11.1 Introduction

Today we're going to begin thinking about *randomized* rounding of LPs by using it for two problems that we've seen before: set cover and facility location.

11.2 Set Cover

Definition 11.2.1 *Given a universe U, collection $\mathcal{S} = \{S_1, \ldots, S_k\}$ with $S_i \subseteq U$ for each $i \in [k]$, and cost function $c : \mathcal{S} \to \mathbb{R}^+$: Construct a collection $T \subset \mathcal{S}$ such that for all $e \in U$ there exists some $S_i \in T$ with $e \in S_i$, which minimizes the total cost $\sum_{S_i \in T} c(S_i)$.*

In other words, find the set of sets in $\mathcal{S}$ that covers all elements in U with minimum cost.

Consider linear programming relaxation (**SC-LP**) below.

$$\begin{aligned}
\text{minimize:} \quad & \sum_{S \in \mathcal{S}} c(S) \cdot x_S && (\textbf{SC-LP}) \\
\text{subject to:} \quad & \sum_{S : e \in S} x_S \geq 1 \quad \forall e \in U && (11.2.1) \\
& 0 \leq x_S \leq 1 \quad \forall S \in \mathcal{S} && (11.2.2)
\end{aligned}$$

It is obvious that this is a valid relaxation: any valid set cover yields an integral solution to the LP of the same cost, and any feasible integral solution to the LP gives a set cover of the same cost.

Now consider the following approximation algorithm: we first solve the LP to get a solution x^*. Let λ be a parameter which we will set later (preview: it's going to be $\Theta(\log n)$). We then set x'_S to be 1 with probability $\min(\lambda x^*_S, 1)$, and 0 otherwise (independently for each $S \in \mathcal{S}$). Equivalently, let T be the collection of sets with $x'_S = 1$. (This is what the algorithm returns).

Lemma 11.2.2 $\mathbf{E}[c(T)] \leq \lambda \cdot LP$*, where LP denotes the cost of the optimal LP solution.*

Proof:

$$\mathbf{E}[c(T)] = \mathbf{E}\left[\sum_{S \in \mathcal{S}} c(S) x'_S\right] = \sum_{S \in \mathcal{S}} c(S) \mathbf{E}\left[x'_s\right] = \sum_{S \in \mathcal{S}} c(S) \cdot \min(\lambda \cdot x^*_S, 1) \leq \lambda \sum_{S \in \mathcal{S}} c(S) x^*_S = \lambda \cdot LP$$

∎

Lemma 11.2.3 *Let $u \in U$. Then* $\mathbf{Pr}[T$ *does not cover* $u] \leq e^{-\lambda}$

Proof:

$$\mathbf{Pr}[u \text{ uncovered}] = \mathbf{Pr}\left[x'_S = 0 \quad \forall S \in \mathcal{S} : u \in S\right] = \prod_{S : u \in S} \mathbf{Pr}\left[x'_S = 0\right],$$

where we used independence for the last equality.

Now we have two cases.

Case 1: There is some $S \in \mathcal{S}$ with $u \in S$ such that $x^*_S \geq 1/\lambda$. Then $x'_S = 1$, so $S \in T$ and u is covered by T. So in this case, $\mathbf{Pr}[u \text{ uncovered}] = 0 \leq e^{-\lambda}$.

Case 2: Otherwise $x^*_S < 1/\lambda$ for all $s \in \mathcal{S}$ with $u \in S$. Then $x'_S = \lambda x^*_S$ for all $S \in \mathcal{S}$ with $u \in S$. Then

$$\prod_{S:u\in S} \mathbf{Pr}[x'_S = 0] = \prod_{S:e\in S} (1 - \lambda x^*_S) \leq \prod_{S:u\in S} (e^{-\lambda x^*_S}) = e^{-\lambda \sum_{S:u\in S} x^*_S} \leq e^{-\lambda}$$

Hence $\Pr[u \text{ uncovered}] \leq e^{-\lambda}$. ■

One quick definition, which we'll use basically throughout the rest of the course: "with high probability" means with probability at least $1 - \frac{1}{n^c}$ for some $c \geq 1$.

Theorem 11.2.4 *Randomized rounding with $\lambda = \Theta(\log n)$ is an $O(\log n)$-approximation: the expected cost is at most $O(\log n) \cdot OPT$, and it returns a feasible solution with high probability.*

Proof: Set $\lambda = c \cdot \ln(n)$. Then Lemma 11.2.2 implies that $\mathbf{E}[c(T)] \leq O(\log(n)) \cdot LP$. Lemma 11.2.3 implies that for every $u \in U$,

$$\mathbf{Pr}[u \text{ uncovered}] \leq e^{-c \cdot \ln(n)} = 1/n^c.$$

So by a simple union bound, $\mathbf{Pr}[T \text{ is not a set cover}] \leq 1/n^{c-1}$. Thus randomized rounding returns a set cover with high probability by setting $c = 2$ (for example). ■

