COMPSCI 632: Approximation Algorithms September 25, 2017

Lecture 9

Lecturer: Debmalya Panigrahi *Scribe: Brandon Fain*

1 Overview

For the last two lectures, we have been looking at primal dual algorithms with increasing sophistication. In the primal dual framework, we use the linear program of an optimization problem to guide the choices of the algorithm, even though we never explicitly solve the linear program. In the simplest case (e.g., vertex cover) we simply increase dual variables until a constraint becomes tight, and choose the variable in the primal program corresponding to that constraint. Last lecture, we saw how to design more sophisticated primal dual algorithms by being more careful about which dual variable we choose to increase, and about how we choose primal outcomes, culminating in a Lagrangian preserving 3-approximate primal dual algorithm for the facility location problem.

We noted that we could use our algorithm for facility location to solve a relaxed version of the k-median problem. In this lecture, we complete this argument to give a $2(3+\varepsilon)$-approximate primal dual algorithm for the k-median problem. Subsequently, we introduce a new topic: dual fitting. In this technique, the linear program for an optimization problem is used only in the analysis, unlike the primal-dual framework where it guides the algorithm itself. We demonstrate the approach with some simple examples.

2 Primal Dual Algorithms - The k-median Problem

In the k-median problem, we have a set of points $\mathscr{P}$ in some metric space and we wish to open k "centers" and map all points in $\mathscr{P}$ to some open center so as to minimize the total distance from points in $\mathscr{P}$ to their mapped centers. More formally, we can write the optimization problem as an integer program. For all $i \in \mathscr{P}, j \in \mathscr{P}$, let d_{ij} be the distance between i and j on the metric space, and let x_{ij} be an indicator variable equal to 1 if j is mapped to i and 0 otherwise (we will maintain the convention of indexing centers by i and all points by j). Let y_i be an indicator variable equal to 1 if i is opened as a center and 0 otherwise. Then the k-median problem can be written as

$$\begin{aligned}
\text{Minimize } & \sum_{i,j} d_{ij}x_{ij} \\
\text{subject to } & \sum_i y_i \leq k \\
& \sum_i x_{ij} \geq 1 \quad \forall j \\
& x_{ij} \leq y_{ij} \quad \forall i,j \\
& x_{ij}, y_i \in \{0,1\} \quad \forall i,j
\end{aligned}$$

As in general with primal dual algorithms, we will take the linear program relaxation. However, we will *also* take the Lagrangian relaxation, particularly with regard to the first constraint. Let λ be a Lagrangian multiplier (intuitively, a penalty the program pays for violating the constraint). Then we can write the relaxation of the k-median problem as

$$\begin{aligned}\text{Minimize } & \sum_{i,j} d_{ij}x_{ij} + \lambda\left(\sum_i y_i - k\right)\\ \text{subject to } & \sum_i x_{ij} \geq 1 \quad \forall j\\ & x_{ij} \leq y_{ij} \quad \forall i,j\\ & x_{ij}, y_i \geq 0 \quad \forall i,j\end{aligned} \tag{1}$$

We can also take the dual of program 1. Let α_j be the variables corresponding to the first constraints. We can simplify the form of the dual to avoid writing dual variables for the second constraints by noting that given a set of centers, it is always optimal to simply map each point to its closest center. We get

$$\begin{aligned}\text{Maximize } & \alpha_j - \lambda k\\ \text{subject to } & \max\left\{\alpha_j - d_{ij},\, 0\right\} \leq \lambda \quad \forall i,j\\ & \alpha_j \geq 0 \quad \forall j\end{aligned} \tag{2}$$

In the last lecture, we noted that the primal program is just an additive constant away from the facility location problem where each facility has cost λ. We were able to show that if we somehow chose λ such that $\sum_i y_i = k$, then that solution would be a 3-approximate solution to the k-median problem. The algorithmic challenge of this lecture then, is twofold: (i) how do we choose λ, and (ii) what do we do if we never find a solution with exactly k centers?

The first question is simpler to answer. Note that the number of centers our algorithm returns will be non increasing with the value of λ. In particular, when $\lambda = 0$ the algorithm can just take every point as a center, and when $\lambda = \sum_{i,j} d_{ij}$, the algorithm will just take a single center. Combining these properties suggests that we can run a bisection search. Let $\delta > 0$ be a parameter we will choose later. Initialize $\lambda_1 = 0$ and $\lambda_2 = \sum_{i,j} d_{ij}$ and run a bisection search on the number of centers returned by running our facility location approximation algorithm on program 1 until $\lambda_2 - \lambda_1 \leq \delta$.

