CS261 Lecture #09

Joshua R. Wang

April 29, 2015

Announcements

- Today: Linear Programming and Duality Examples
- graded HW#1 and solutions available after class

1 Bipartite Matching

Recall [Lecture #07] the min-cost perfect bipartite matching problem, where we are given:

- bipartite graph $G = (A, B, E)$ where $|A| = |B|$ and a perfect matching exists
- edge costs c_e

Goal: find a perfect matching M minimizing $\sum_{e \in M} c_e$.

The Hungarian Algorithm [Kuhn, Munferes] solved this problem in $\tilde{O}(mn)$ time by finding vertex prices p_v such that $p_u + p_v \leq c_{(u,v)} \forall (u, v) \in E$.

We can write the following LP for this problem, with variables x_e:

$$\begin{array}{lll} \text{minimize} & \sum_e c_e x_e & \\ \text{subject to} & \sum_{e \in \delta(a)} x_e = 1 & \forall a \in A \\ & \sum_{e \in \delta(b)} x_e = 1 & \forall b \in B \\ & x_e \geq 0 & \forall e \in E \end{array}$$

Note that a perfect matching M corresponds to an integer solution to this LP; we set $x_e = 1$ if $e \in M$ and $x_e = 0$ otherwise. However, the LP also has fractional solutions; hence we call it a relaxed LP for our problem (we relaxed the constraints).

Lemma 1.1. *Given a feasible solution x, we can recover a perfect matching M whose cost is at most the value of x.*

Proof. Consider a feasible solution x with at least one fractional variable $x_{(a,b)}$. Since $\sum_{e \in \delta(a)} x_e = 1$, we know that there must be another fractional variable adjacent to a, say $X_{(a,b')}$. We repeat this logic, replacing a with b' to get another fractional variable adjacent to b'. Continuing this logic, we must eventually see the same vertex twice (since there are finitely many vertices) and get a cycle of fractional variables. If we increase every other variable by Δ and decrease the rest by Δ, we maintain the feasibility of the solution. However, the objective function value may go up or down, but one direction of increase will not increase the objective function value. We can continue modifying variable weights until a variable hits a value of zero or one.

This process generates another feasible solution with one fewer fractional variable. If we repeatedly apply this transformation, we must eventually end up at a feasible solution with no fractional variables, i.e. a perfect matching M. Each transformation only improved the objective function value, so the final perfect matching has cost at most the cost of x. □

$$\begin{array}{lll} \text{maximize} & \sum_{a \in A} p_a + \sum_{b \in B} p_b & \\ \text{subject to} & p_a + p_b \leq c_{(a,b)} & \forall (a, b) \in E \end{array}$$

If this dual seems familiar, that's because the dual variables correspond *exactly* to the potentials used in the Hungarian algorithm! Strong duality promises that the best fractional matching and the best fractional prices achieve the same value. The Hungarian algorithm shows that the matching and the prices minimized/maximized by integer values.

2 Max-Flow Min-Cut

Recall [Lecture #01] the max-flow problem, where we are given:

- a graph $G = (V, E)$
- a source $s \in V$
- a sink $t \in V$
- edge capacities u_e

Goal: find a feasible flow from s to t of maximum value.

We can formulate max-flow as an LP based on the definition of feasible $s - t$ flow [Exercise #18], which results in a LP which is linear in the size of the maximum flow instance.

However, we will present another formulation which is simpler but larger. Let P be the set of simple paths from s to t; we have one variable x_p for each $p \in P$:

$$\begin{array}{lll} \text{maximize} & \sum_{p \in P} x_p & \\ \text{subject to} & \sum_{p \in P: e \in p} x_p \leq u_e & \forall e \in E \\ & x_p \geq 0 & \forall p \in P \end{array}$$

Since a network may have exponentially many paths from s to t, this formulation has an exponential number of variables. This is okay, however, since we intend to use it for analysis and not for feeding to an LP solver. We first confirm that this alternate formulation solves maximum flow as we originally defined it.

