Approximation Algorithms and Hardness of Approximation March 26, 2013

Lecture 11

Lecturer: Alantha Newman *Scribes: Farah Charab*

1 Linear Programming Duality

In this lecture, we will show how to provide lower (upper) bounds on the value of the optimal solution of a given LP corresponding to a minimization (maximization) problem. Such methods are advantageous for several reasons:

- It is often expensive to compute the exact solution for an LP.
- Recall that the first step in designing an approximation algorithm for a minimization (maximization) problem is to efficiently compute a "tight" lower (upper) bound on the optimal value.
- A feasible solution itself is not a lower (upper) bound on the optimal value of the minimization (maximization) problem.

To demonstrate the idea of duality, we consider the following LP:

$$\begin{array}{lrcl} \min & 7x+3y & & \\ \text{subject to} & x+y & \geq & 2 \\ & 3x+y & \geq & 4 \\ & x,y & \geq & 0 \end{array}$$

Suppose that an oracle tells you that $x = y = 1$ is a feasible solution with the optimal value of 10. For a minimization LP, any feasible solution of the LP provides an upper bound on the optimal value, but the crucial thing is how to provide a good *lower* bound on the optimal value. Specifically, given an optimal solution, can we prove that this is indeed the minimum solution? How do we prove that there is no solution with a smaller value? Consider the following: Multiply the second constraint by 2, and then add it to the first constraint. We get the following:

$$\underbrace{(x+y)}_{\geq 2} + 2\underbrace{(3x+y)}_{\geq 2(4)} = 7x+3y$$
$$\geq 10$$

The above approach shows that 10 is the minimum feasible solution for the considered LP. The method of LP duality generalizes the above approach. The basic idea is to find the multipliers for the constraints that yields the largest lower bound (smallest upper bound) for the minimization (maximization) LP problem. How do we do this? In this lecture, we will show how to use the "dual" of the LP to obtain the desired bounds. In order to present the LP-Duality theorem formally, we will first show how to transform a given "primal" LP to its "dual".

1.1 Primal($\mathcal{P}$) to Dual($\mathcal{D}$)

From now on, we will consider the case in which the primal LP is a minimization problem. The case in which the primal LP is a maximization LP is analogous. The primal LP can be written in the following eqiuvalent forms:

$$\begin{array}{lrcl} \min & c^T x & & \\ \text{subject to:} & \boldsymbol{A}x & \geq & b \\ & x & \geq & 0 \end{array} \quad \rightarrow \quad \begin{array}{lrcll} \min & \sum_{i=1}^{n} c_i x_i & & & \\ \text{subject to:} & \sum_{i=1}^{n} A_{ji} x_i & \geq & b_j & \forall j \leq m \\ & x_i & \geq & 0 & \forall i \leq n \end{array}$$

The Dual LP is obtained as follows:

$$\begin{array}{lrcl} \max & b^T y & & \\ \text{subject to} & \boldsymbol{A^T} y & \leq & c \\ & y & \geq & 0 \end{array} \rightarrow \begin{array}{lrcll} \max & \sum_{j=1}^{m} b_j y_j & & & \\ \text{subject to} & \sum_{j=1}^{m} A_{ji} y_j & \leq & c_i & \forall i \leq n \\ & y_j & \geq & 0 & \forall j \leq m \end{array}$$

1. For each constraint in $\mathcal{P}$, we assign a multiplier variable y_j.
2. For each variable in $\mathcal{P}$, we have a corresponding constraint in $\mathcal{D}$ that upper bounds the value of the coefficient of that variable. (That is, after multiplying each constraint in $\mathcal{P}$ with its corresponding multiplier and then summing all of these constraints, the coefficient corresponding to each variable in $\mathcal{P}$ should be upper bounded by its corresponding coefficient in the primal objective function.)
3. The dual objective function maximizes the lower bound on $\mathcal{P}$ by maximizing $b^T y$.

It might seem that there is no benefit in calculating the solution to the dual LP. That is, if obtaining a lower bound by solving the dual LP requires us to solve an LP, why not simply solve the primal LP? However, note that once we have solved the dual LP, the multipliers obtained (i.e. the y_j values) can be used to prove that an optimal solution to the primal LP is actually optimal. Given these multipliers, the process of checking for optimality could be much faster than (re)solving the primal LP.

