IT584 Approximation Algorithms	January 9, 2024
Lecture 3	
Lecturer: Rachit Chhaya	*Scribe/s: Niraj Agrawal*

1 Set Cover Problem and Greedy Algorithm

In this scribe note, we will explore the Set Cover Problem and analyze the approximation guarantee of the greedy algorithm for solving it.

1.1 Set Cover Problem

Given:

- A finite set of elements $E = \{e_1, e_2, \dots, e_n\}$.
- A collection of subsets $S = \{S_1, S_2, \dots, S_m\}$, where each subset S_j is a subset of E.
- Non-negative weights w_j associated with each subset S_j. These weights signify the cost or importance of including a particular subset in the cover.

Objective: The goal is to find a minimum-weight collection of subsets that covers all elements in E. This can be expressed as finding a set $I = \{1, 2, \dots, k\}$ of indices that minimizes the total weight:

$$\min_I \sum_{j \in I} w_j$$

subject to the constraint that the union of the selected subsets covers all elements in E:

$$\bigcup_{j \in I} S_j = E$$

1.2 Greedy Algorithm

$$I \leftarrow \emptyset \tag{1}$$

$$\hat{S}_j \leftarrow S_j \quad \forall j \tag{2}$$

while I is not a set cover do

$$l \leftarrow argmin_{j:\hat{S}_j \neq} \frac{w_j}{\left|\hat{S}_j\right|} \tag{3}$$

$$I \leftarrow I \cup \{l\} \tag{4}$$

$$\hat{S}_j \leftarrow \hat{S}_j - S_l \quad \forall j \tag{5}$$

A natural approach to solve the Set Cover Problem is the greedy algorithm. The algorithm iteratively selects the subset that covers the maximum number of uncovered elements until all elements are covered.

Useful Facts

1. k^{th} Harmonic number: $H_k = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{k}$
2. For positive numbers $a_1, \dots, a_k$ and $b_1, \dots, b_k$, then

$$\min_{i=1,\dots,k} \frac{a_i}{b_i} \leq \frac{\sum_{i=1}^k a_i}{\sum_{i=1}^k b_i} \leq \max_{i=1,\dots,k} \frac{a_i}{b_i}$$

1.3 Set Cover Proof

Lemma 1: In the k^{th} iteration,

$$\min_{j:\hat{S}_j \neq \emptyset} \frac{w_i}{\left|\hat{S}_j\right|} \leq \frac{OPT}{n_k}$$

Here, n_k is the number of elements uncovered at the start of iteration k.

proof: O - indices of set in the optimal solution
Claim:

$$\min_{j:\hat{S}_j \neq \emptyset} \frac{w_i}{\left|\hat{S}_j\right|} \leq \frac{\sum_{j \in O} w_j}{\sum_{j \in O} \left|\hat{S}_j\right|} = \frac{OPT}{\sum_{i \in O} \left|\hat{S}_j\right|} \leq \frac{OPT}{n_k}$$

We basically mean, $\sum_{j \in O} \left|\hat{S}_j\right| \geq n_k$. Optimal is at least a feasible solution.

Lemma 2: For set j chosen in k^{th} iteration

$$w_j \leq \frac{n_k - n_{k+1}}{n_k} \cdot OPT$$

Proof: Here is the proof.
Let j minimizes the ratio, so, $\frac{w_j}{|s_j|} \leq \frac{OPT}{n_k}$

$$w_j \leq \frac{|\hat{S}_j| OPT}{n_k} = \frac{n_k - n_k + 1}{n_k} OPT \qquad ....(n_{k+1} = n_k - |\hat{S}_j|) \tag{6}$$

The final part of the proof, Given the claimed inequality by Lemma 2, we can prove the theorem.

Let I contain the indices of the sets in our final solution. Then

$$\sum_{j \in I} w_j \leq \sum_{k=1}^{l} \frac{n_k - n_{k+1}}{n_k} \cdot OPT$$

$$\leq OPT \cdot \textstyle\sum_{k=1}^{l} \left(\frac{1}{n_k} + \frac{1}{n_k - 1} + \ldots + \frac{1}{n_{k+1} + 1} \right) - 1 \quad \text{(a)}$$

$$= OPT \cdot \textstyle\sum_{k=1}^{l} \frac{1}{i}$$

$$= H_n \cdot OPT$$

where the inequality (a) follows from the fact that $\frac{1}{n_k} \leq \frac{1}{n_k - i}$ for $0 \leq i < n_k$.

Now, let's formalize the intuition. Denote n_k as the number of elements that remain uncovered at the beginning of the k-th iteration. Assuming the algorithm takes l iterations, we set $n_1 = n$, and $n_{l+1} = 0$. Consider an arbitrary iteration k. Let I_k represent the indices of the sets chosen in iterations 1 through $k-1$. For each $j = 1, \ldots, m$, let $S_j^{\wedge}$ denote the set of elements in S_j that remain uncovered at the start of this iteration.

In other words,

$$S_j^{\wedge} = S_j - \bigcup_{p \in I_k} S_p$$

This expression signifies the set of elements that remain uncovered in S_j after accounting for the subsets S_p chosen in the preceding iterations up to $k-1$.