IOE 691: Approximation Algorithms Date: 02/15/2017

Lecture Notes: Randomized Rounding

Instructor: Viswanath Nagarajan Scribe: Kevin J. Sung

1 Randomized approximation algorithms

A randomized algorithm is an algorithm that makes random choices. A randomized α-approximation algorithm for a minimization problem is a randomized algorithm that, with probability $\Omega(1/\text{poly})$, returns a feasible solution that has value at most $\alpha \cdot \text{OPT}$, where OPT is the value of an optimal solution.

2 Minimum cut

The *minimum cut* problem is defined as follows.

- Input: A directed graph $G = (V, E)$ with a cost $c_e \geq 0$ associated with each edge $e \in E$, and two vertices $s, t \in V$.
- Output: A subset $F \subseteq E$ such that if the edges in F are removed from G, the resulting graph has no path from s to t. Equivalently, output a subset S that contains s (the set F is then defined by $F = \{(u, v) \in E : u \in S, v \notin S\}$).

2.1 Integer program formulation

The minimum cut problem can be formulated as an integer program as follows:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e x_e \\ s.t. \quad & y_t = 1, \\ & y_s = 0, \\ & y_v \leq y_u + x_{uv}, \qquad && \forall (u, v) \in E, \\ & x_e, y_v \in \{0, 1\}, \qquad && \forall e \in E, v \in V. \end{aligned}$$

The variables have the following interpretation: x_e indicates whether we choose e as part of the cut, and $y_v = 0$ indicates whether, after cutting edges, there is a path from s to v. The inequality constraint expresses the fact that if $(u, v) \in E$ and there is a path from s to u, then there must also be a path from s to v.

2.2 Linear program relaxation and randomized rounding procedure

We can turn this into a linear program by replacing the constraint

$$x_e, y_v \in \{0, 1\}$$

with the constraint

$$0 \le x_e, y_v \le 1.$$

By solving this LP and "rounding" the solution, we can get an approximate solution to the IP, and hence to minimum cut.

Let (x, y) be an optimal solution to the LP, and consider the following rounding procedure:

1. Pick a number $\tau \in (0, 1)$ uniformly at random.
2. Set $S = \{v : y_v < \tau\}$.

Note that this always returns a feasible solution. We will prove that this gives a 1-approximation, i.e., it always returns an optimal solution.

Lemma 2.1 *For all* $e \in E$, $\Pr_\tau[e \text{ is cut}] \le x_e$.

Proof: Let $e = (u, v)$. Then e is cut if and only if $y_u < \tau < y_v$. The probability of this occuring is $y_v - y_u$, and this is at most x_{uv}, by one of the constraints of the LP. ■

Theorem 2.1 *The rounding procedure described above returns an optimal solution.*

Proof: Let α be the optimal value of the IP, i.e., the minimum cost of a cut, and let β be the optimal value of the LP, so that $\beta \le \alpha$. The expected cost of the cut is

$$\sum_{e \in E} c_e \cdot \Pr[e \text{ is cut}] \le \sum_{e \in E} c_e x_e = \beta \le \alpha.$$

where the first inequality comes from the lemma above. On the other hand, the expected cost must be at least α, since any cut has cost at least α. Therefore, the expected cost is equal to α. Since the procedure can never return a cut with cost strictly less than α, it must always return a cut with cost exactly equal to α, i.e., an optimal cut. ■

3 Facility location

The *facility location* problem is defined as follows.

- Input: A set F of m facilities and a set C of n clients, along with a cost $f_i \ge 0$ associated with opening facility i, and a distance $d_{ij} \ge 0$ between each pair of a facility i and client j. The d_{ij} do not necessarily form a metric.
- Output: A subset of facilities $H \subseteq F$ to open that minimizes the cost

$$\sum_{i \in H} f_i + \sum_{j \in C} \min_{i \in H} d_{ij}.$$

3.1 Integer program formulation

The facility location problem can be formulated as an integer program as follows:

$$\begin{aligned}
\min \quad & \sum_{i \in F} f_i x_i + \sum_{i \in F, j \in C} d_{ij} y_{ij} \\
s.t. \quad & \sum_{i \in F} y_{ij} = 1, \qquad \forall j \in C, \\
& y_{ij} \leq x_i, \qquad \forall i \in F, j \in C \\
& x_i, y_{ij} \in \{0, 1\}, \qquad \forall i \in F, j \in C.
\end{aligned}$$

The variables have the following interpretation: x_i indicates whether to open facility i, and y_{ij} indicates whether facility i is assigned to client j, i.e., whether it is the closest open facility. The equality constraint expresses the fact that each client is assigned exactly one facility, and the inequality constraint expresses the fact that a facility must be open to be assigned to a client.