These final two values of λ give us two integer solutions, S_1, S_2 corresponding to λ_1, λ_2 respectively, where a solution is just a set of centers, and $|S_1| \geq k$, $|S_2| \leq k$. If either inequality is tight, we can just output that solution. If not, we must address our second question. We do so by creating a new fractional solution by mixing S_1 and S_2 in the correct proportions. In particular, we create a new solution of the form:

$$S = \gamma_1 S_1 + \gamma_2 S_2 \text{ where } \gamma_1 = \frac{k - |S_2|}{|S_1| - |S_2|} \text{ and } \gamma_2 = \frac{|S_1| - k}{|S_1| - |S_2|}$$

where we have slightly abused notation (by $\gamma_1 S_1$ we mean to include a γ_1 proportion of the items in S_1). We also mix the dual solutions in the same proportions to get

$$\alpha = \gamma_1 \alpha^1 + \gamma_2 \alpha^2$$

where α^1, α^2 denote the duals corresponding to λ_1 and λ_2 respectively. The following remark is easy to see from the fact that $\lambda_2 \geq \lambda_1$, and so the dual constraints for λ_2 are looser than for λ_1.

Remark 1. *α is a feasible dual solution for the dual program corresponding to λ_2.*

Lemma 1. *Choose $\delta < \frac{\varepsilon \min_{i,j} d_{ij}}{k}$. Then the resulting fractional S generated by the bisection search is a $3(1+\varepsilon)$-approximation.*

Proof. Let $c(S)$ denote the primal cost of a solution S. Then using weak duality we can bound

$$\begin{aligned} c(S_1) &\leq 3\left(\sum_j \alpha_j^1 - \lambda_1 k\right) \\ &= 3\left(\sum_j \alpha_j^1 - \lambda_2 k\right) + 3\left(\lambda_2 - \lambda_1\right)k \\ &\leq 3\left(\sum_j \alpha_j^1 - \lambda_2 k\right) + 3\delta k \end{aligned}$$

Also,

$$c(S_2) \leq 3\left(\sum_j \alpha_j^2 - \lambda_2 k\right)$$

But note that the cost of S is just a mixture of the cost of S_1 and S_2, so

$$\begin{aligned} c(S) &\leq 3\left(\sum_j \alpha_j - \lambda_2 k\right) + 3\delta k \\ &\leq 3OPT_k + 3\delta k \\ &< 3OPT_k + 3\varepsilon \min_{i,j} d_{ij} \end{aligned}$$

where OPT_k is the optimal primal value for k centers, the second line follows from remark 1 and weak duality, and the third line follows from our choice of δ. Noting that $OPT_k \geq \min_{i,j} d_{ij}$ completes the argument. □

Theorem 2. *We can round S to give a $2(3+\varepsilon)$-approximate integral solution to the k-median problem.*

Proof. There are two cases.

Case 1: $\gamma_2 \geq 1/2$**.** This is the easy case. Recall that S_2 is a feasible ($|S_2| \leq k$) integer solution. If the mixture puts high weight on S_2, i.e., when $\gamma_2 \geq 1/2$, just round S by choosing S_2, we no more than double the primal cost for any point (since $\gamma_2 \geq 1/2$), so the theorem follows from Lemma 1 in this case.

Case 2: $\gamma_2 < 1/2$**.** This is the harder case, S_1 need not be feasible, so we will need a more clever rounding algorithm. We will use the following three step scheme:

1. For every center $i \in S_2$ (note that there are no more than k) include the center in S_1 that is closest to i.

2. If we chose fewer than $|S_2|$ centers in step 1, then arbitrarily choose centers in S_1 until we have chosen $|S_2|$ total centers.

3. We have $|S_1| - |S_2|$ centers left in S_1 that we have not picked. Choose $k - |S_2|$ from this set uniformly at random.

We will analyze the cost of our rounding by focusing on an arbitrary client j. Denote by c_j the "cost" incurred by the algorithm because of client j. Let 1 and 2 be j's closest center in S_1 and S_2 respectively. Then with probability at least $\frac{k-|S_2|}{|S_1|-|S_2|} = \gamma_1$, we actually included center 1 during step 3, and so $c_j = d_{1j}$. If 1 is not picked, we must have chosen the closest center to 2 in S_1; call this center i^*. Combining these facts, we can bound c_j using the triangle inequality as

$$\begin{aligned} cost(j) &\leq \gamma_1 d_{1j} + \gamma_2 d_{i^*j} \\ &\leq \gamma_1 d_{1j} + \gamma_2 \left(d_{2i^*} + d_{2j}\right) \text{ [triangle ineq.]} \\ &\leq \gamma_1 d_{1j} + \gamma_2 \left(d_{12} + d_{2j}\right) \text{ [definition of } i^*\text{]} \\ &\leq \gamma_1 d_{1j} + \gamma_2 \left(d_{1j} + d_{2j} + d_{2j}\right) \text{ [triangle ineq.]} \\ &\leq 2\gamma_1 d_{1j} + 2\gamma_2 d_{2j} \text{ [case assumption]} \end{aligned}$$

Thus, if we sum over j, we find that the cost of our rounded solution is no more than $2\gamma_1 c(S_1) + 2\gamma_2 c(S_2) = 2c(S)$. Noting lemma 1 completes the argument for this case. □

Note that in this second case we have a randomized rounding algorithm and have been considering expected cost. This is not necessary, as this rounding procedure can be derandomized.

