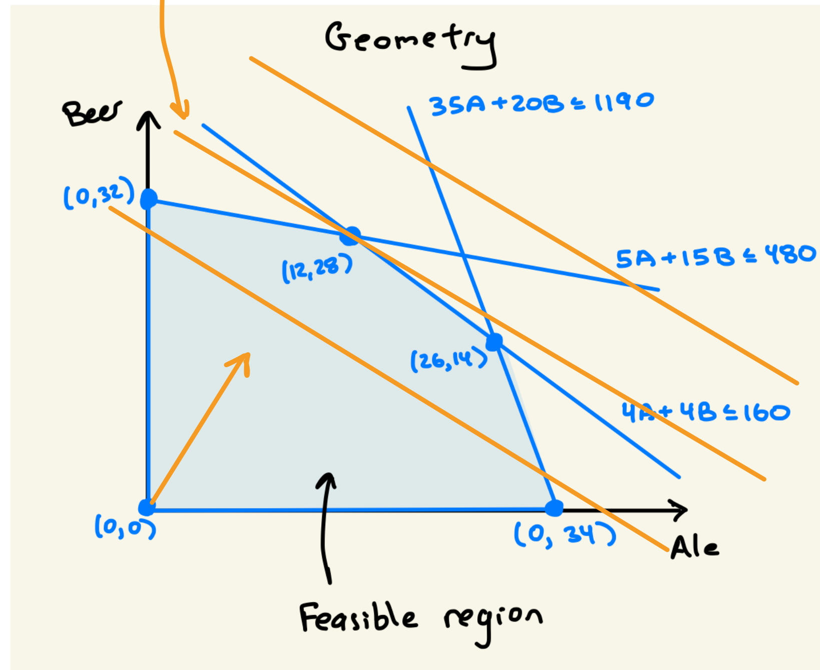
$$\max_{A,B} \quad 13A + 23B$$

$$\text{s.t.} \quad 5A + 15B \le 480$$
$$4A + 4B \le 160$$
$$35A + 20B \le 1190$$
$$A, B \ge 0$$

optimal solution: $A = 12$, $B = 28$

optimal value: 800

13A+23B = 800



How do we know we found an optimal solution?

Upper bound on optimal value

$$\boxed{\begin{aligned} \max_{A,B}\quad & 13A + 23B \\ \text{s.t.}\quad & 5A + 15B \le 480 \\ & 4A + 4B \le 160 \\ & 35A + 20B \le 1190 \\ & A, B \geq 0 \end{aligned}}$$

$\xrightarrow{3\times}$ $15A + 45B \le 1440$

$\xrightarrow{6\times}$ $24A + 24B \le 960$

optimal solution: $A = 12$, $B = 28$

optimal value: 800

How do we know we found an optimal solution?

Upper bound on optimal value

$$\begin{aligned}
\max_{A,B} \quad & 13A + 23B \\
\text{s.t.} \quad & 5A + 15B \le 480 \\
& 4A + 4B \le 160 \\
& 35A + 20B \le 1190 \\
& A, B \ge 0
\end{aligned}$$

1x, 2x, 0x

$$\begin{array}{rl}
 & 5A + 15B \le 480 \\
(+) & 8A + 8B \le 320 \\
\hline
 & 13A + 23B \le 800
\end{array}$$

optimal solution: $A = 12, B = 28$

optimal value: 800

<u>New Problem</u>: Derive the smallest upper bound on optimal value by combining constraints

- Coefficient on each constraint ≥ 0
- Combination upper bounds the objective

How Dual Linear Program optimal solution?

