# CS 473: Algorithms

Chandra Chekuri Ruta Mehta

University of Illinois, Urbana-Champaign

Fall 2016

# LP Duality

Lecture 20

November 2, 2016

# An easy LP?

$$\max cx \text{ subject to } Ax = b$$

which is compact form for

$$\begin{aligned} \max c_1x_1 + c_2x_2 + \ldots + c_nx_n & \\ a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n &= b_i \quad 1 \leq i \leq m \end{aligned}$$

**Question:** Is this a geneal LP problem or is it some how easy?

# An easy LP?

$$\max cx \text{ subject to } Ax = b$$

Basically reduces to linear system solving. Three cases for $Ax = b$.

- The system $Ax = b$ is infeasible, that is, no solution
- The system $Ax = b$ has a unique solution $x^*$ when $rank([A\ b]) = n$ (full rank). Optimum solution value is $cx^*$
- The system $Ax = b$ has infinite solutions when $rank([A\ b]) < n$. There all vectors of the form $x^* + y$ are feasible where $y$ is null-space$(A) = \{y \mid Ay = 0\}$. Let $d$ be dimension of null-space$(A)$ and let $e_1, e_2, \ldots, e_d$ be an orthonormal basis. Then $cx = cx^* + cy = cx^* + c(\lambda_1 e_1 + \lambda_2 e_2 + \ldots + \lambda_d e_d$. If $ce_i \neq 0$ for any $i$ then optimum solution value is unbounded. Otherwise $cx^*$.

# LP Canonical Forms

Two basic canonical forms:

- $\mathbf{max}\ cx, Ax = b, x \geq \mathbf{0}$
- $\mathbf{max}\ cx, Ax \leq b, x \geq \mathbf{0}$

What makes LP non-trivial and different from linear system solving is the additional non-negativity constraint on variables.

# Part I

# Derivation and Definition of Dual LP

# Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

# Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value $2$ for the objective function.

# Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{llllll} \text{maximize} & 4x_1 + & 2x_2 & & \\ \text{subject to} & x_1 + & 3x_2 & \leq & 5 \\ & 2x_1 - & 4x_2 & \leq & 10 \\ & x_1 + & x_2 & \leq & 7 \\ & x_1 & & \leq & 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value $2$ for the objective function.
2. Thus, optimal value $\sigma^*$ is at least $4$.

# Feasible Solutions and Lower Bounds

Consider the program

$$\begin{array}{lrrl} \text{maximize} & 4x_1 + & 2x_2 & \\ \text{subject to} & x_1 + & 3x_2 & \leq 5 \\ & 2x_1 - & 4x_2 & \leq 10 \\ & x_1 + & x_2 & \leq 7 \\ & x_1 & & \leq 5 \end{array}$$

1. $(0, 1)$ satisfies all the constraints and gives value $2$ for the objective function.
2. Thus, optimal value $\sigma^*$ is at least $4$.
3. $(2, 0)$ also feasible, and gives a better bound of $8$.

# Feasible Solutions and Lower Bounds

Consider the program

$$
\begin{array}{lrrl}
\text{maximize} & 4x_1 + & 2x_2 & \\
\text{subject to} & x_1 + & 3x_2 & \leq 5 \\
 & 2x_1 - & 4x_2 & \leq 10 \\
 & x_1 + & x_2 & \leq 7 \\
 & x_1 & & \leq 5
\end{array}
$$

1. $(0, 1)$ satisfies all the constraints and gives value $2$ for the objective function.
2. Thus, optimal value $\sigma^*$ is at least $4$.
3. $(2, 0)$ also feasible, and gives a better bound of $8$.
4. How good is $8$ when compared with $\sigma^*$?