11.3 Discussion about Randomized Bounds

The type of guarantee that we gave for set cover (feasible w.h.p., cost in expectation) is extremely common when designing randomized approximation algorithms. Why is this enough? To some extent, it depends on the problem, but the basic idea is pretty universal: we can use Markov's inequality to bound the probability of being much worse than the expectation, and then repeat enough so that we get a bound on the cost w.h.p. as well. I'm not going to do this super formally, but we know by Markov that the probability that our algorithm costs more than $1 + \epsilon$ times the expected cost is at most $\frac{1}{1+\epsilon}$. If we repeat the algorithm $\ell = O(\log n)$ times, then the probability that we're more than $(1 + \epsilon)$ times the expectation on all of them is at most

$$\left(\frac{1}{1+\epsilon}\right)^{\ell} \leq O\left(\frac{1}{n^c}\right)$$

for any constant c that we want (by increasing the constant in front of $\log n$ in ℓ). Since we're feasible in each of these ℓ iterations with high probability, we get that after the ℓ iterations we have high probability bounds on both the cost and the feasibility.

11.4 UFL

Let's go beyond set cover to give a randomized rounding for UFL. Recall that we gave a 4-approximation last class. Let's do better by using randomization.

First, recall our last algorithm. We solved the UFL LP to get a fractional solution (x, y), and then for every $j \in V$ we defined $\Delta_j = \sum_{i \in V} d(i,j)x_{ij}$ and $B_j = \{i \in V : d(i,j) \leq \alpha\Delta_j\}$ (we set $\alpha = 4/3$).

Our algorithm picked the unassigned j with smallest Δ_j, found the cheapest facility in B_j to open (we called this $a(j)$), and then opened it and assigned j to it. We then looked at all other unassigned j', and if $B_{j'}$ intersected with B_j we assigned j' to $a(j)$ as well. We then repeated this process until all nodes were assigned.

How can we use randomization to do better? We're going to use almost the exact same algorithm, but instead of picking $a(j)$ to be the cheapest facility in B_j, we're going to pick $a(j)$ randomly from B_j, using the LP as a probability distribution. In particular, the *filtered* solution gives a probability distribution with support in B_j. More formally, for all $j \in V$, let $M_j = \sum_{i \in B_j} x_{ij}$. By the LP constraints we know that $M_j \leq 1$, and by Markov's inequality we know that $M_j \geq (\alpha - 1)/\alpha$. When we consider j in the algorithm, we pick $a(j)$ from B_j from the distribution where $i \in B_j$ gets probability x_{ij}/M_j. Clearly this is a valid distribution, since all probabilities are nonnegative and sum to 1. Let $(\hat{x}, \hat{y})$ be the integral solution defined by this rounding, and let S be the set of facilities that we've opened.

So that's the algorithm. Clearly it returns a feasible solution (with probability 1), so we just need to analyze its cost. As in the deterministic case, let $\mathcal{C}$ be the set of all nodes considered by the algorithm in the main loop (i.e., the nodes who cause us to open a new facility). Note that $\mathcal{C}$ has nothing to do with the random choices: we will get the exact same $\mathcal{C}$ as we would in the deterministic case. So as before, if $j, j' \in \mathcal{C}$ then $B_j \cap B_{j'} = \emptyset$. For all $j \in \mathcal{C}$ and $i \in B_j$, let $A(i,j)$ be an indicator random variable which is 1 if $i = a(j)$ and is 0 otherwise.

Let's begin by analyzing the facility opening costs.

Lemma 11.4.1 $\mathbf{E}[F(\hat{x}, \hat{y})] \leq \frac{\alpha}{\alpha-1} F(x,y)$.

Proof:

$$\mathbf{E}[F(\hat{x}, \hat{y})] = \mathbf{E}\left[\sum_{i \in V} f(i)\hat{y}_i\right] = \mathbf{E}\left[\sum_{j \in \mathcal{C}} \sum_{i \in B_j} f(i)A(i,j)\right] = \sum_{j \in \mathcal{C}} \sum_{i \in B_j} f(i)\mathbf{E}[A(i,j)].$$

We have used linearity of expectations here, which also required us to use the fact that $\mathcal{C}$ is actually deterministic (not randomized) and that $B_j \cap B_{j'} = \emptyset$ for all $j \neq j' \in \mathcal{C}$. Now we can use the definition of $A(i,j)$ to continue the proof:

$$\begin{aligned}\mathbf{E}[F(\hat{x}, \hat{y})] &= \sum_{j \in \mathcal{C}} \sum_{i \in B_j} f(i)\frac{x_{ij}}{M_j} \leq \frac{\alpha}{\alpha-1} \sum_{j \in \mathcal{C}} \sum_{i \in B_j} f(i)x_{ij} \leq \frac{\alpha}{\alpha-1} \sum_{j \in \mathcal{C}} \sum_{i \in B_j} f(i)y_i \\ &\leq \frac{\alpha}{\alpha-1} \sum_{i \in V} f(i)y_i = \frac{\alpha}{\alpha-1} F(x,y),\end{aligned}$$

where we used that $x_{ij} \leq y_i$ by the LP, and again used that $B_j \cap B_{j'} = \emptyset$ for all $j \neq j' \in \mathcal{C}$. ∎

It's worth noting that this analysis of the facility opening costs actually gives precisely the same bound as in the deterministic case (loss of $\alpha/(\alpha-1)$). Where we're going to improve is the connection costs, which in the deterministic setting had a loss of 3α.

