Homework 1: Solution

Problem 1

A standard counter example for Set Cover is a collection of R-bit strings $X = \{0,1\}^R$. Notice that $|X| = 2^R$, so $R = \log_2 |X|$. We define the sets $S_i = \{x \in X : x_i = 1\}$ for all $i = 1, \ldots, R$ and $S'_R = \{x \in X : x_R = 0\}$. The optimal solution picks 2 sets, S'_R and S_R to cover all elements in X. Greedy can be fooled to pick $S_1, \ldots, S_R$, which is $\log_2 n$ sets. This shows that Greedy can be a factor of $\frac{1}{2}\log_2 n$ worse than OPT.

The following lemma argues formally that Greedy can be fooled, i.e. after i steps, suppose $S_1, \ldots, S_i$ have been chosen, the the size of $S_j \setminus (\bigcup_{i' \le i} S_{i'})$ is the same as $S'_R \setminus (\bigcup_{i' \le i} S_{i'})$

Lemma 0.1. *For any $i = 1, \ldots, R-1$, define the collection of sets $\mathcal{S}_i = \{S_1, \ldots, S_i\}$. Then $|S_j \setminus \bigcup \mathcal{S}_i| = |S'_R \setminus \bigcup \mathcal{S}_i|$ for all $j > i$. In other words, Greedy may be fooled to pick S_{i+1}.*

Proof. This is a simple probabilistic argument. For $j > i$, notice that $|S_j \setminus \bigcup \mathcal{S}_i|/|X|$ is exactly the probability that a randomly chosen string $x \in X$ belongs to $S_j \setminus \bigcup \mathcal{S}_i$. This is exactly the probability that a randomly chosen string x satisfies "$x_h = 0$ for all $h \in \{1, \ldots, i\}$ and $x_j = 1$". This probability term is $1/2^{i+1}$. Similarly $|S'_R \setminus \bigcup \mathcal{S}_i|/|X|$ is equal to the probability that "$x_h = 0$ for all $h \in \{1, \ldots, i\}$ and $x_R = 0$". This is $1/2^{j+1}$ as well. This implies that $|S'_R \setminus \bigcup \mathcal{S}_i| = |S_j \setminus \bigcup \mathcal{S}_i|$ for all $j > i$. $\square$

Problem 2

There are two solutions that I will be presenting here. First, one can reduce the problem to set cover and use the set cover's $O(\log n)$ approximation algorithm to solve it. The second way is to do it by LP rounding.

Solution 2.1

We show how to create an equivalent set cover instance from the facility location instance. For each facility $i \in F$ and for each subset $D' \subseteq D$, we have a set $S(i, D')$ of cost $w(i, D') = f_i + \sum_{j \in D'} c_{ij}$. This is the cost of opening the facility i and assigning all clients in D' to it. The set cover instance consists of elements D and sets $\{S(i, D')\}_{i \in F, D' \subseteq D}$ where each set $S(i, D')$ has weight $w(i, D')$ and covers elements in D'. Notice that the number of sets is exponential, but we will deal with this later. The following lemma says that we can equivalently solve the problem in this setting.

Lemma 0.2. *Let OPT_{sc} denote the optimal value of the set cover instance. Then $\mathsf{OPT}_{sc} \le \mathsf{OPT}$. Moreover, any solution to the set cover instance can be turned into a solution of the facility location instance of the same or lower cost.*

Proof. Let F^* be the set of facilities opened in the optimal solution. For each facility $i \in F^*$, denote by $D_i \subseteq D$ the clients served by i, so the total cost can be written as

$$\sum_{i \in F^*} (f_i + \sum_{j \in D_i} c_{ij}) = \sum_{i \in F^*} w(i, D_i)$$

This is the total weight of the sets $S(i, D_i)$ for all $i \in F^*$, which is feasible for set cover instance because the sets cover all the elements in D. This implies that $\mathsf{OPT}_{sc} \leq \mathsf{OPT}$.

Now to prove the converse, consider any feasible solution and first observe that for each $i \in F$, the solution would not pick more than one set of the form $S(i, D')$; suppose not, and two sets $S(i, D')$ and $S(i, D'')$ were chosen. We could modify the solution to choose $S(i, D' \cup D'')$ instead.

So we can safely assume that the solution is of the form $\{S(i, D_i)\}_{i \in F'}$. The cost of this solution is $\sum_{i \in F'} (f_i + \sum_{j \in D_i} c_{ij})$. We construct the facility location solution by openning $F' \subseteq F$ and assign all D_i for each $i \in F'$ to i. This will cost exactly the same. □

Finally, we need to show how to greedily select the best set, i.e. the set with minimum ratio $\frac{w(i,D')}{|\tilde{D} \cap S(i,D')|}$ where $\tilde{D}$ denotes the elements that have not yet been covered. Even though we have exponentially many sets, only a small number of sets matters: For each set of the form $S(i, D')$ such that $|D' \cap \tilde{D}| = k$, the best set D' must be the one that takes k clients closest to i in $\tilde{D}$. More formally, if we consider $|\tilde{D} \cap S(i, D')| = k$, the ratio $\frac{w(i,D')}{k}$ only depends on the numerator, which is $f_i + \sum_{j \in D'} c_{ij}$. If we order $\tilde{D} = \{1, \ldots, |\tilde{D}|\}$ such that $c_{i1} \leq c_{i2} \leq \ldots \leq c_{i|\tilde{D}|}$, then the best D' would be $D' = \{1, \ldots, k\}$. For each such k, there are only $|F|$ choices of best (i, D') (i.e. one for each i), and there are only $|D|$ possible values of k.