# Obtaining Upper Bounds

1. Let us multiply the first constraint by **2** and the and add it to second constraint

$$\begin{array}{rrrl} 2( & x_1 + & 3x_2 & ) \le 2(5) \\ +1( & 2x_1 - & 4x_2 & ) \le 1(10) \\ \hline & 4x_1 + & 2x_2 & \le 20 \end{array}$$

2. Thus, 20 is an upper bound on the optimum value!

# Generalizing . . .

1. Multiply first equation by $y_1$, second by $y_2$, third by $y_3$ and fourth by $y_4$ (all of $y_1, y_2, y_3, y_4$ being positive) and add

$$\begin{array}{rrrl} y_1( & x_1 + & 3x_2 & ) \le y_1(5) \\ +y_2( & 2x_1 - & 4x_2 & ) \le y_2(10) \\ +y_3( & x_1 + & x_2 & ) \le y_3(7) \\ +y_4( & x_1 & & ) \le y_4(5) \\ \hline \multicolumn{4}{l}{(y_1 + 2y_2 + y_3 + y_4)x_1 + (3y_1 - 4y_2 + y_3)x_2 \le \ldots} \end{array}$$

2. $5y_1 + 10y_2 + 7y_3 + 5y_4$ is an upper bound, provided coefficients of $x_i$ are same as in the objective function, i.e.,

$$y_1 + 2y_2 + y_3 + y_4 = 4 \quad 3y_1 - 4y_2 + y_3 = 2$$

3. The best upper bound is when $5y_1 + 10y_2 + 7y_3 + 5y_4$ is minimized!

# Dual LP: Example

Thus, the optimum value of program

$$\begin{aligned} \text{maximize} \quad & 4x_1 + 2x_2 \\ \text{subject to} \quad & x_1 + 3x_2 \leq 5 \\ & 2x_1 - 4x_2 \leq 10 \\ & x_1 + x_2 \leq 7 \\ & x_1 \leq 5 \end{aligned}$$

is upper bounded by the optimal value of the program

$$\begin{aligned} \text{minimize} \quad & 5y_1 + 10y_2 + 7y_3 + 5y_4 \\ \text{subject to} \quad & y_1 + 2y_2 + y_3 + y_4 = 4 \\ & 3y_1 - 4y_2 + y_3 = 2 \\ & y_1, y_2 \geq 0 \end{aligned}$$

# Dual Linear Program

Given a linear program $\Pi$ in canonical form

$$\begin{array}{ll} \text{maximize} & \sum_{j=1}^{d} c_j x_j \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad i = 1, 2, \ldots n \end{array}$$

the dual $\mathrm{Dual}(\Pi)$ is given by

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{n} b_i y_i & \\ \text{subject to} & \sum_{i=1}^{n} y_i a_{ij} = c_j & j = 1, 2, \ldots d \\ & y_i \geq 0 & i = 1, 2, \ldots n \end{array}$$

# Dual Linear Program

Given a linear program $\Pi$ in canonical form

$$\begin{array}{lll} \text{maximize} & \sum_{j=1}^{d} c_j x_j & \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i & i = 1, 2, \ldots n \end{array}$$

the dual $\mathrm{Dual}(\Pi)$ is given by

$$\begin{array}{lll} \text{minimize} & \sum_{i=1}^{n} b_i y_i & \\ \text{subject to} & \sum_{i=1}^{n} y_i a_{ij} = c_j & j = 1, 2, \ldots d \\ & y_i \geq 0 & i = 1, 2, \ldots n \end{array}$$

## Proposition

$\mathrm{Dual}(\mathrm{Dual}(\Pi))$ *is equivalent to* $\Pi$

# Duality Theorems

## Theorem (Weak Duality)

*If* $\mathbf{x}'$ *is a feasible solution to* $\mathbf{\Pi}$ *and* $\mathbf{y}'$ *is a feasible solution to* $\mathbf{Dual(\Pi)}$ *then* $\mathbf{c} \cdot \mathbf{x}' \leq \mathbf{y}' \cdot \mathbf{b}$.

# Duality Theorems

## Theorem (Weak Duality)

*If* $x'$ *is a feasible solution to* $\Pi$ *and* $y'$ *is a feasible solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x' \leq y' \cdot b$.