Next, we prove the "Weak Duality Theorem" which states that any feasible solution to the dual LP is a lower bound on the optimal value of the corresponding primal LP.

Theorem 1 (Weak Duality) *If x is a primal-feasible solution and y is a dual-feasible solution, then $c^T x \geq b^T y$.*

Proof

$$\begin{aligned} b^T y = \sum_j b_j y_j &\leq \sum_j (\sum_i A_{ji} x_i) y_j \\ &= \sum_i (\sum_j A_{ji} y_j) x_i \\ &\leq \sum_i c_i x_i = c^T x. \end{aligned}$$

■

For completeness, we will state the "Strong Duality Theorem". However, for the purpose of obtaining lower bounds for the approximation algorithms that we will see in the rest of this lecture, Weak Duality suffices.

Theorem 2 (Strong Duality) *If x is an optimal primal-feasible solution and y is an optimal dual-feasible solution, then $c^T x = b^T y$, i.e the primal and the dual have the same optimal objective value.*

2 Dual-Fitting

One way to use the dual in the design of approximation algorithms is the method of *dual fitting*. In this method, we find an *integer* solution x for the primal LP and show that:

$$c^T x \leq \alpha \cdot b^T y, \tag{1}$$

for some dual-feasible solution y. Note that by Weak Duality, $b^T y$ is a lower bound on the value of an optimal solution. Thus, such a solution x would result in an α-approximation algorithm. Suppose we can obtain x and y without solving an LP, and we can show that x and y are feasible for the primal and dual LPs, respectively, and that (1) holds. Then we never have to actually solve an LP! We will demonstrate the method of Dual Fitting using the following examples.

2.1 Unweighted Set Cover

We are given a universe of n elements $\mathcal{U} = \{e_1, e_2 \dots e_n\}$, a collection of subsets of the universe $\mathcal{S} = \{s_1, s_2 \dots s_k\}$. The goal is to find the minimum number of sets required to cover all the elements in the universe. Let us formulate this problem as an LP, and then obtain the dual.
Primal (Covering) LP:

$$\begin{array}{lrcl} \text{min} & \sum_{s \in \mathcal{S}} x_s & & \\ \text{subject to} & \sum_{s: e \in s} x_s & \geq & 1 \quad \forall e \in \mathcal{U} \\ & x_s & \geq & 0 \end{array}$$

Dual (Packing) LP:

$$\begin{array}{lrcl} \text{max} & \sum_{e \in \mathcal{U}} y_e & & \\ \text{subject to} & \sum_{e: e \in S} y_e & \leq & 1 \quad \forall s \in \mathcal{S} \\ & y_e & \geq & 0 \end{array}$$

Recall the greedy algorithm for Set Cover from Lecture 2:

Algorithm "Most Bang for the Buck"
Input: A collection of subsets $\mathcal{S} = \{s_1, s_2 \dots s_k\}$ of the set $\mathcal{U} = \{e_1, e_2 \dots e_n\}$.
Output: A collection of subsets that cover $\mathcal{U}$.

- $C \leftarrow \emptyset$
- **while** $C \neq \mathcal{U}$ **do**
  - Find a set S that maximizes $|S \cap (\mathcal{U} \backslash \mathcal{C})|$
  - Set $p(e) = \frac{1}{|S \cap (\mathcal{U} \backslash \mathcal{C})|}$ for each $e \in S \backslash C$
  - $C \leftarrow C \cup S$
- **end while**

Note that the cost of the set cover (number of sets) produced by the algorithm is a feasible integral solution to the covering LP:

$$P = \sum_{e_i \in \mathcal{U}} p(e_i).$$

This has the same form seen in the objective function of the dual LP, although this particular solution may not be dual-feasible. However, we will show that if we divide the dual by a suitable factor, i.e $\frac{P}{\alpha}$, then this "shrunk dual" is a feasible solution for the dual LP. Thus, we will have shown that our algorithm is an α approximation for the unweighted set cover problem.

