In this lecture discuss the general notion of Linear Programming Duality, a powerful tool that you should definitely master.

1 Linear Programming Duality

Consider the following LP

\[ P = \max(2x_1 + 3x_2) \]
\[ \text{s.t.} \quad 4x_1 + 8x_2 \leq 12 \]
\[ 2x_1 + x_2 \leq 3 \]
\[ 3x_1 + 2x_2 \leq 4 \]
\[ x_1, x_2 \geq 0 \]

In an attempt to solve \( P \) we can produce upper bounds on its optimal value.

- Since \( 2x_1 + 3x_2 \leq 4x_1 + 8x_2 \leq 12 \), we know \( \text{OPT}(P) \leq 12 \). (The first inequality uses that \( 2x_1 \leq 4x_1 \) because \( x_1 \geq 0 \), and similarly \( 3x_2 \leq 8x_2 \) because \( x_2 \geq 0 \).)

- Since \( 2x_1 + 3x_2 \leq \frac{1}{2}(4x_1 + 8x_2) \leq 6 \), we know \( \text{OPT}(P) \leq 6 \).

- Since \( 2x_1 + 3x_2 \leq \frac{1}{3}((4x_1 + 8x_2) + (2x_1 + x_2)) \leq 5 \), we know \( \text{OPT}(P) \leq 5 \).

In each of these cases we take a positive linear combination of the constraints, looking for better and better bounds on the maximum possible value of \( 2x_1 + 3x_2 \).

\[ \text{Why positive? If you multiply by a negative value, the sign of the inequality changes.} \]
How do we find the “best” lower bound that can be achieved as a linear combination of the constraints? This is just another algorithmic problem, and we can systematically solve it, by letting \( y_1, y_2, y_3 \) be the (unknown) coefficients of our linear combination. Then we must have

\[
\begin{align*}
4y_1 + 2y_2 + 3y_2 & \geq 2 \\
8y_1 + y_2 + 2y_3 & \geq 3 \\
y_1, y_2, y_3 & \geq 0 \tag{2}
\end{align*}
\]

and we seek \( \min(12y_1 + 3y_2 + 4y_3) \)

This too is an LP! We refer to this LP (2) as the “dual” and the original LP 1 as the “primal”. We designed the dual to serve as a method of constructing an upper bound on the optimal value of the primal, so if \( y \) is a feasible solution for the dual and \( x \) is a feasible solution for the primal, then \( 2x_1 + 3x_2 \leq 12y_1 + 3y_2 + 4y_3 \). If we can find two feasible solutions that make these equal, then we know we have found the optimal values of these LP.

In this case the feasible solutions \( x_1 = \frac{1}{2}, x_2 = \frac{5}{4} \) and \( y_1 = \frac{5}{16}, y_2 = 0, y_3 = \frac{1}{4} \) give the same value 4.75, which therefore must be the optimal value.

**Exercise 1:** The dual LP is a minimization LP, where the constraints are of the form \( \text{lhs}_i \geq \text{rhs}_i \). You can try to give lower bounds on the optimal value of this LP by taking positive linear combinations of these constraints. E.g., argue that

\[
12y_1 + 3y_2 + 4y_3 \geq 4y_1 + 2y_2 + 3y_2 \geq 2
\]

(since \( y_i \geq 0 \) for all \( i \)) and

\[
12y_1 + 3y_2 + 4y_3 \geq 8y_1 + y_2 + 2y_3 \geq 3
\]

and

\[
12y_1 + 3y_2 + 4y_3 \geq \frac{2}{3}(4y_1 + 2y_2 + 3y_2) + (8y_1 + y_2 + 2y_3) \geq \frac{4}{3} + 3 = 4.6666.
\]

Formulate the problem of finding the best lower bound obtained by linear combinations of the given inequalities as an LP. Show that the resulting LP is the same as the primal LP 1.