3 Dual Fitting

When designing primal dual algorithms, we use use the linear programming relaxation of an optimization problem to guide the decisions of an algorithm. Instead, under the dual fitting paradigm, the algorithm is defined independent of any linear program, but we will use that program in our analysis of the approximation factor. In this lecture, we simply introduce the topic with two simple and familiar examples.

3.1 Vertex Cover

In the vertex cover problem, we are given an unweighted and undirected graph $G = (V, E)$ and want to find a minimum size set of vertices $S \subseteq V$ such that every edge $e \in E$ is adjacent to some vertex $u \in S$. Our algorithm is simple: iteratively choose an arbitrary edge, add both of its endpoints to S, and then remove all vertices adjacent to either endpoint.

Theorem 3. *The above greedy algorithm has an approximation factor of 2 for the vertex cover problem.*

Proof. To analyze the algorithm, we write the dual program over edge variables y_e as

$$\begin{aligned} \text{Maximize } & \sum_{e \in E} y_e \\ \text{subject to } & \sum_{e \in E_v} y_e \leq 1 \quad \forall v \in V \\ & y_e \geq 0 \quad \forall e \in E \end{aligned}$$

where E_v is the set of edges incident on vertex v. Now, note that setting $y_e = 1$ for every edge we picked over the course of our algorithm is a feasible dual solution (since we removed all adjacent vertices after each step, we never picked two edges incident on the same vertex). But for every such edge, we only choose two vertices, and the primal cost is simply the number of vertices we choose. By weak duality therefore, this algorithm is a two approximation. □

3.2 Set Cover

We can employ the same technique to solve set cover. In this problem, we have a universe U of n elements, and a set of sets $\mathscr{S} := \{S_1, S_2, \ldots, S_m\}$ where for each $S_i \in \mathscr{S}, S_i \subseteq U$. We want to choose a minimum number of sets in $\mathscr{S}$ so as to "cover" every element in U, meaning that each element in U should be present in at least one set we pick. Our algorithm will again be simple: Iteratively choose the set that maximizes the number of elements it covers that are not already covered by some set we previously chose.

Theorem 4. *The greedy set cover algorithm has an $O(\log(n))$ approximation factor.*

Proof. For the analysis, we write the dual program over element variables y_e as

$$\begin{aligned} \text{Maximize } & \sum_{e \in E} y_e \\ \text{subject to } & \sum_{e \in S} y_e \leq 1 \quad \forall S \in \mathscr{S} \\ & y_e \geq 0 \quad \forall e \in E \end{aligned}$$

The program looks almost identical to the vertex cover dual, but we will need to be slightly more careful in setting the dual variables for the analysis. Every time the algorithm chooses a set S, let n_S be the number of previously uncovered elements in U that the set covers. A natural setting of the dual variables might be, for every element e, to set $y_e := 1/n_S$ for the set S that first covered e. Then $\sum_e y_e$ equals the primal objective, however our dual solution may be infeasible. Thus, we want to bound, for a given S, how large $\sum_{e \in S} y_e$ can be, and if we are successful, we can simply scale down all of the y_e values to get an approximation.

Consider some set S chosen by our algorithm. Suppose w.l.o.g., that the algorithm covers elements in S in the order $e_1, e_2, \ldots, e_{|S|}$. Clearly, $y_{e_1} \leq 1/|S|$. If S first covered e_1, then $y_{e_1} = 1/|S|$, and if another set first covered e_1, it must have covered at least as many new elements as S, and so $y_{e_1} \leq 1/|S|$. We can apply the same argument iteratively to get

$$\begin{aligned} y_{e_1} &\leq \frac{1}{|S|} \\ y_{e_2} &\leq \frac{1}{|S|-1} \\ y_{e_3} &\leq \frac{1}{|S|-2} \\ &\vdots \\ y_{e_{|S|}} &\leq 1 \end{aligned}$$

So summing, we find that $\sum_{e \in S} y_e \leq H_{|S|}$, the $|S|$'th harmonic number. Asymptotically then, $\sum_{e \in S} y_e \leq O(\log(|S|))$. We can thus scale all of the y_e dual variables down by a logarithmic factor in the size of the largest set in $\mathscr{S}$ to get a feasible dual solution yielding a lower bound no more than $O(\log(n))$ off from our primal objective, completing the proof.

□

4 Summary

We concluded our discussion of primal dual algorithms with an in depth look at a primal dual algorithm for the k-median problem. Our algorithm exploited a Lagrangian preserving approximation algorithm for the facility location problem by taking the Lagrangian relaxation of the k-median problem. After arguing that we could appropriately round a fractional solution with small support, we found that this technique provided us with a $2(3+\varepsilon)$-approximation. Subsequently, we introduced the dual fitting technique of analysis, and gave two simple examples: vertex cover and set cover.