primal (P) optimization problem

$$
\begin{aligned}
\max_{A,B} \quad & 13A + 23B \\
\text{s.t.} \quad & 5A + 15B \le 480 \quad (y_c) \\
& 4A + 4B \le 160 \quad (y_H) \\
& 35A + 20B \le 1190 \quad (y_M) \\
& A, B \ge 0
\end{aligned}
$$

optimal solution: $A = 12, B = 28$

optimal value: 800

dual (D) optimization problem

$$
\begin{aligned}
\min_{y_c, y_H, y_M} \quad & 480y_c + 160y_H + 1190y_M \\
& 5y_c + 4y_H + 35y_M \ge 13 \\
& 15y_c + 4y_H + 20y_M \ge 23 \\
& y_c, y_H, y_M \ge 0
\end{aligned}
$$

optimal solution: $y_c = 1, y_H = 2, y_M = 0$

optimal value: 800

$$y_c \times (5A + 15B) \le y_c \times 480$$

$$y_H \times (4A + 4B) \le y_H \times 160$$

$$y_M \times (35A + 20B) \le y_M \times 1190$$

The Dual of a Linear Program

primal (P) optimization problem

$$\max_{x \in \mathbb{R}^n} \; c^T x$$
$$\text{s.t.} \quad Ax \leq b \quad (y \in \mathbb{R}^m)$$
$$x \geq 0$$

$c \in \mathbb{R}^n$

$A \in \mathbb{R}^{m \times n}$

$b \in \mathbb{R}^m$

dual (D) optimization problem

$$\min_{y \in \mathbb{R}^m} \; y^T b$$
$$\text{s.t.} \quad A^T y \geq c$$
$$y \geq 0$$

Weak Duality

For any feasible $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$

$$c^T x \leq y^T A x \leq y^T b$$

The Dual of a Linear Program

Fact: The dual of the dual is the primal

Fact: Can take the dual without converting to standard form.

Primal	maximize	minimize	Dual
constraints	$a_i x = b_i$ $a_i x \leq b_i$ $a_i x \geq b_i$	y_i unrestricted $y_i \geq 0$ $y_i \leq 0$	variables
variables	$x_i \geq 0$ $x_i \leq 0$ x_i unrestricted	$a_i y \geq c_i$ $a_i y \leq c_i$ $a_i y = c_i$	constraints

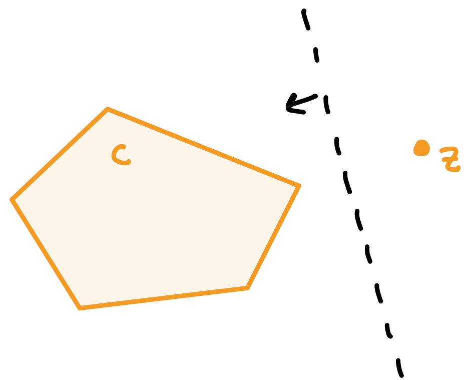
Strong LP Duality

Theorem: If the primal and dual are both feasible then they have the same optimal value

Special cases:

1. If the dual is infeasible, the primal is unbounded
2. If the dual is unbounded, the primal is infeasible

Strong Duality Proof Overview (Idea #1)



Separating Hyperplane Theorem: If $C \subseteq \mathbb{R}^n$ is a closed convex set and $z \in \mathbb{R}^n$ is any point not in C, there exists $\alpha \in \mathbb{R}^n$, $\beta \in \mathbb{R}$ s.t.

1. $\alpha^T x \geq \beta$ for all $x \in C$
2. $\alpha^T z < \beta$

Strong Duality Proof Overview (Idea #2)

<u>Farkas' Lemma</u>: Given $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, exactly one of the following is true:

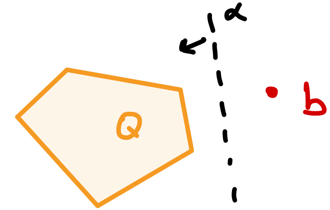
① There exists $x \in \mathbb{R}^n$ s.t. $x \geq 0$ and $Ax = b$

② There exists $y \in \mathbb{R}^m$ s.t. $y^T A \geq 0$ and $y^T b < 0$

Strong Duality Proof Overview (Idea #2)

> **Farkas' Lemma:** Given $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, exactly one of the following is true:
>
> ① There exists $x \in \mathbb{R}^n$ s.t. $x \geq 0$ and $Ax = b$
>
> ② There exists $y \in \mathbb{R}^m$ s.t. $y^T A \geq 0$ and $y^T b < 0$

Proof Sketch:



- Let $Q = \{ w : \exists x \geq 0 \quad Ax = w \}$
- Assume !① so $b \notin Q$
- SHT $\Longrightarrow$ $\exists$ $\alpha^T w \geq 0$ for every $w \in Q$ and $\alpha^T b < 0$

  small trick here

  easy part
- Claim: setting $y = \alpha$ satisfies $y^T A \geq 0$ and $y^T b < 0$
  - $(\alpha^T A)_j = \alpha^T \cdot (j^{th} \text{ column of } A)$
  - $(j^{th} \text{ column of } A) \in Q \Longrightarrow (\alpha^T A)_j \geq 0$ for all $j = 1, \ldots, n$

Strong Duality Proof Overview (Idea #3)

slack form

$$(P) \quad \max_{x} \; c^T x \quad \text{s.t.} \quad Ax = b, \quad x \geq 0$$

assume feasibility

optimal value is v^*

optimal value is $v^* + \varepsilon$ for $\varepsilon \to 0$

$$(D) \quad \min_{y} \; y^T b \qquad y^T A = c, \quad y \text{ unrestricted}$$

$y^T b - (v^* + \varepsilon) < 0$

$y^T b < v^* + \varepsilon$ $y^T A = c$

$\Rightarrow$ does not exist x s.t.

$$c^T x \geq v^* + \varepsilon, \quad Ax = b, \quad x \geq 0$$

$$A' = \begin{bmatrix} c \\ \hline A \end{bmatrix}$$

$$b' = \begin{bmatrix} v^* + \varepsilon \mid b \end{bmatrix}$$

$\xRightarrow{\text{Farkas' Lemma}}$ there exists y s.t.

$$y^T A' \geq 0$$

$$y^T b' < 0$$

Application: The Minimax Theorem

Zero-Sum Games:

- Two players Rowena and Colin
- Rowena chooses an action in $[m]$ Colin chooses in $[n]$
- Payoffs $A \in \mathbb{R}^{m \times n}$

$$\left.\begin{array}{l}\text{Rowena plays } i \\ \text{Colin plays } j\end{array}\right\} \Longrightarrow \begin{array}{l}\text{Rowena gets } A_{ij} \\ \text{Colin gets } -A_{ij}\end{array}$$

"zero-sum"

- Players can play randomly

$$\left.\begin{array}{lll}\text{Rowena: } r = (r_1, \dots, r_m) & \sum_i r_i = 1 & r_i \geq 0 \\ \text{Colin: } c = (c_1, \dots c_n) & \sum_j c_j = 1 & c_j \geq 0\end{array}\right\} \Longrightarrow$$

Rowena's expected payoff is

$$\sum_{ij} r_i c_j A_{ij} = r^T A c$$

Application: Minimax Thm

How would Rowena play if she went first?

How would Colin play if he went first?

Minimax Theorem:

Zero-Sum Games:

- Two players Rowena and Colin
- Rowena chooses an action in $[m]$ Colin chooses in $[n]$
- Payoffs $A \in \mathbb{R}^{m \times n}$

$$\left.\begin{array}{l}\text{Rowena plays } i \\ \text{Colin plays } j\end{array}\right\} \Rightarrow \underbrace{\begin{array}{l}\text{Rowena gets } A_{ij} \\ \text{Colin gets } -A_{ij}\end{array}}_{\text{"zero-sum"}}$$

- Players can play randomly

$$\left.\begin{array}{llll}\text{Rowena:} & r = (r_1, \dots, r_m) & \sum_i r_i = 1 & r_i \geq 0 \\ \text{Colin:} & c = (c_1, \dots, c_n) & \sum_j c_j = 1 & c_j \geq 0\end{array}\right\} \Rightarrow$$