Lemma 3 *The vector $\boldsymbol{y} = (y_1 y_2 \dots y_{|\mathcal{U}|})$, where $\forall\ e \in \mathcal{U}$, $y_e = \frac{p(e)}{H_n}$ is a dual feasible solution. (Recall that H_n is the n^{th} harmonic number.)*

Proof To show that $\{y_e\}$ is a feasible solution, we need to show that for each set $s \in \mathcal{S}$, the following holds:

$$\sum_{e_i \in s} y_{e_i} = \sum_{e_i \in s} \frac{p(e_i)}{H_n} \leq 1. \tag{2}$$

Consider a set $s \in \mathcal{S}$ consisting of k elements. Let $e_k, e_{k-1}, \ldots e_1$ be the elements covered by the algorithm in that order, i.e e_1 was covered the last and e_k was covered first.

Claim 4 $p(e_j) \leq \frac{1}{j}$.

Why? Because when element e_j was covered, there existed a set with j uncovered elements, namely s, i.e $|s \cap (\mathcal{U} \backslash \mathcal{C})| = j$. Since the algorithm chooses the set with the most number of uncovered elements, it can not choose a smaller set, and hence the claim follows.

So now, we can upper bound the cost of the elements in the set s as follows:

$$\sum_{e_i \in s} p(e_i) \leq \sum_{i=1}^{k} \frac{1}{i} \leq H_k \leq H_n.$$

Since for any $s \in \mathcal{S} : \sum_{e_i \in s} p(e_i) \leq H_n$, we obtain (2), and the lemma follows. ■

In summary, we have the following theorem:

Theorem 5 *The approximation guarantee of the greedy set cover algorithm is H_n.*

Proof

$$\sum_{e_i \in \mathcal{U}} p(e_i) = \sum_{e_i \in \mathcal{U}} H_n \cdot y_{e_i} = H_n \sum_{e_i \in \mathcal{U}} y_{e_i} \leq H_n \cdot OPT.$$

Since we have show that the values $\{y_{e_i}\} = \{p(e_i)/H_n\}$ are dual-feasible, it follows from Weak Duality that $\sum_{e_i \in \mathcal{U}} y_{e_i} = \sum_{e_i \in \mathcal{U}} p(e_i)/H_n$ is a lower bound on the cost of a set cover. ■

2.2 Vertex Cover

The problem of Vertex Cover is as follows. Given a graph $G = (V, E)$, the goal is to find the minimum subset of vertices such that every edge is covered. The LP for this problem and its dual are formulated below:

Primal (Vertex Cover) LP:

$$\begin{array}{lrcll} \text{min} & \sum_{v \in V} x_v & & & \\ \text{subject to} & x_u + x_v & \geq & 1 & \forall\, (u,v) \in E \\ & x_v & \geq & 0 & \end{array}$$

Dual (Matching) LP:

$$\begin{array}{lrcll} \text{max} & \sum_{e \in E} y_e & & & \\ \text{subject to} & \sum_{e: v \in e} y_e & \leq & 1 & \forall\, v \in V \\ & y_e & \geq & 0 & \end{array}$$

Let $\mathcal{M}$ be a maximal matching of the graph G, and let x_{ij} be defined as follows:

$$x_{ij} = \begin{cases} 1 & \text{if } e = (i,j) \in \mathcal{M} \\ 0 & \text{o.w} \end{cases}$$

Now we construct a vertex cover as follows. For all $(i,j) \in \mathcal{M}$, we add vertices i, j to the vertex cover. This set covers all edges of the graph, since $\mathcal{M}$ was chosen to be maximal. Hence, the cost of vertex cover is:

$$cost(VC) = 2 \cdot \sum_{i,j \in E} x_{ij} = 2 \cdot |\mathcal{M}|.$$

Since $\mathcal{M}$ is a maximal matching, $\{x_{ij}\}$ is dual feasible. Hence, by Weak Duality, $|\mathcal{M}|$ is a lower bound for the vertex cover LP. Thus, we obtain a 2-approximation algorithm for the vertex cover problem.

2.3 Feedack Arc Set on Tournaments

Let $G = (V, A)$ be a tournament, which is a complete graph in which each edge is oriented in one direction. The goal of the Feedback Arc Set Problem is to find a minimum cardinality subset of arcs $S \subset A$ such that $G' = (V, A \setminus S)$ is acyclic. The primal LP and its dual are shown below. Note that $\mathcal{C}$ is the set of directed cycles in G.