3.2 Linear program relaxation

We can turn this into a linear program by replacing the constraint

$$x_i, y_{ij} \in \{0, 1\}$$

with the constraint

$$x_i, y_{ij} \geq 0.$$

For a solution T, we will denote by $\text{cost}(T)$ the objective value of T.

3.3 The special case of set cover

We first observe that this problem encompasses the set cover problem. Given a set cover instance specified by m sets and n elements, we can create a facility location instance by letting F be the sets and C the elements, setting $f_i = 1$ for all $i \in F$, and setting d_{ij} to be 0 if $j \in i$ and ∞ otherwise. The optimal choice of facilities then corresponds to an optimal set cover. This shows that we cannot hope to get an $o(\log n)$-approximation to the facility location problem. However, we will show how to get a randomized $(2 + 2 \ln n)$-approximation.

We now describe a randomized rounding procedure for the special case of set cover. In this case, the LP looks like

$$\begin{aligned}
\min \quad & \sum_{i \in F} x_i \\
s.t. \quad & \sum_{i \,:\, j \in i} x_i \geq 1, \qquad \forall j \in C, \\
& x_i \geq 0, \qquad \forall i \in F.
\end{aligned}$$

Let x be an optimal solution to this LP, and consider the following rounding procedure: pick each set i independently with probability $\min\{1, \alpha \cdot x_i\}$, where $\alpha = 1 + \ln n$. Let T be the subset of F returned by this procedure.

Observation 3.1 $\mathrm{E}[\mathrm{cost}(T)] \leq \alpha \cdot \mathrm{OPT}^*$, *where* OPT^* *is the optimal value of the LP.*

Proof:

$$\mathrm{E}[\mathrm{cost}(T)] = \mathrm{E}[|T|] = \sum_{i \in F} \Pr[i \text{ is chosen}] \leq \sum_{i \in F} \alpha \cdot x_i = \alpha \cdot \sum_{i \in F} x_i = \alpha \cdot \mathrm{OPT}^*.$$

■

Lemma 3.1 *For any* $j \in C$, $\Pr[j \text{ is not covered by } T] \leq e^{-\alpha}$.

Proof:

$$\Pr[j \text{ is not covered by } T] = \prod_{i\,:\,j \in i} \Pr[i \notin T] = \prod_{i\,:\,j \in i} (1 - \min\{1, \alpha x_i\}).$$

If $\alpha x_i > 1$ for any i, then this is zero, and the lemma is proven. Otherwise,

$$\prod_{i\,:\,j \in i} (1 - \min\{1, \alpha x_i\}) \leq \prod_{i\,:\,j \in i} (1 - \alpha x_i) \leq \prod_{i\,:\,j \in i} e^{-\alpha x_i} = e^{-\alpha \sum_{i:j \in i} x_i} \leq e^{-\alpha}.$$

In the last inequality, we used the LP constraint $\sum_{i:j \in i} x_i \geq 1$. ■

Lemma 3.2 $\Pr[T \text{ is infeasible}] \leq n \cdot e^{-\alpha}$.

Proof: In this case, T being infeasible means it is not a set cover. This follows from the previous lemma and the union bound:

$$\Pr[T \text{ is infeasible}] \leq \sum_{j \in C} \Pr[j \text{ is not covered by } T] \leq n \cdot e^{-\alpha}.$$

■

Let F be the event that the algorithm returns a feasible solution. The above lemma says that $\Pr[\overline{F}] \leq n \cdot e^{-\alpha}$. If we plug in $\alpha = 1 + \ln n$, we find that

$$\Pr[F] = 1 - \Pr[\overline{F}] \geq 1 - \frac{1}{e}. \tag{1}$$

Therefore, with some constant probability, the algorithm returns feasible solution. We now want to show that the feasible solution will also be a good approximation; that is, we want to upper bound

$$\mathrm{E}[\mathrm{cost}(T)|F].$$

We have

$$\mathrm{E}[\mathrm{cost}(T)] = \Pr[F] \cdot \mathrm{E}[\mathrm{cost}(T)|F] + \Pr[\overline{F}] \cdot \mathrm{E}[\mathrm{cost}(T)|\overline{F}],$$

so

$$\Pr[F] \cdot \mathrm{E}[\mathrm{cost}(T)|F] = \mathrm{E}[\mathrm{cost}(T)] - \Pr[\overline{F}] \cdot \mathrm{E}[\mathrm{cost}(T)|\overline{F}] \leq \mathrm{E}[\mathrm{cost}(T)].$$