## Theorem (Strong Duality)

*If* $x^*$ *is an optimal solution to* $\Pi$ *and* $y^*$ *is an optimal solution to* $\mathrm{Dual}(\Pi)$ *then* $c \cdot x^* = y^* \cdot b$.

Many applications! Maxflow-Mincut theorem can be deduced from duality.

# Proof of Weak Duality

We already saw the proof by the way we derived it but we will do it again formally.

Since $y'$ is feasible to $\mathrm{Dual}(\Pi)$: $y'A = c$

Therefore $c \cdot x' = y'Ax'$

Since $x'$ is feasible $Ax' \leq b$ and hence,

$$c \cdot x' = y'Ax' \leq y' \cdot b$$

# Duality for another canonical form

$$\begin{aligned}
\text{maximize} \quad & 4x_1 + x_2 + 3x_3 \\
\text{subject to} \quad & x_1 + 4x_2 \leq 2 \\
& 2x_1 - x_2 + x_3 \leq 4 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}$$

# Duality for another canonical form

$$\begin{array}{lrrrl} \text{maximize} & 4x_1+ & x_2+ & 3x_3 & \\ \text{subject to} & x_1+ & 4x_2 & & \leq 2 \\ & 2x_1- & x_2+ & x_3 & \leq 4 \\ & & & x_1, x_2, x_3 & \geq 0 \end{array}$$

Choose non-negative $y_1, y_2$ and multiply inequalities

$$\begin{array}{lrrrl} \text{maximize} & 4x_1+ & x_2+ & 3x_3 & \\ \text{subject to} & y_1(x_1+ & 4x_2 & ) & \leq 2y_1 \\ & y_2(2x_1- & x_2+ & x_3) & \leq 4y_2 \\ & & & x_1, x_2, x_3 & \geq 0 \end{array}$$

# Duality for another canonical form

Choose non-negative $y_1, y_2$ and multiply inequalities

$$\begin{array}{lrrrl} \text{maximize} & 4x_1 + & x_2 + & 3x_3 & \\ \text{subject to} & y_1(x_1 + & 4x_2 & ) & \leq 2y_1 \\ & y_2(2x_1 - & x_2 + & x_3) & \leq 4y_2 \\ & & & x_1, x_2, x_3 & \geq 0 \end{array}$$

Adding the inequalities we get an inequality below that is valid for any feasible $x$ and any non-negative $y$:

$$(y_1 + 2y_2)x_1 + (4y_1 - y_2)x_2 + y_2 \leq 2y_1 + 4y_2$$

Suppose we choose $y_1, y_2$ such that
$y_1 + 2y_2 \geq 4$ and $4y_2 - y_2 \geq 1$ and $2y_1 \geq 3$
Then, since $x_1, x_2, x_3 \geq 0$, we have $4x_1 + x_2 + 3x_3 \leq 2y_1 + 4y_2$

# Duality for another canonical form

$$\begin{array}{lrrrl} \text{maximize} & 4x_1 + & x_2 + & 3x_3 & \\ \text{subject to} & x_1 + & 4x_2 & & \leq 2 \\ & 2x_1 - & x_2 + & x_3 & \leq 4 \\ & & & x_1, x_2, x_3 & \geq 0 \end{array}$$

is upper bounded by

$$\begin{array}{lrrl} \text{minimize} & 2y_1 + & 4y_2 & \\ \text{subject to} & y_1 + & 2y_2 & \geq 4 \\ & 4y_1 - & y_2 & \geq 1 \\ & 2y_1 & & \geq 3 \\ & & y_1, y_2 & \geq 0 \end{array}$$

# Duality for another canonical form

Compactly,

For the primal LP max $\boldsymbol{cx}$ subject to $\boldsymbol{Ax} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}$
the dual LP is min $\boldsymbol{yb}$ subject to $\boldsymbol{yA} \geq \boldsymbol{c}, \boldsymbol{y} \geq \boldsymbol{0}$

# Some Useful Duality Properties

Assume primal LP is a maximization LP.