**Exercise 2:** Consider the LP:

\[
P = \max(7x_1 - x_2 + 5x_3)
\]

s.t. \( x_1 + x_2 + 4x_3 \leq 8 \)

\( 3x_1 - x_2 + 2x_3 \leq 3 \)

\( 2x_1 + 5x_2 - x_3 \leq -7 \)

\( x_1, x_2, x_3 \geq 0 \)

Show that the problem of finding the best upper bound by linear combinations of the constraints can be written as the following dual LP:

\[
D = \min(8y_1 + 3y_2 - 7y_3)
\]

s.t. \( y_1 + 3y_2 + 2y_3 \geq 7 \)

\( y_1 - y_2 + 5y_3 \geq -1 \)

\( 4y_1 + 2y_2 - y_3 \geq 5 \)

\( y_1, y_2, y_3 \geq 0 \)

Also, now formulate the problem of finding a lower bound for the dual LP. Show this lower-bounding LP is just the primal (P).
1.1 The Method

Consider the examples/exercises above. In all of them, we started off with a “primal” maximization LP:

\[
\begin{align*}
\text{maximize} & \quad c^T x \\
\text{subject to} & \quad Ax \leq b \\
& \quad x \geq 0,
\end{align*}
\]

The constraint \( x \geq 0 \) is just short-hand for saying that the \( x \) variables are constrained to be non-negative.\(^2\) And to get the best lower bound we generated a “dual” minimization LP:

\[
\begin{align*}
\text{minimize} & \quad r^T y \\
\text{subject to} & \quad Py \geq q \\
& \quad y \geq 0,
\end{align*}
\]

The important thing is: this matrix \( P \), and vectors \( q, r \) are not just any vectors. Look carefully: \( P = A^T \), \( q = c \) and \( r = b \). The dual is in fact:

\[
\begin{align*}
\text{minimize} & \quad y^T b \\
\text{subject to} & \quad y^T A \geq c^T \\
& \quad y \geq 0,
\end{align*}
\]

And if you take the dual of (5) to try to get the best lower bound on this LP, you’ll get (4). The dual of the dual is the primal. The dual and the primal are best upper/lower bounds you can obtain as linear combinations of the inputs.

The natural question is: maybe we can obtain better bounds if we combine the inequalities in more complicated ways, not just using linear combinations. Or do we obtain optimal bounds using just linear combinations? In fact, we get optimal bounds using just linear combinations, as the next theorems show.

1.2 The Theorems

It is easy to show that the dual (5) provides an upper bound on the value of the primal (4):

Theorem 1 (Weak Duality) If \( x \) is a feasible solution to the primal LP (4) and \( y \) is a feasible solution to the dual LP (5) then

\[ c^T x \leq y^T b. \]

Proof: This is just a sequence of trivial inequalities that follow from the LPs above:

\[ c^T x \leq (y^T A)x = y^T (Ax) \leq y^T b. \]

The amazing (and deep) result here is to show that the dual actually gives a perfect upper bound on the primal (assuming some mild conditions).

\(^2\)We use the convention that vectors like \( c \) and \( x \) are column vectors. So \( c^T \) is a row vector, and thus \( c^T x \) is the same as the inner product \( c \cdot x = \sum c_i x_i \). We often use \( c^T x \) and \( c \cdot x \) interchangeably. Also, \( a \leq b \) means component-wise inequality, i.e., \( a_i \leq b_i \) for all \( i \).
**Theorem 2 (Strong Duality Theorem)** Suppose the primal LP (4) is feasible (i.e., it has at least one solution) and bounded (i.e., the optimal value is not $\infty$). Then the dual LP (5) is also feasible and bounded. Moreover, if $x^*$ is the optimal primal solution, and $y^*$ is the optimal dual solution, then

$$c^T x^* = (y^*)^T b.$$  

In other words, the maximum of the primal equals the minimum of the dual.

Why is this useful? If I wanted to prove to you that $x^*$ was an optimal solution to the primal, I could give you the solution $y^*$, and you could check that $x^*$ was feasible for the primal, $y^*$ feasible for the dual, and they have equal objective function values.

This min-max relationship is like in the case of $s$-$t$ flows: the maximum of the flow equals the minimum of the cut. Or like in the case of zero-sum games: the payoff for the max-min-optimum strategy of the row player equals the (negative) of the payoff of the max-min-optimal strategy of the column player. Indeed, both these things are just special cases of strong duality!

We will not prove Theorem 2 in this course, though the proof is not difficult. But let’s give a geometric intuition of why this is true in the next section.

### 1.3 A Geometric Viewpoint

To give a geometric view of the strong duality theorem, consider an LP of the following form:

$$\begin{align*}
\text{maximize} & \quad c^T x \\ 
\text{subject to} & \quad Ax \leq b
\end{align*}$$

(6)

Given two constraints like $a_1 \cdot x \leq b_1$ and $a_2 \cdot x \leq b_2$, notice that you can add them to create more constraints that have to hold, like $(a_1 + a_2) \cdot x \leq b_1 + b_2$, or $(0.7a_1 + 2.9a_2) \cdot x \leq (0.7b_1 + 2.9b_2)$. In fact, any positive linear combination has to hold.