Therefore,

$$\mathrm{E}[\mathrm{cost}(T)|F] \leq \frac{\mathrm{E}[\mathrm{cost}(T)]}{\Pr[F]} \leq \frac{\alpha \cdot \mathrm{OPT}^*}{1 - 1/e} = \frac{1 + \ln n}{1 - 1/e} \cdot \mathrm{OPT}^*.$$

Now, we can use Markov's inequality to get that

$$\Pr[\mathrm{cost}(T) > 2(1 + \ln n) \cdot \mathrm{OPT}^* | F] \leq \frac{1}{2(1 - 1/e)}, \tag{2}$$

so that

$$\Pr[\text{cost}(T) \leq 2(1+\ln n)\cdot \text{OPT}^*] \geq 1 - \frac{1}{2(1-1/e)}.$$

We can now prove

Theorem 3.1 *The algorithm described above is a randomized* $(2+2\ln n)$*-approximation.*

Proof: Let F be the event that the algorithm returns a feasible solution, and let G be the event that the algorithm returns a $(2+2\ln n)$-approximation. We will show that the event $F \cap G$ occurs with constant probability. From (1) and (2), we have

$$\Pr[F] \geq 1 - \frac{1}{e}$$

and

$$\Pr[G|F] \geq 1 - \frac{1}{2(1-1/e)}.$$

Therefore,

$$\Pr[F \cap G] = \Pr[F]\cdot \Pr[G|F] \geq \left(1-\frac{1}{e}\right)\left(1-\frac{1}{2(1-1/e)}\right) = \frac{1}{2} - \frac{1}{e}.$$

■

3.4 The general case

We now return to the general facility location problem and show that we can get a $(2+2\ln n)$-approximation. Let x be an optimal solution to the LP for the general problem. Consider the following rounding procedure:

1. For each $i \in F$, independently pick a number $\tau_i \in [0,1]$ uniformly at random.
2. Open facility i if and only if $\alpha \cdot x_i > \tau_i$.
3. Assign client j to facility i if and only if $\alpha \cdot y_{ij} > \tau_i$.

Note that since $y_{ij} \leq x_i$, if a client gets assigned to a facility, then that facility was opened.

Let T be the set of facilities returned by this procedure. We now prove analogous observations and lemmas to the ones above for set cover, which will give the result.

Observation 3.2 $\text{E}[\text{cost}(T)] \leq \alpha \cdot \text{OPT}^*$*, where* OPT^* *is the optimal value of the LP.*

Proof:

$$\begin{aligned}
\text{E}[\text{cost}(T)] &\leq \sum_{i\in F} f_i \cdot \Pr[i \text{ is chosen}] + \sum_{i\in F, j\in C} d_{ij}\Pr[j \text{ is assigned to } i] \\
&\leq \sum_{i\in F} f_i \cdot \alpha x_i + \sum_{i\in F, j\in C} d_{ij}\cdot \alpha y_{ij} \\
&= \alpha \cdot \left(\sum_{i\in F} f_i x_i + \sum_{i\in F, j\in C} d_{ij} y_{ij}\right) \\
&= \alpha \cdot \text{OPT}^*.
\end{aligned}$$

■

Lemma 3.3 *For any $j \in C$,* $\Pr[j \text{ is not assigned to some } i \in T] \le e^{-\alpha}$.

Proof:

$$\Pr[j \text{ is not assigned to some } i \in T] = \prod_{i \in F} \Pr[j \text{ is not assigned to } i].$$

If $\alpha y_{ij} > 1$ for any i, then the product is zero, and the lemma is proven. Otherwise,

$$\prod_{i \in F} \Pr[j \text{ is not assigned to } i] \le \prod_{i \in F} (1 - \alpha y_{ij}) \le \prod_{i \in F} e^{-\alpha y_{ij}} = e^{-\alpha \sum_{i \in F} y_{ij}} = e^{-\alpha}.$$

■

Lemma 3.4 $\Pr[\text{some } j \text{ is not assigned to any } i \in T] \le n \cdot e^{-\alpha}$

Proof: This follows from the above lemma and the union bound. ■

As in the discussion of set cover, we can use these results to obtain the following theorem:

Theorem 3.2 *The algorithm described above is a randomized* $(2 + 2 \ln n)$*-approximation.*