- For a given LP, Dual is another LP. The variables in the dual correspond to "non-trival" primal constraints and vice-versa.
- Dual of the dual LP give us back the primal LP.
- Weak and strong duality theorems.
- If primal is unbounded (objective achieves infinity) then dual LP is infeasible. Why? If dual LP had a feasible solution it would upper bound the primal LP which is not possible.
- If primal is infeasible then dual LP is unbounded.
- Primal and dual optimum solutions satisfy complementary slackness conditions (discussed soon).

# Part II

# Examples of Duality

# Max matching in bipartite graph as LP

Input: $G = (V = L \cup R, E)$

$$
\begin{aligned}
\textbf{max} \quad & \sum_{uv \in E} x_{uv} & \\
s.t. \quad & \sum_{uv \in E} x_{uv} \leq 1 & \forall v \in V. \\
& x_{uv} \geq 0 & \forall uv \in E
\end{aligned}
$$

# Network flow

$s$-$t$ flow in directed graph $G = (V, E)$ with capacities $c$. Assume for simplicity that no incoming edges into $s$.

$$
\begin{aligned}
\textbf{max} \quad & \sum_{(s,v)\in E} x(s,v) & \\
& \sum_{(u,v)\in E} x(u,v) - \sum_{(v,w)\in E} x(v,w) = 0 & \forall v \in V \setminus \{s,t\} \\
& x(u,v) \le c(u,v) & \forall (u,v) \in E \\
& x(u,v) \ge 0 & \forall (u,v) \in E.
\end{aligned}
$$

# Dual of Network Flow

# Part III

# Farkas Lemma and Strong Duality

# Optimization vs Feasibility

Suppose we want to solve LP of the form:

$$\textbf{max } \boldsymbol{c}\boldsymbol{x} \text{ subject to } \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$$

It is an optimization problem. Can we reduce it to a decision problem?

# Optimization vs Feasibility

Suppose we want to solve LP of the form:

$$\max cx \text{ subject to } Ax \leq b$$

It is an optimization problem. Can we reduce it to a decision problem? Yes, via binary search. Find the largest values of $\sigma$ such that the system of inequalities

$$Ax \leq b, cx \geq \sigma$$

is *feasible*. Feasible implies that there is at least one solution.
Caveat: to do binary search need to know the range of numbers. Skip for now since we need to worry about precision issues etc.

# Certificate for (in)feasibility

Suppose we have a system of $m$ inequalities in $n$ variables defined by

$$Ax \leq b$$

- How can we convince some one that there is a feasible solution?
- How can we convince some one that there is **no** feasible solution?

# Theorem of the Alternatives

## Theorem

*Let $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{m}\times\boldsymbol{n}}$ and $\boldsymbol{b} \in \mathbb{R}^{\boldsymbol{m}}$. The system $\boldsymbol{Ax} \leq \boldsymbol{b}$ is either feasible of if it is infeasible then there is a $\boldsymbol{y} \in \mathbb{R}^{\boldsymbol{m}}$ such that $\boldsymbol{y} \geq \mathbf{0}$ and $\boldsymbol{yA} = \mathbf{0}$ and $\boldsymbol{yb} < \mathbf{0}$.*

In other words, if $\boldsymbol{Ax} \leq \boldsymbol{b}$ is infeasible we can demonstrate it via the following compact contradiction. Find a non-negative combination of the rows of $\boldsymbol{A}$ (given by certificate $\boldsymbol{y}$) to derive $\mathbf{0} < \mathbf{1}$.

The preceding theorem can be used to prove strong duality. A fair amount of formal detail though geometric intuition is reasonable.

# Farkas Lemma

From the theorem of alternatives we can derive a useful version of Farkas lemma.

**Theorem**

*A system* $Ax = b, x \geq \mathbf{0}$ *is either feasible or there is a* $y$ *such that* $yA \geq \mathbf{0}$ *and* $yb < \mathbf{0}$. *Then the following hold:*

Nice geometric interpretation.