To get a feel of what this looks like geometrically, say we start with constraints $x_1 \leq 1$ and $x_2 \leq 1$. These imply $x_1 + x_2 \leq 2$ (the red inequality), $x_1 + 2x_2 \leq 3$ (the green one), etc.

In fact, you can create any constraint running through the intersection point $(1, 1)$ that has the entire feasible region on one side by using different positive linear combinations of these inequalities.

Now, suppose you have the LP (6) in $n$ variables with objective $c \cdot x$ to maximize. As we mentioned when talking about the simplex algorithm, unless the feasible region is unbounded (and let’s assume
for this entire discussion that the feasible region is bounded), the optimum point will occur at some vertex \( x^* \) of the feasible region, which is an intersection of \( n \) of the constraints, and have some value \( v^* = c \cdot x^* \).

Consider the \( n \) inequality constraints that define the vertex \( x^* \), say these are

\[
  a_1 \cdot x \leq b_1, \quad a_2 \cdot x \leq b_2, \ldots, \quad a_n \cdot x \leq b_n,
\]

so that for each \( i \in \{1, 2, \ldots, n\} \) the point \( x^* \) satisfies the equalities

\[
  a_1 \cdot x = b_1, \quad a_2 \cdot x = b_2, \ldots, \quad a_n \cdot x = b_n.
\]

Just as in the simple example above, if you take these \( n \) inequality constraints that define the vertex \( x^* \) and look at all positive linear combinations of these, you can again create any constraint you want going through \( x^* \) that has the entire feasible region on one side. One such constraint is \( c \cdot x \leq v^* \). It goes through \( x^* \) (since we have \( c \cdot x^* = v^* \)) and every point in the feasible region is contained in it (since no feasible point has value more than \( v^* \)). So it is possible to create the constraint \( c \cdot x \leq v^* \) using some positive linear combination of the \( a_i \cdot x \leq b_i \) constraints.

Why is this interesting?

We’ve shown a short proof (a “succinct certificate”) that \( x^* \) is optimal. Indeed, if I gave you a solution \( x^* \) and claimed it was optimal for the given constraints and the objective function \( c \cdot x \), it is not clear how I would convince you of \( x^* \)’s optimality. In 2-dimensions I could draw a figure, but in higher dimensions things get more difficult. But we’ve just shown that I can take a positive linear combination of the given constraints \( a_i \cdot x \leq b_i \) and create the constraint \( c \cdot x \leq v^* = c \cdot x^* \), hence showing we can’t do any better.

How do we find this positive linear combination of the constraints? Hey, it’s actually just another linear program. Indeed, suppose we want to find the best possible bound \( c \cdot x \leq v \) for as small a value \( v \) as possible. Say the original LP had the \( m \) constraints

\[
  a_1 \cdot x \leq b_1, \quad a_2 \cdot x \leq b_2, \ldots, \quad a_m \cdot x \leq b_m,
\]

written compactly as \( A \times x \leq b \).

What’s our goal? We want to find positive values \( y_1, y_2, \ldots, y_m \) such that

\[
  \sum_i y_i a_i = c.
\]

From this positive linear combination we can infer the upper bound

\[
  c \cdot x = (\sum_i y_i a_i) \cdot x \leq \sum_i y_i b_i.
\]

And we want this upper bound to be as “tight” (i.e., small) as possible, so let’s solve the LP:

\[
  \min \sum_i b_i y_i \quad \text{subject to} \quad \sum_i y_i a_i = c.
\]

(In matrix notation, if \( y \) is a \( m \times 1 \) column vector consisting of the \( y_i \) variables, then we want to minimize \( y^T b \) subject to \( y^T A = c \).) This is yet again the same process as in the example at the beginning of lecture.
Let us summarize: we started off with the “primal” LP,

\[
\begin{align*}
\text{maximize} & \quad \mathbf{c}^T \mathbf{x} \\
\text{subject to} & \quad A \mathbf{x} \leq \mathbf{b}
\end{align*}
\]

and were trying to find the best bound on the optimal value of this LP. And to do this, we wrote the “dual” LP:

\[
\begin{align*}
\text{minimize} & \quad \mathbf{y}^T \mathbf{b} \\
\text{subject to} & \quad \mathbf{y}^T A = \mathbf{c}^T \\
& \quad \mathbf{y} \geq \mathbf{0}.
\end{align*}
\]

Note that this primal/dual pair looks slightly different from the pair \([4]\) and \([5]\). There the primal had non-negativity constraints, and the dual had an inequality. Here the variables of the primal are allowed to be negative, and the dual has equalities. But these are just cosmetic differences; the basic principles are the same.