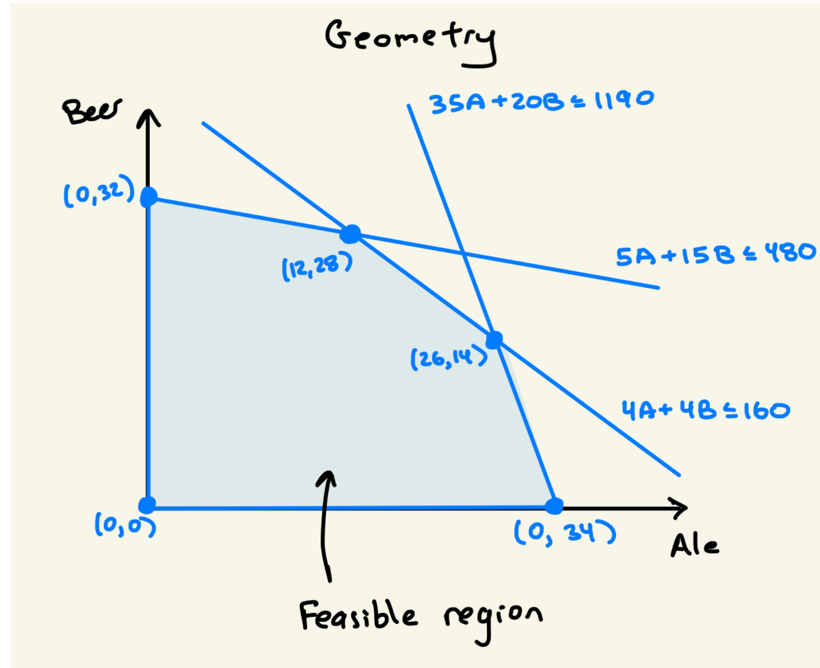
Rowena's expected payoff is

$$\sum_{ij} r_i c_j A_{ij} = r^T A c$$

Application : Minimax Thm Proof

Solving Linear Programs: Simplex

Basic Feasible Solutions (Geometry)



Basic Feasible Solutions (Algebra)

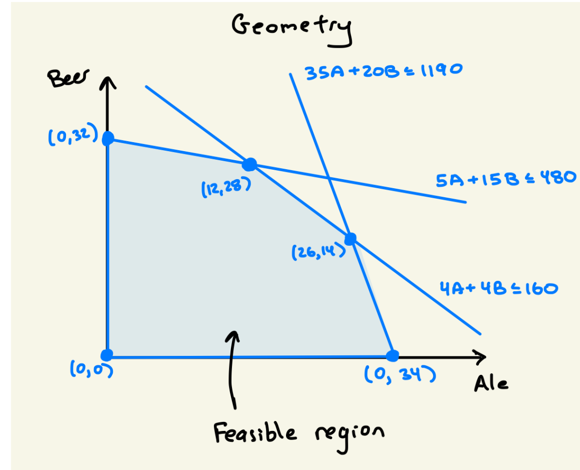
slack form LP

$$\max_{A,B,S_C,S_H,S_M} \quad 13A + 23B$$

s.t.

$$\begin{aligned} 5A + 15B + S_C \quad\quad &= 480 \\ 4A + 4B \quad + S_H \quad &= 160 \\ 35A + 20B \quad\quad + S_M &= 1190 \end{aligned}$$

$$A, B, S_C, S_H, S_M \geq 0$$



constraint matrix

$$\begin{array}{ccccc} (A) & (B) & (S_C) & (S_H) & (S_M) \end{array}$$

$$\begin{bmatrix} 5 & 15 & 1 & 0 & 0 \\ 4 & 4 & 0 & 1 & 0 \\ 35 & 20 & 0 & 0 & 1 \end{bmatrix}$$

The Simplex Algorithm (30,000' view)

Given an LP in standard form

$$\max_x \; c^T x \quad Ax = b,\; x \geq 0$$

Simplex algorithm

- Start with a BFS x_0 corresponding to constraint set S_0

How?

- Repeat until optimality:

How?

  - Find an adjacent BFS x_i corresponding to constraint set S with

$$c^T x_i \geq c^T x_{i-1}$$

Thm: Only terminates at an optimal solution

The Simplex Al orithm (Pivot)

program

max z s.t.

$$13A + 23B \qquad\qquad - z = 0$$

$$\begin{aligned} 5A + 15B + S_c &= 480 \\ 4A + 4B \quad + S_H &= 160 \\ 35A + 20B \quad\quad + S_M &= 1190 \end{aligned}$$

matrix

basis: $\{S_c, S_H, S_M\}$

$$\begin{bmatrix} 5 & 15 & 1 & 0 & 0 \\ 4 & 4 & 0 & 1 & 0 \\ 35 & 20 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ S_c \\ S_H \\ S_M \end{bmatrix} = \begin{bmatrix} 480 \\ 160 \\ 1190 \end{bmatrix}$$

sol

$z = 0$

$A = 0$

$B = 0$

$S_c = 480$

$S_H = 160$

$S_M = 1190$

---

program

max z s.t.