# Complementary Slackness

## Theorem

*Let $x^*$ be any optimum solution to primal LP $\Pi$ in the canonical form $\max cx, Ax \leq b, x \geq 0$ and $y^*$ be an optimum solution to the dual LP $Dual(\Pi)$ which is $\min yb, yA \geq c, y \geq 0$. Then the following hold.*

- *If $y_i^* > 0$ then $\sum_{j=1}^{n} a_{ij}x_j = b_j$ (the primal constraint for row $i$ is tight).*
- *If $x_j^* > 0$ then $\sum_{i=1}^{m} y_i a_{ij} = c_j$ (the dual constraint for row $j$ is tight).*

*The converse also hold: if $x^*$ and $y^*$ are primal and dual feasible and satisfy complementary slackness conditions then both must be optimal.*

Very useful in various applications. Nice geometric interpretation.

# Part IV

# Integer Linear Programming

# Integer Linear Programming

## Problem

Find a vector $x \in Z^d$ (integer values) that

$$\begin{array}{ll} \text{maximize} & \sum_{j=1}^{d} c_j x_j \\ \text{subject to} & \sum_{j=1}^{d} a_{ij} x_j \leq b_i \quad \text{for } i = 1 \ldots n \end{array}$$

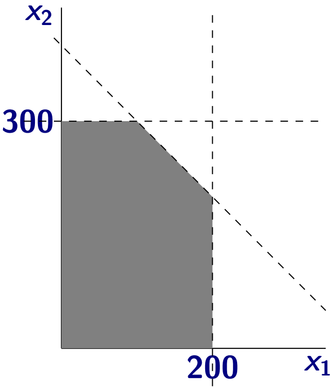
Input is matrix $A = (a_{ij}) \in \mathbb{R}^{n \times d}$, column vector $b = (b_i) \in \mathbb{R}^n$, and row vector $c = (c_j) \in \mathbb{R}^d$

# Factory Example

$$\begin{array}{llll} \text{maximize} & & x_1 + 6x_2 & \\ \text{subject to} & x_1 \leq 200 & x_2 \leq 300 & x_1 + x_2 \leq 400 \\ & & x_1, x_2 \geq 0 & \end{array}$$

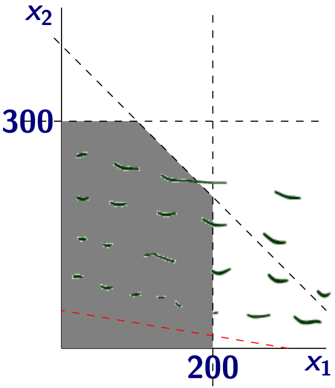
Suppose we want $x_1, x_2$ to be integer valued.

# Factory Example Figure



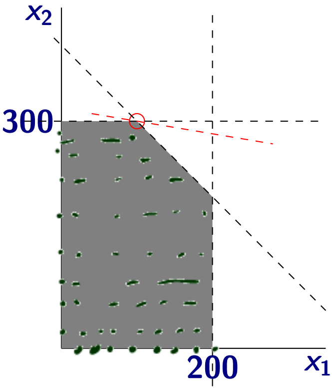

1. Feasible values of $x_1$ and $x_2$ are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

# Factory Example Figure




1. Feasible values of $x_1$ and $x_2$ are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

# Factory Example Figure




1. Feasible values of $x_1$ and $x_2$ are integer points in shaded region
2. Optimization function is a line; moving the line until it just leaves the final integer point in feasible region, gives optimal values

# Integer Programming

Can model many difficult discrete optimization problems as integer programs!

Therefore integer programming is a hard problem. NP-hard.

Can relax integer program to linear program and *approximate*.

Practice: integer programs are solved by a variety of methods

1. branch and bound
2. branch and cut
3. adding cutting planes
4. linear programming plays a fundamental role

# Example: Maximum Independent Set

## Definition

Given undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$ is an independent set (also called a stable set) if for there are no edges between nodes in $S$. That is, if $u, v \in S$ then $(u, v) \notin E$.