2 Example #1: Shortest Paths

Duality allows us to write problems in multiple ways, which gives us power and flexibility. For instance, let us see two ways of writing the shortest s-t path problem, and why they are equal.

Here is an LP for computing an s-t shortest path with respect to the edge lengths \(w(u, v)\):

\[
\begin{align*}
\text{max} & \quad d_t \\
\text{subject to} & \quad d_s = 0 \\
& \quad d_v - d_u \leq \ell(u, v) \quad \forall (u, v) \in E
\end{align*}
\]

The constraints are the natural ones: the shortest distance from \(s\) to \(s\) is zero. And if the \(s-u\) distance is \(d_u\), the \(s-v\) distance is at most \(d_u + \ell(u, v)\) — i.e., \(d_v \leq d_u + \ell(u, v)\). It’s like putting strings of length \(\ell(u, v)\) between \(u, v\) and then trying to send \(t\) as far from \(s\) as possible—the farthest you can send \(t\) from \(s\) is when the shortest \(s-t\) path becomes tight.

Here is another LP that also computes the \(s-t\) shortest path:

\[
\text{LP}_t := \min \sum_e \ell_e y_e
\]

\[
\begin{align*}
\text{subject to} & \quad \sum_{w:(s,w) \in E} y_{sw} = 1 \\
& \quad \sum_{v:(v,t) \in E} y_{vt} = 1 \\
& \quad \sum_{v:(u,v) \in E} y_{uv} = \sum_{w:(v,w) \in E} y_{vw} \quad \forall w \in V \setminus \{s,t\} \\
& \quad y_e \geq 0.
\end{align*}
\]

In this one we’re sending one unit of flow from \(s\) to \(t\), where the cost of sending a unit of flow on an edge equals its length \(\ell_e\). Naturally the cheapest way to send this flow is along a shortest \(s-t\) path length. So both the LPs should compute the same value.
2.1 Duals of Each Other

Take the first LP. Since we’re setting $d_s$ to zero, we could hard-wire this fact into the LP and rewrite it as

$$
\text{max } d_t \\
\text{subject to } d_v - d_u \leq \ell(u, v) \quad \forall (u, v) \in E, s /\in \{u, v\} \\
d_v \leq \ell(s, v) \quad \forall (s, v) \in E \\
d_u \leq \ell(u, s) \quad \forall (u, s) \in E
$$

(12)

How to find an upper bound on the value of this LP? Let us define $E^\text{out}_s := \{(s, v) \in E\}$, $E^\text{in}_s := \{(u, s) \in E\}$, and $E^\text{rest} := E \setminus (E^\text{out}_s \cup E^\text{in}_s)$. For every arc $e = (u, v)$ we will have a variable $y_e$. We want to get the best upper bound on $d_t$ by linear combinations of the the constraints, so we should find a solution to

$$
\sum_{e \in E^\text{rest}} y_{uv}(d_v - d_u) + \sum_{e \in E^\text{out}_s} y_{sv}d_v - \sum_{e \in E^\text{in}_s} y_{us}d_u = d_t
$$

(13)

This is like $y^T A = c$ and we want to

$$
\text{minimize } \sum_{(u, v) \in E} y_{uv} \ell(u, v).
$$

(14)

(This is like $\text{min } y^T b$.) Hey, the objective function (14) is exactly what we want, but what about the craziness in (13)? Let’s see what it is saying. Just collect all copies of each of the variables $d_v$, and it’s saying

$$
\sum_{v \neq s} d_v \left( \sum_{u: (u, v) \in E} y_{uz} - \sum_{w: (v, w) \in E} y_{vw} \right) = d_t.
$$

and since these equalities must hold regardless of the $d_v$ values, this is really the same as

$$
\sum_{u: (u, v) \in E} y_{uv} - \sum_{w: (v, w) \in E} y_{vw} = 0 \quad \forall v \neq \{s, t\}. \\
\sum_{u: (u, v) \in E} y_{u} - \sum_{w: (t, w) \in E} y_{w} = 1.
$$

(15)

So we’ve got the LP with objective function (14) and constraints (15), which is exactly the same alternate LP (10) we wrote earlier!