$$\tfrac{16}{3}A \qquad - \tfrac{23}{15}S_c \qquad - z = -736$$

$$\begin{aligned} \tfrac{1}{3}A + B + \tfrac{1}{15}S_c &= 32 \\ \tfrac{8}{3}A + \quad - \tfrac{4}{15}S_c + S_H &= 32 \\ \tfrac{85}{3}A + \quad - \tfrac{4}{3}S_c \quad + S_M &= 550 \end{aligned}$$

matrix

basis: $\{B, S_H, S_M\}$

$$\begin{bmatrix} 1/3 & 1 & 1/15 & 0 & 0 \\ 8/3 & 0 & -4/15 & 1 & 0 \\ 85/3 & 0 & -4/3 & 0 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ S_c \\ S_H \\ S_M \end{bmatrix} = \begin{bmatrix} 32 \\ 32 \\ 550 \end{bmatrix}$$

sol

$z = 736$

$A = 0$

$B = 32$

$S_c = 0$

$S_H = 32$

$S_M = 550$

The Simplex Algorithm (30,000' view)

Given an LP in standard form

$$\max_x \; c^T x$$
$$Ax = b$$
$$x \geq 0$$

Simplex algorithm

- Start with a BFS x_0 corresponding to constraint set S_0

How?

- Repeat until optimality:

How?

  - Find an <u>adjacent BFS</u> x_i corresponding to constraint set S with

$$c^T x_i \geq c^T x_{i-1}$$

<u>Thm</u>: Only terminates at an optimal solution

Simplex in Practice

Theory: Might need exponentially many pivots to terminate

Practice: Can solve LPs with millions of variables/constraints (usually $\leq 2(n+m)$ pivots)

Many Issues to Resolve:

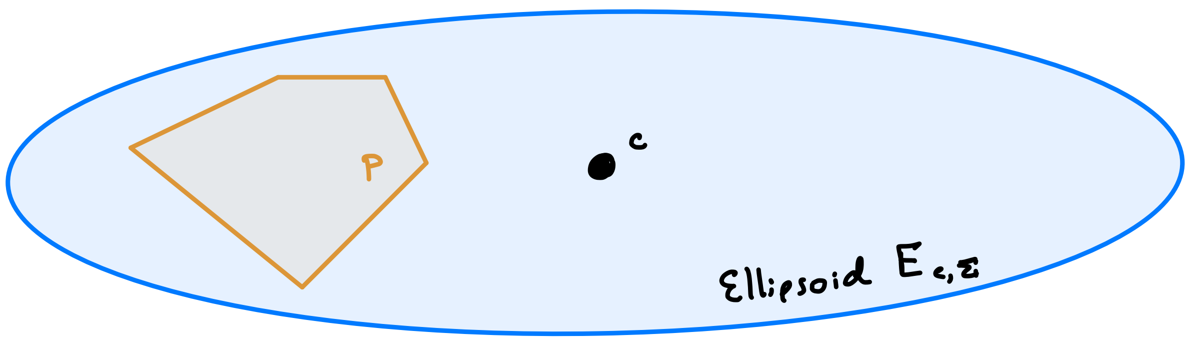
1. What if the LP in infeasible/unbounded?
2. How to choose a good pivot rule?
3. How to avoid cycling?
4. How to maintain sparsity?
5. How to be numerically stable?
6. How to preprocess the LP to be smaller?

Solving Linear Programs: Ellipsoid

Solving linear programs in worst-case polynomial time

⓪ Enough to find a feasible point. (Why?)

① Find an ellipsoid containing P. (How?)