Solution 2.2

Another solution is by LP rounding. This is, indeed, a bit more complicated, but it's worth knowing this solution. For each facility i, we use variable y_i to indicate whether facility i is open. For each facility $i \in F$ and client $j \in D$, variable x_{ij} denotes whether j is connected to i.

$$
\begin{aligned}
&\text{(LP)} \\
&\quad \min \quad \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} c_{ij} x_{ij} \\
&\quad \text{s.t.} \quad x_{ij} \leq y_i \text{ for all } i \in F, j \in D \\
&\qquad\qquad \sum_{i \in F} x_{ij} = 1 \text{ for all } j \in D \\
&\qquad\qquad y_i, x_{ij} \in [0, 1]
\end{aligned}
$$

For each client j, we can write the *fractional connecting cost of* j as $\mathsf{cost}_j = \sum_{i \in F} x_{ij} c_{ij}$. So the total *connecting cost* is rewritten as $\sum_{j \in D} \mathsf{cost}_j$. Our goal is to ensure that, each client $j \in D$ is connected to some facility i which is not too "far" from j, compared to the cost cost_j. For a subset $F' \subseteq F$ (tentative opening facilities), we say that j is **close** to F if $d(j, F) \leq 2\mathsf{cost}_j$. The following lemma argues that we can compute a cheap F' such that every client is close to F'.

Lemma 0.3. *We can compute, with high probability, a subset F' such that $\sum_{i \in F'} f_i \leq O(\log n) \sum_{i \in F} f_i y_i$ and for each $j \in D$, client j is close to F'.*

Before formally proving the lemma, let us use the lemma to conclude an $O(\log n)$ approximation algorithm. We can invoke the lemma to compute such set F' whose opening cost is $O(\log n)\mathsf{OPT}$. For each $j \in D$, the connecting cost is $d(j, F') \leq 2\mathsf{cost}_j$, so in total we have $\sum_{j \in D} 2\mathsf{cost}_j \leq 2\mathsf{OPT}$. Now it only remains to show the lemma.

Proof. For each client $j \in D$, we define the set of facilities $F_j = \{i : d(i,j) \leq 2\mathsf{cost}_j\}$. This is the set of facilities close to j. By Markov's inequality, $\sum_{i \in F_j} y_i \geq 1/2$, and it is enough to ensure that our set F' satisfies $F' \cap F_j \neq \emptyset$ for all $j \in D$ (because $F' \cap F_j \neq \emptyset$ is the same as saying that j is close to F').

Consider the random experiment: For each $i \in F$, include i into F' with probability y_i.

Claim 0.1. *For each client $j \in D$, the probability that $F' \cap F_j = \emptyset$ is at most $e^{-1/2}$.*

Proof. The event $F' \cap F_j = \emptyset$ happens with probability $\prod_{i \in F_j}(1 - y_i)$ (due to the fact that we sample each i independently). This term is at most $e^{-\sum_{i \in F_j} y_i} \leq e^{-1/2}$, using the identity $1 + \alpha \leq e^{\alpha}$ for all α. □

So we can repeat the experiment $O(\log n)$ times as follows. For $\ell = 1, \ldots, 100 \log n$, the round ℓ of experiment constructs a set F'_ℓ by the above random process. The final solution is $F' = \bigcup_\ell F'_\ell$. Since these experiments are independent, the probability that $F' \cap F_j = \emptyset$ is at most

$$\mathbf{Pr}\left[(\forall \ell) F'_\ell \cap F_j = \emptyset\right] = \mathbf{Pr}\left[F'_\ell \cap F_j = \emptyset\right]^{100 \log n} \leq e^{-50 \log n} \leq 1/n^{10}$$

for each $j \in D$. By union bounds, the probability of event "$(\exists j \in D) F_j \cap F' = \emptyset$" is at most $1/n^9$. This concludes the proof of the lemma.

□

Problem 3

Let $S_1, \ldots, S_k$ be the solution chosen by Greedy, and $S^*_1, \ldots, S^*_{k^*}$ be the solution chosen by optimal. When Greedy chooses set S_i in round i, we will charge the cost of this set to elements in E. The total charge will be at least k, thus bounding the cost of our greedy solution.

Now we define the charging scheme that bounds the cost. Let $\tilde{S}^{(i)}$ be the uncovered elements before round i. When S_i is chosen, the cost (of 1) is distributed over elements in $\tilde{S}^{(i)} \cap S_i$ equally, and this ensures that each element is only charged once.

Lemma 0.4. *For each S^*_j that is not chosen by greedy, the total charge to the elements in S^*_j is at most H_s.*

Proof. Define $n_i = |S^*_j \cap \tilde{S}^{(i)}|$, so the total charge in round i is done to $(n_i - n_{i+1})$ elements. Since the set S^*_j is also considered by greedy, we know that greedy in round i must cover at least n_i elements. In other words, the charge per element in round i is at most $1/n_i$, and therefore the total charge to set S^*_j in round i is at most $(n_i - n_{i+1})/n_i \leq \sum_{i'=n_{i+1}+1}^{n_i} \frac{1}{i'}$. Combining the contribution from all rounds, we get H_s. □