Lemma 2.1. *If we have a feasible flow, we can find a feasible solution to our alternate formulation with the same value. If we have a feasible solution to our alternate formulation, we can find a feasible flow with the same value.*

Proof. Converting a feasible flow f into paths is Problem #01 from the homework.

Suppose we have a feasible solution x. We select $f_e = \sum_{p \in P: e \in p} x_p$, and claim that this is a feasible flow of equal value. Notice that our LP constraint promises that $f_e \leq u_e$, and that $x_p \geq 0$ implies $f_e \geq 0$, so we have satisfied capacity constraints. For conservation constraints, we have:

$$\sum_{e \in \delta^-(v)} f_e = \sum_{p \in P: v \in p} x_p = \sum_{e \in \delta^+(v)} f_e \forall v \in V \setminus \{s, t\}$$

The flow has the same value as x since every $s - t$ path has a single edge exiting s. □

We now write down the dual LP:

$$\begin{array}{lll} \text{minimize} & \sum_{e \in E} u_e y_e & \\ \text{subject to} & \sum_{e \in p} y_e \geq 1 & \forall p \in P \\ & y_e \geq 0 & \forall e \in P \end{array}$$

The dual assigns a weight to each edge, y_e, which we can think of as a "length". The constraints say that along any path, s and t are at distance at least one from each other. In other word, the dual variables express a way of separating s from t. If we think about it for a bit, we can see that the dual is a relaxation of the minimum cut problem.

Lemma 2.2. *Given an $s - t$ cut (A, B), there is a feasible solution y to the dual whose cost equals the capacity of the cut.*

Proof. Let $y_{(u,v)} = 1$ if $u \in A$ and $v \in B$, and $y_{(u,v)} = 0$ otherwise. Notice $\sum_{e \in E} u_e y_e = \sum_{u \in A, v \in B} u_{(u,v)} =$ capacity(A, B). Furthermore, we satisfy the dual LP constraints since every simple $s - t$ path must cross the $s - t$ cut. □

From strong duality, we know that the maximum flow is equal to the minimum value of the dual. From the max-flow min-cut theorem, we know that these must both be equal to the minimum $s - t$ cut. We have shown that we can produce a feasible solution to the dual from a cut. It turns out that we can also go in the other direction; we can produce a cut from a feasible solution to the dual.

Lemma 2.3. *Given a feasible solution y to the dual, we can find an $s - t$ cut (A, B) whose capacity is at most the value of y.*

Proof. We interpret the y_e as edge lengths, and use Dijkstra's to find the distance from s to every other vertex $v \in V$, which we denote $d(v)$. The dual constraints imply $d(t) \geq 1$.

Suppose we choose a threshold T uniformly at random from $[0, 1)$, and let A be the subset of vertices who are at most T from s. Notice for every choice of T, $s \in A$ and $t \in V \setminus A$. Hence, $(A, V \setminus A)$ is always an $s - t$ cut. How much capacity does this cut have?

We can compute the expected capacity of $(A, V \setminus A)$ using linearity of expectation:

$$\mathbb{E}_{T \sim [0,1)} \text{capacity}(A, V \setminus A) = \sum_{(u,v) \in E} u_{(u,v)} Pr[u \in A, v \notin A]$$

What is $Pr[u \in A, v \notin A]$? We need $d(u) \leq T < d(v)$, which occurs with $d(v) - d(u)$ probability. By the triangle inequality, we know that $d(v) \leq d(u) + y_{(u,v)}$, so this is at most $y_{(u,v)}$ probability. *Note: the negative case is covered by the guarantee that y is nonnegative.* Hence:

$$\mathbb{E}_{T \sim [0,1)} \text{capacity}(A, V \setminus A) = \sum_{e \in E} u_e y_e$$

This proves that an average cut we get has capacity at most the value of y. Hence there must be an $s - t$ cut which has capacity at most the value of y. □