4.1 Set Cover

- Valid instances : Universe U, $|U| = n$. Family of sets $F = \{S_1, \ldots, S_m\}$, $S_i \subseteq U$ for all i.
- Feasible solutions : A set $I \subseteq [m]$ such that $\bigcup_{i \in I} S_i = U$.
- Objective function : Minimizing $|I|$.
- Greedy algorithm : In each iteration, pick a set which covers most uncovered elements, until all the elements are covered.

Theorem 4.1.1 *If OPT contains k sets, the greedy algorithm uses $\leq k(1 + \ln \frac{n}{k})$ sets.*

Proof: Let I_t be the sets selected by the greedy algorithm up to t iterations. Let n_t be the number of uncovered elements at iteration t. Then $n_t = n - |\bigcup_{i \in I_t} S_i|, n_0 = n, I_0 = \varnothing$.

We claim that:

Claim 4.1.2 $n_t \leq (1 - \frac{1}{k}) n_{t-1}$

Proof: Let $J_t = U \backslash (\bigcup_{i \in I_t} S_i)$, then OPT covers J_{t-1} with $\leq k$ sets.

Because $|J_{t-1}| = n_{t-1}$, we know that OPT covers n_{t-1} uncovered elements with $\leq k$ sets. Therefore there exists a set in OPT which covers at least $\frac{n_{t-1}}{k}$ uncovered elements.

Because the greedy algorithm always chooses the set which covers most uncovered elements, the greedy algorithm covers at least $\frac{n_{t-1}}{k}$ uncovered elements at iteration t.

Therefore $n_t \leq n_{t-1} - \frac{n_{t-1}}{k} = (1 - \frac{1}{k}) n_{t-1}$ ■

Now, by induction, $n_t \leq (1 - \frac{1}{k})^t n$.

Consider $t = k \ln \frac{n}{k}$,

$$n_t \leq \left(1 - \frac{1}{k}\right)^{k \ln \frac{n}{k}} n \leq e^{-\ln \frac{n}{k}} \cdot n \leq \frac{k}{n} \cdot n = k$$

The greedy algorithm covers the remaining k elements using at most k sets, so the greedy algorithm uses at most $k + k \ln \frac{n}{k} = k(1 + \ln \frac{n}{k})$ sets pverall. ■

Is the analysis tight? Yes, here is example:

$U = \{a_1, \ldots, a_n, b_1, \ldots, b_n, c_1, \ldots, c_n\}$, $S_1 = \{a_1, \ldots, a_n\}$, $S_2 = \{b_1, \ldots, b_n\}$, $S_3 = \{c_1, \ldots, c_n\}$, $S_i' = \{a_j, b_j, c_j \mid \frac{n}{2^i} < j \leq \frac{n}{2^{i-1}}\}$, $i = 1, \ldots, \log n + 1$.

The greedy algorithm will choose all the set S'_i, $i = 1, \ldots, \log n + 1$, since each one covers exactly half of the remaining elements. But the optimal solution is S_1, S_2, S_3.

Is there any better algorithm? No, due to a recent result of Dinur and Steurer [DS14]:

Theorem 4.1.3 *Unless $P = NP$, there is no $C \cdot \ln n$-approx for set cover problem with constant $C < 1$.*

[F98] shows a weaker result: Unless $NP \subseteq DTIME(n^{poly \log n})$, there is no $C \cdot \ln n$-approx for set cover problem with constant $C < 1$.

4.2 Weighted Set Cover

- Valid instances : Universe U, $|U| = n$. Family of sets $F = \{S_1, \ldots, S_m\}$, $S_i \subseteq U$ for all i. Each set S_i has a cost c_i.
- Feasible solutions : A set $I \subseteq [m]$ such that $\bigcup_{i \in I} S_i = U$.
- Objective function : Minimizing $\sum_{i \in I} c_i$.
- Greedy algorithm : In each iteration, pick a set which maximized $\frac{\text{number of uncovered elements}}{\text{cost of the set}}$, until all the elements are covered.

