Max k-coverage

Anil Maheshwari

anil@scs.carleton.ca
School of Computer Science
Carleton University
Canada

Introduction

Max k-Coverage Problem

Input: A universe U of n elements, and a collection of m sets $\mathcal{S} = \{S_1, \ldots, S_m\}$ of U, where each $S_i \subset U$, and an integer parameter $k > 0$.

Problem: Find k sets from $\mathcal{S}$ such that their union has the largest possible cardinality.

A Greedy Algorithm

Greedy Algorithm for max k-coverage problem

Step 1: Set $\mathcal{S}' := \emptyset$.

Step 2: For $i := 1$ to k do

- Pick the set from $\mathcal{S}$ that covers the maximum number of uncovered elements of U.
- Without loss of generality, let that set be X.
- $\mathcal{S}' := \mathcal{S}' \cup X$.

Step 3: Output $\mathcal{S}'$

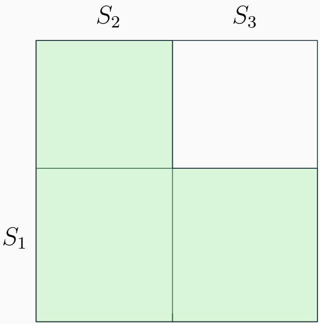
Example: Let n be even and $k = 2$. Let S_1 has $\frac{n}{2} + 2$ elements. Sets S_2 and S_3 each have $n/2$ elements, such that $S_2 \cap S_3 = \emptyset$, $|S_1 \cap S_2| = n/4 + 1$, and $|S_1 \cap S_3| = n/4 + 1$.

Greedy algorithm chooses the sets S_1 and one of S_2 or S_3.

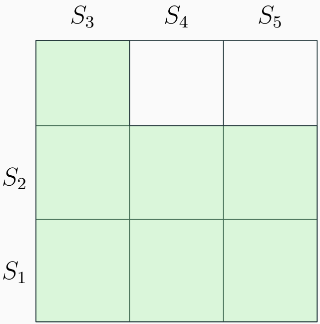
Optimal algorithm will choose S_2 and S_3.

Competitive ratio: $\frac{3n/4+1}{n} \to \frac{3}{4}$, as $n \to \infty$.

Examples for $k = 2, 3$



$$S_1 > S_2 = S_3$$



$$S_1 > S_2 > S_3 = S_4 = S_5$$

Competitive Ratio of Greedy Algorithm

Competitive Ratio of Greedy

Approximation ratio of the greedy algorithm for the max k-coverage problem is $1 - \left(1 - \frac{1}{k}\right)^k \approx 1 - \frac{1}{e}$ for large values of k.

Proof Sketch:

- Assume $i < k$. Suppose greedy algorithm has selected $i - 1$ sets, and in this step it selects the i-th set.

- WLOG, let the selected $i - 1$ sets be $S_1, \ldots, S_{i-1}$, and S_i will be the set selected in the i-th step.

- Let $L_{i-1} = |\bigcup_{j=1}^{i-1} S_j| =$ Total number of elements of U that are covered by the sets $S_1, \ldots, S_{i-1}$.

- Let L^* be the maximum number of elements in the union of an optimal choice of k sets from $\mathcal{S}$.

- We claim that S_i covers at least $\frac{1}{k}(L^* - L_{i-1})$ new elements, i.e., $L_i - L_{i-1} \geq \frac{1}{k}(L^* - L_{i-1})$.

Competitive Ratio of Greedy Algorithm (contd.)

Claim 1

$L_i - L_{i-1} \geq \frac{1}{k}(L^* - L_{i-1})$

Proof:

- Suppose greedy algorithm can pick k sets in this step. Then it will cover $L^* - L_{i-1}$ new elements, as it can pick all the sets of an optimal solution.

- Thus, there is a set among these k sets that has at least $\frac{1}{k}(L^* - L_{i-1})$ new elements.

- The set chosen by greedy algorithm covers $\geq \frac{1}{k}(L^* - L_{i-1})$ new elements.

- Hence, $L_i - L_{i-1} \geq \frac{1}{k}(L^* - L_{i-1})$ $\square$

Competitive Ratio of Greedy Algorithm (contd.)

Claim 2

$L_k \geq \left(1 - \left(1 - \frac{1}{k}\right)^k\right) L^*.$

Proof:

$$
\begin{aligned}
L_k &= (L_k - L_{k-1}) + L_{k-1} \\
&\geq \frac{1}{k}(L^* - L_{k-1}) + L_{k-1} \\
&= \frac{L^*}{k} + \left(1 - \frac{1}{k}\right) L_{k-1} \\
&\geq \frac{L^*}{k} + \left(1 - \frac{1}{k}\right) \frac{L^*}{k} + \left(1 - \frac{1}{k}\right)^2 L_{k-2} \\
&\geq \frac{L^*}{k} \left[1 + \left(1 - \frac{1}{k}\right) + \left(1 - \frac{1}{k}\right)^2 + \cdots + \left(1 - \frac{1}{k}\right)^{k-1}\right] \\
&= \frac{L^*}{k} \left[\frac{1 - \left(1 - \frac{1}{k}\right)^k}{1 - \left(1 - \frac{1}{k}\right)}\right] = \left(1 - \left(1 - \frac{1}{k}\right)^k\right) L^*
\end{aligned}
$$

Competitive Ratio of Greedy Algorithm (contd.)

Theorem

Competitive ratio of the greedy algorithm is $1 - \frac{1}{e}$.

- From previous claim, we have $L_k \geq \left(1 - \left(1 - \frac{1}{k}\right)^k\right) L^*$.
- Thus $\frac{L_k}{L^*} \geq 1 - \left(1 - \frac{1}{k}\right)^k$
- Observe $\lim_{k \to \infty} 1 - \left(1 - \frac{1}{k}\right)^k = 1 - \frac{1}{e}$

Greedy Algorithm for max k-coverage problem

Step 1: Set $\mathcal{S}' := \emptyset$.

Step 2: For $i := 1$ to k do

- Pick the set from $\mathcal{S}$ that covers the maximum number of uncovered elements of U.
- Without loss of generality, let that set be X.
- $\mathcal{S}' := \mathcal{S}' \cup X$.

Step 3: Output $\mathcal{S}'$