Primal (Covering) LP:

$$\begin{array}{lrcl} \min & \sum_{(i,j)\in A} x_{ij} & & \\ \text{subject to:} & \sum_{(i,j)\in c} x_{ij} & \geq & 1, \quad \forall\, c \in \mathcal{C} \\ & x_{ij} & \geq & 0 \end{array}$$

Dual (Packing) LP:

$$\begin{array}{lrcl} \max & \sum_{c\in\mathcal{C}} y_c & & \\ \text{subject to:} & \sum_{c:(i,j)\in c} y_c & \leq & 1, \quad \forall\, (i,j) \in A \\ & y_c & \geq & 0 \end{array}$$

We consider the following algorithm for the Feedback Arc Set Problem on Tournaments. Note that this problem is equivalent to finding an ordering of the vertices that minimizes the number of backward edges, i.e. the set S comprises the backward edges in the ordering.

ALGORITHM "RANDOMFAS"
Input: G = (V,A).
Output: Ordering of vertices in V.

- Choose $i \in V$ at random.
- if $(j,i) \in A, \;\Rightarrow\; j \to L$
- if $(i,j) \in A, \;\Rightarrow\; j \to R$
- **return** $(RandomFAS(L, A_L), RandomFAS(R, A_R))$

We now prove that RandomFAS is a 3-approximation algorithm for the problem of Feedback Arc Set on Tournaments.

Let T be the set of directed triangles: $\{t \in T : i \to j \to k \to i\}$. Note that $T \subset \mathcal{C}$. Let A_t be the event that one vertex of $t = \{i, j, k\}$ is chosen before triangle t is broken by the algorithm, i.e. when $\{i, j, k\}$ occur in same recursive call. Let p_t be the probability of event A_t. Then, we have:

$$E[\text{cost of the solution}] = E[\text{number of backward edges}] = \sum_{t\in T} p_t.$$

In the next lemma, we will show that setting $y_c = 0$ for all $c \notin T$, and $y_t = \frac{p_t}{3}$ for all $t \in T$ gives a dual-feasible solution.

Lemma 6 *Setting* $y_c = y_t = \frac{p_t}{3}$*, if* $c = t \in T$ *and* 0 *otherwise is dual feasible.*

Proof Let B_e be the event that edge e is backwards in the output ordering. Let $B_e \wedge A_t$ be the event that edge e is backwards due to A_t. For example, suppose vertices i, j, k form triangle t which is yet unbroken when vertex k is chosen as the pivot. Then we say that edge $e = (i, j)$ is backwards due to event A_t. Since, given event A_t, each edge in t is equally likely to be a backwards edge, we have:

$$\begin{aligned} Pr(B_e \wedge A_t) &= Pr(B_e | A_t) Pr(A_t) \\ &= \frac{1}{3} \times p_t \\ &= \frac{p_t}{3}. \end{aligned}$$

Note that for any $t \neq t' \in T$ such that $e \in t$ and $e \in T'$, $B_e \wedge A_t$ and $B_e \wedge A_{t'}$ are disjoint events. Hence, $\sum\limits_{t:e \in t} Pr(B_e \wedge A_t) \leq 1$. This implies that, forall $e \in A$:

$$\sum_{c:e \in c} y_c = \sum_{t:e \in t} \frac{p_t}{3} \leq 1.$$

We can therefore conclude that $\{y_c\}$ is a dual-feasible solution. ■

Thus, we obtain a 3-approximation algorithm for the problem of Feedback Arc Set on Tournaments.

Theorem 7 *The approximation guarantee of RandomFAS is* 3.

Proof

$$E[cost\ of\ solution] = \sum_{t \in T} p_t = \sum_{c \in C} 3y_c = 3 \cdot \sum_{c \in \mathcal{C}} y_c \leq 3 \cdot OPT.$$

Since we have show that the values $\{y_c\} = \{p_t/3\}$ are dual-feasible, it follows from Weak Duality that $\sum_{t \in T} p_t/3 = \sum_{c \in \mathcal{C}} y_c$ is a lower bound on the size of minimum feedback arc set. ■