Theorem 4.2.1 *The greedy algorithm is an $H_n = \Theta(\log n)$-approximation algorithm. Here $H_n = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}$.*

Proof: Let I_t be the sets selected by the greedy algorithm up to t iterations. Let n_t be the number of uncovered elements at iteration t. Let $C^* = \sum_{i \in OPT} c_i$. Then $n_t = n - |\bigcup_{i \in I_t} S_i|, n_0 = n, I_0 = \varnothing$.

We claim that:

Claim 4.2.2 *The greedy algorithm picks a set with density $= \frac{number\ of\ uncovered\ elements}{cost\ of\ the\ set} \geq \frac{n_{t-1}}{c^*}$ on iteration t.*

This claim can be directly derived by the following claim:

Claim 4.2.3 *On iteration t, there exists a set in OPT with $\frac{cost\ of\ the\ set}{number\ of\ uncovered\ elements} \leq \frac{c^*}{n_{t-1}}$.*

Proof: Let $J_t = U \setminus (\bigcup_{i \in I_t} S_i)$, then OPT covers J_{t-1} with cost $\leq c^*$.

Suppose the claim is false. We have:

$$c^* = \sum_{i \in OPT} c_i = \sum_{i \in OPT} \frac{c_i}{|S_i \cap J_{t-1}|} |S_i \cap J_{t-1}| > \sum_{i \in OPT} \frac{c^*}{n_{t-1}} |S_i \cap J_{t-1}| \geq \frac{c^*}{n_{t-1}} |J_{t-1}| = c^*$$

Contradicted, thus proved. ∎

Now, assume the greedy algorithm picks $S'_1, \ldots, S'_k$, then $\frac{c(S'_t)}{|S'_t \cap J_{t-1}|} \leq \frac{c^*}{n_{t-1}}$. Let $x_t = |S'_t \cap J_{t-1}|$ be the number of elements that greedy covers in iteration t. Then $c(S'_t) \leq x_t \frac{c^*}{n_{t-1}}$

$$
\begin{aligned}
c(Greedy) &= \sum_{i=1}^{k} c(S'_i) \leq \sum_{i=1}^{k} x_i \tfrac{c^*}{n_{i-1}} \\
&= x_1 \tfrac{c^*}{n} + x_2 \tfrac{c^*}{n-x_1} + x_3 \tfrac{c^*}{n-x_1-x_2} + \ldots + x_k \tfrac{c^*}{n-x_1-x_2-\ldots-x_{k-1}} \\
&= c^*(\underbrace{\frac{1}{n} + \ldots + \frac{1}{n}}_{x_1} + \underbrace{\frac{1}{n-x_1} + \ldots + \frac{1}{n-x_1}}_{x_2} + \ldots \\
&\quad + \underbrace{\frac{1}{n-x_1-\ldots-x_{k-1}} + \ldots + \frac{1}{n-x_1-\ldots-x_{k-1}}}_{x_k}) \\
&\leq c^*(\tfrac{1}{n} + \tfrac{1}{n-1} + \ldots + 1) = c^* H_n
\end{aligned}
$$

■

Remark: In some scenarios it is natural to consider problem instances where n is small, but m (the number of sets in the family) is extremely large (exponential in n). It is worth noting that in order to function, the greedy algorithm just needs to be able to pick the set of maximum density. Even when m is exponential, sometimes it is reasonable to assume that we can do this, i.e. we can find the set of maximum density in polynomial time despite an exponential number of sets. This is known as a "density oracle", and if one exists then the greedy algorithm still gives an H_n-approximation in polynomial time.

In fact, it suffices to simply be able to approximate the density:

Theorem 4.2.4 *If there exists an α-approximation for the max density problem, then there exists an αH_n-approximation for the original problem.*

4.3 Max k-Cover Problem

This is essentially the maximization version of Set Cover.