**Input** Graph $G = (V, E)$

**Goal** Find maximum sized independent set in $G$

$x_u \quad u \in V \quad x_u \in \{0,1\}$

$$\max \sum_{u \in V} x_u$$

$$x_u + x_v \leq 1 \quad \forall uv \in E$$

$$x_u \in \{0,1\} \quad \forall u \in V$$

$$0 \leq x_u \leq 1 \quad x_u \in \mathbb{Z}$$

# Example: Dominating Set

## Definition

Given undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$ is a dominating set if for all $v \in V$, either $v \in S$ or a neighbor of $v$ is in $S$.

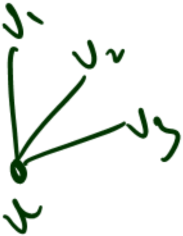
**Input** Graph $G = (V, E)$, weights $w(v) \geq 0$ for $v \in V$

**Goal** Find minimum weight dominating set in $G$

$$x_u \in \{0,1\} \quad u \in V$$

$$\min \sum_{u \in V} x_u$$

$$x_u + \sum_{v \in N(u)} x_v \geq 1 \quad \forall u \in V$$




$$x_u \in \{0,1\} \quad \forall u \in V$$

# Example: s-t minimum cut and implicit constraints

**Input** Graph $G = (V, E)$, edge capacities $c(e), e \in E$. $s, t \in V$

**Goal** Find minimum capacity $s$-$t$ cut in $G$.

$x_e \quad e \in E \qquad x_e \in \{0,1\}$

$$\min \sum_e c(e)\, x(e)$$

$$\sum_{e \in p} x_e \geq 1 \qquad \forall p \in \mathcal{P}_{s,t}$$

$$x_e \in [0,1]$$

$x_v \in \{0,1\} \quad v \in V$

$x_v = 0 \Rightarrow v$ is on $s$-side

$= 1 \Rightarrow v$ is on $t$-side

---

$y_e \in \{0,1\}$ is $e$ cut or not

$$\min \sum_e c(e)\, y(e)$$

$$x_s = 0, \; x_t = 1$$

$0 \leq y_e \leq 1 \qquad y_{uv} \geq x_v - x_u$

$0 \leq x_u \leq 1 \qquad x_u \in \{0,1\},\; y_e \in \{0,1\}$

# Linear Programs with Integer Vertices

*Suppose* we know that for a linear program *all* vertices have integer coordinates.

# Linear Programs with Integer Vertices

*Suppose* we know that for a linear program *all* vertices have integer coordinates.
Then solving linear program is same as solving integer program. We know how to solve linear programs efficiently (polynomial time) and hence we get an integer solution for free!

# Linear Programs with Integer Vertices

*Suppose* we know that for a linear program *all* vertices have integer coordinates.
Then solving linear program is same as solving integer program. We know how to solve linear programs efficiently (polynomial time) and hence we get an integer solution for free!

*Luck* or *Structure*:

1. Linear program for flows with integer capacities have integer vertices
2. Linear program for matchings in bipartite graphs have integer vertices
3. A complicated linear program for matchings in general graphs have integer vertices.

All of above problems can hence be solved efficiently.

# Linear Programs with Integer Vertices

Meta Theorem: A combinatorial optimization problem can be solved efficiently if and only if there is a linear program for problem with integer vertices.

Consequence of the Ellipsoid method for solving linear programming.

*In a sense* linear programming and other geometric generalizations such as convex programming are the most general problems that we can solve efficiently.

# Summary

1. Linear Programming is a useful and powerful (modeling) problem.
2. Can be solved in polynomial time. Practical solvers available commercially as well as in open source. Whether there is a strongly polynomial time algorithm is a major open problem.
3. Geometry and linear algebra are important to understand the structure of LP and in algorithm design. Vertex solutions imply that LPs have poly-sized optimum solutions. This implies that LP is in **NP**.
4. Duality is a critical tool in the theory of linear programming. Duality implies the Linear Programming is in **co-NP**. Do you see why?
5. Integer Programming in **NP-Complete**. LP-based techniques critical in heuristically solving integer programs.