Lemma 11.4.2 $\mathbf{E}[C(\hat{x},\hat{y})] \le (2\alpha+1)C(x,y)$

Proof: Let $j \in V$. We want to show that $\mathbf{E}[d(j,S)] \le (2\alpha+1)\Delta_j$. We break into two cases depending on whether $j \in \mathcal{C}$.

1. Suppose that $j \in \mathcal{C}$. Then intuitively, we should expect that $\mathbf{E}[d(j,S)] \le \Delta_j$ no matter what α is: choosing from the original distribution defined by the x_{ij} values would have expectation Δ_j by definition, and by defining B_j the way we have we have only shifted probability mass closer to j. More formally:

$$\begin{aligned}
\mathbf{E}[d(j,S)] &= \mathbf{E}\left[\sum_{i\in B_j} d(i,j)A(i,j)\right] = \sum_{i\in B_j} d(i,j)\frac{x_{ij}}{M_j} = \sum_{i\in B_j}\frac{1}{M_j}d(i,j)x_{ij} \\
&\le \sum_{i\in B_j}\left(d(i,j) + \left(\frac{1}{M_j}-1\right)d(i,j)\right)x_{ij} \\
&\le \sum_{i\in B_j} d(i,j)x_{ij} + \sum_{i\in B_j}\alpha\Delta_j\left(\frac{1}{M_j}-1\right)x_{ij} \\
&= \sum_{i\in B_j} d(i,j)x_{ij} + \alpha\Delta_j(1-M_j) \\
&= \sum_{i\in B_j} d(i,j)x_{ij} + \alpha\Delta_j\sum_{i\notin B_j} x_{ij} \\
&\le \sum_{i\in B_j} d(i,j)x_{ij} + \sum_{i\notin B_j} d(i,j)x_{ij} \\
&= \Delta_j.
\end{aligned}$$

2. Now suppose that $j \notin \mathcal{C}$. Then as in the deterministic case, there is some j' with $\Delta_{j'} \le \Delta_j$ so that $B_j \cap B_{j'} \ne \emptyset$ and j was assigned to $a(j')$. Let $i \in B_j \cap B_{j'}$. Then we get that

$$\begin{aligned}
\mathbf{E}[d(j,S)] &\le \mathbf{E}\big[d(j,i) + d(i,j') + d(j',a(j'))\big] \\
&= d(j,i) + d(i,j') + \mathbf{E}\big[d(j',a(j'))\big] \\
&\le \alpha\Delta_j + \alpha\Delta_{j'} + \Delta_{j'} \\
&= (2\alpha+1)\Delta_j,
\end{aligned}$$

where we used the previous case to bound $\mathbf{E}[d(j',a(j'))]$.

Now we can put this together via linearity of expectations: $\mathbf{E}[C(\hat{x},\hat{y})] = \mathbf{E}\Big[\sum_{j\in V} d(j,S)\Big] \le (2\alpha+1)\sum_{j\in V}\Delta_j = (2\alpha+1)C(x,y)$. ■

Theorem 11.4.3 *This algorithm gives a* $2+\sqrt{3} \approx 3.73$*-approximation.*

Proof: The approximation ratio is the maximum of the loss in the facility opening costs and the connection costs:

$$\begin{aligned}
\mathbf{E}[Z(\hat{x},\hat{y})] &= \mathbf{E}[F(\hat{x},\hat{y})] + \mathbf{E}[C(\hat{x},\hat{y})] \\
&\le \frac{\alpha}{\alpha-1}F(x,y) + (2\alpha+1)C(x,y) \\
&\le \max\left(\frac{\alpha}{\alpha-1}, 2\alpha+1\right)(F(x,y)+C(x,y)) \\
&= \max\left(\frac{\alpha}{\alpha-1}, 2\alpha+1\right)Z(x,y).
\end{aligned}$$

To minimize the max, we can set them equal to each other and then solve for α. This gives us

$$\begin{aligned}
&\frac{\alpha}{\alpha-1} = 2\alpha+1 \\
\Leftrightarrow &\alpha = (2\alpha+1)(\alpha-1) = 2\alpha^2-\alpha-1 \\
\Leftrightarrow &2\alpha^2-2\alpha-1 = 0.
\end{aligned}$$

Applying the quadratic formula (and the fact that α is nonnegative), we get that

$$\alpha = \frac{2 \pm \sqrt{4+8}}{4} = \frac{2+2\sqrt{3}}{4} = \frac{1}{2}(1+\sqrt{3}).$$

Thus we get an approximation ratio of

$$2\alpha + 1 = 2 + \sqrt{3}$$

as claimed. ■