- Valid instances : Universe U, $|U| = n$. Family of sets $F = \{S_1, \ldots, S_m\}$, $S_i \subseteq U$ for all i. Integer $k \leq n$.
- Feasible solutions : A set $I \subseteq [m]$ such that $|I| \leq k$.
- Objective function : Maximizing $|\bigcup_{i \in I} S_i|$.
- Greedy algorithm : In each iteration, pick a set which covers most uncovered elements, until k sets are selected.

Theorem 4.3.1 *The greedy algorithm is a $(1 - \frac{1}{e})$-approximation algorithm.*

Proof: Let I_t be the sets selected by the greedy algorithm up to t iterations, $J_t = U \backslash (\bigcup_{i \in I_t} S_i)$. Assume the greedy algorithm picks $S'_1, \ldots, S'_k$. Let $x_t = |S'_t \cap J_{t-1}|$, $z_t = OPT - \sum_{j \leq i} x_j = OPT - |\bigcup_{j \leq i} S_j|$. The key inequality is that $|OPT \backslash \bigcup_{j \leq i} S_j| \geq z_i$.

We claim that:

Claim 4.3.2 $x_{i+1} \geq \frac{z_i}{k}$.

Proof: Because OPT covers at least z_i uncovered elements with k sets, we know that there exists a set which covers at least $\frac{z_i}{k}$ uncovered elements. From the property of the greedy algorithm, $x_{i+1} \geq \frac{z_i}{k}$. ■

We also claim that:

Claim 4.3.3 $z_i \leq (1 - \frac{1}{k})^i OPT$.

Proof: We prove the claim by induction method. The base case is $z_0 \leq OPT$, which is clearly true since $z_0 = OPT$. Now assume that $z_{i-1} \leq (1 - \frac{1}{k})^{i-1} OPT$. Then

$$z_i = z_{i-1} - x_i \leq z_{i-1} - \frac{z_{i-1}}{k} = z_{i-1}\left(1 - \frac{1}{k}\right) \leq \left(1 - \frac{1}{k}\right)^i OPT$$

Therefore proved. ■

Now, we know that:

$$Greedy = \sum_{i=1}^{k} x_i = OPT - z_k \geq OPT - \left(1 - \frac{1}{k}\right)^k OPT \geq OPT - \frac{1}{e} OPT = \left(1 - \frac{1}{e}\right) OPT,$$

which proves the theorem. ■

4.4 Vertex Cover

We can see the vertex cover problem as a special set cover problem: the universe U is the edge set E, and the family of sets is $F = \{S_u \mid u \in V\}$ where $S_u = \{\{u, v\} \mid \{u, v\} \in E\}$. But this view naturally leads to the following question: why does vertex cover have a 2-approximation, when the best possible for set cover is $\ln n$?

Definition 4.4.1 *The frequency of $e \in U$ is $f_e = |\{S \mid S \in F, e \in S\}|$, the number of sets in F that contain e.*

Theorem 4.4.2 *If $\forall e \in U$, $f_e \leq f$, then there is an f-approximation algorithm for set cover problem.*

Algorithm: In each iteration, arbitrarily choose an uncovered element and select all the sets that contain this element. Repeat until all elements covered.

This is an f-approximation for the same reason that our algorithm for vertex cover was a 2-approximation. Informally, for every two elements $e, e' \in U$ considered by this algorithm, there are no sets which cover both e and e' (or else whichever was covered first would have caused this set to be included, so the algorithm would not consider the second element). The OPT has to be at least as large as the number of iterations of this algorithm. On the other hand, in each iteration this algorithm only picks f sets. Hence it includes at most $f \cdot OPT$ sets.

References

DS14 I. Dinur and D. Steurer, Analytical approach to parallel repetition, *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, 2014, pp. 624–633.

F98 U. Feige, A threshold of ln n for approximating set cover, *Journal of the ACM*, 1998, pp. 634–652.