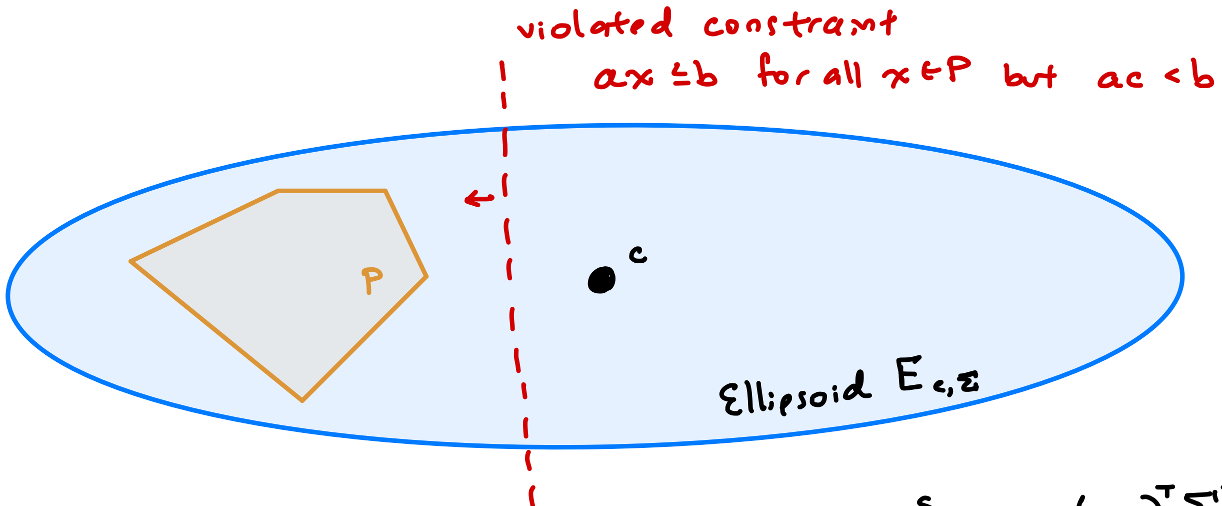


$$E = \{ x : (x-c)^T \Sigma^{-1} (x-c) \le 1 \}$$

Solving Linear Programs: Ellipsoid

Solving linear programs in worst-case polynomial time

② Either $c \in P$ or there is a violated constraint

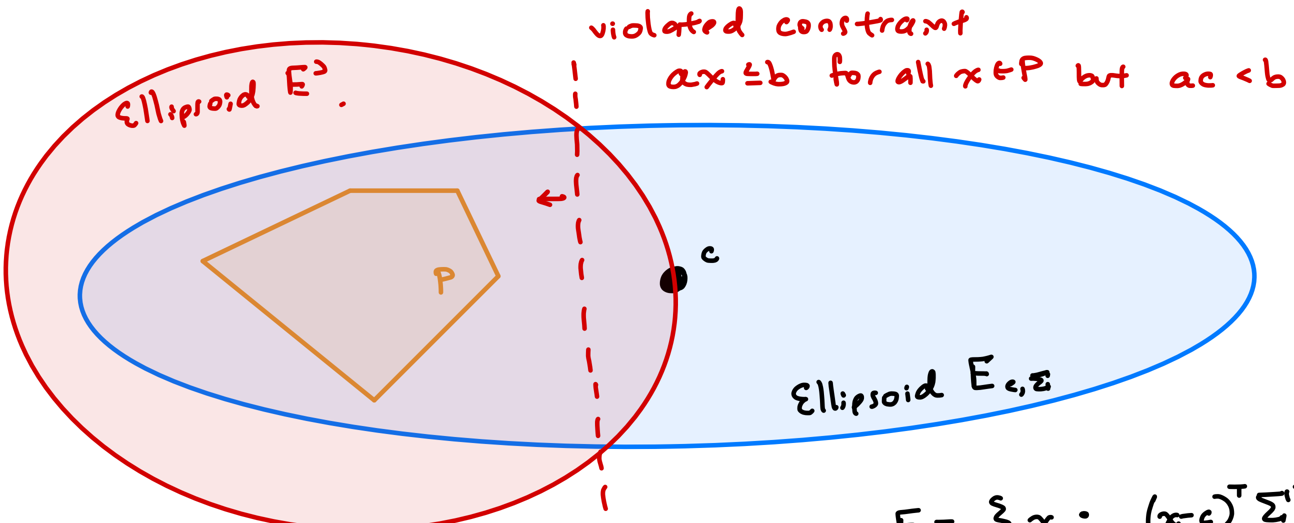


$$E = \{ x : (x-c)^T \Sigma^{-1} (x-c) \leq 1 \}$$

Solving Linear Programs: Ellipsoid

Solving linear programs in worst-case polynomial time

③ Use the violated constraint to find a smaller ellipsoid containing P



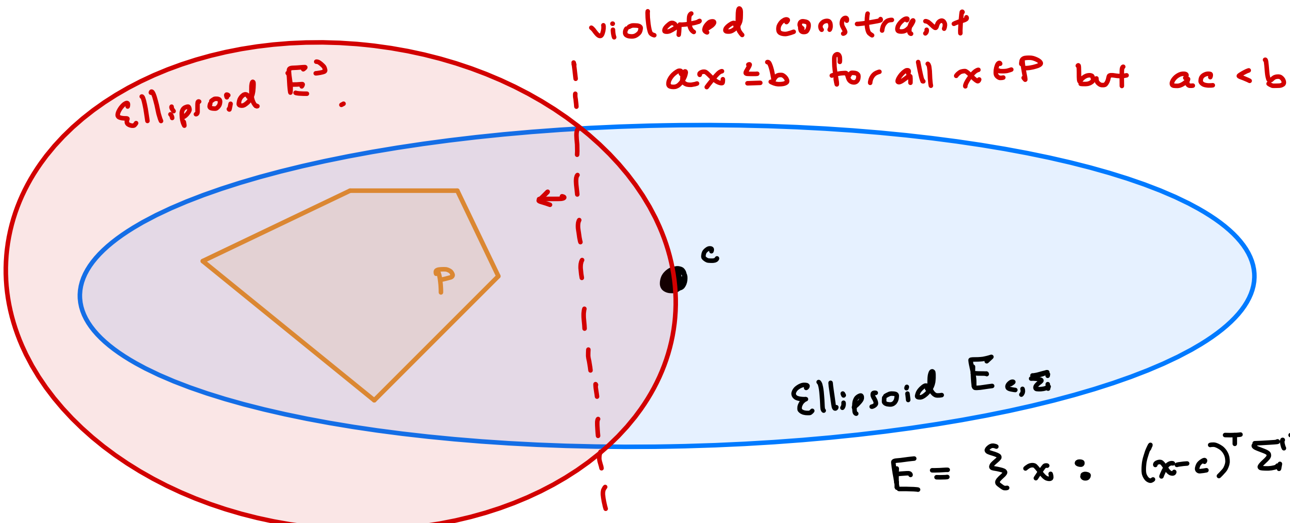
$$E = \{x : (x-c)^T \Sigma^{-1}(x-c) \le 1\}$$

Key Theorem: Can choose E' so that $\frac{\mathrm{vol}(E')}{\mathrm{vol}(E)} \le \left(1 - \frac{1}{2n+2}\right)$

Solving Linear Programs: Ellipsoid

Solving linear programs in worst-case polynomial time

③ Use the violated constraint to find a smaller ellipsoid containing P



$$E = \{x : (x-c)^T \Sigma^{-1} (x-c) \le 1\}$$

Key Theorem: Can choose E' so that $\dfrac{\mathrm{vol}(E')}{\mathrm{vol}(E)} \le \left(1 - \dfrac{1}{2n+2}\right)$

Key Fact: Works any time we can find a "separation oracle" for P!

Linear Programming: Summary

Summary (of Network Flow Algorithms)

- **Last Class:** Can solve maximum flow in time $O(m \cdot v^*)$
  - Can be very slow when capacities are large
  - Cannot be improved if we allow arbitrary augmenting paths

- **Today:** Improving running time by choosing better paths
  - **Widest Augmenting Path:** $O(m \cdot \log v^*)$
  - **Shortest Augmenting Path:** $O(m^2 n)$

- **Still actively studied!**
  - Can solve maximum flow in $O(mn)$ using augmenting path* algos
  - **Recent Breakthrough:** Can solve maximum flow in time* $m^{1+o(1)}$

- **Later On:** Using maximum-flow as a building block for solving many more problems