Coverage Approximations

1 Set Cover Approximation

SET COVER: Given a universe U of n elements and subsets $S_1, \ldots, S_m \subseteq U$ with costs c_j, find a $T \subseteq \{1, \ldots, m\}$ with $\bigcup_{j \in T} S_j = U$ such that $\sum_{j \in T} c_j$ is minimized.

Algorithm 1 Greedy Approximation Algorithm for Set Cover

Start with $R = U, T = \emptyset$
while $R \neq \emptyset$ **do**
  Let S_j be a set minimizing $\frac{c_j}{S_j \cap R}$ (i.e., $\frac{\text{cost}}{\text{benefit}}$)
  Set $R = R \setminus S_j, T = T \cup S_j$
end while
Return T

Theorem 1. *This is an $\mathcal{O}(\ln n)$ approximation.*

Proof. Let T^* be the optimal solution and T our final solution. Thus, $OPT = C(T^*) = \sum_{j \in T^*} c_j$, and $C(T) = \sum_{j \in T} c_j$.

Define element costs as follows: suppose we cover element u for the first time with a set S_j, and at that iteration, $|S_j \cap R| = k$. Then, the price $p(u) = \frac{c_j}{k}$.

$$C(T) = \sum_{j \in T} c_j = \sum_{j \in T} \frac{c_j}{k} k$$

$$= \sum_{j \in T} \sum_{\text{newly covered elements } u \text{ by } S_j} p(u)$$

$$= \sum_{u \in U} p(u)$$

Lemma 2. *For all sets S_j, $c_j H_{|S_j|} \geq \sum_{u \in S_j} p(u)$ where H_n is the n^{th} harmonic number.*

Proof. Consider some set $S_j = \{i_1, i_2, \ldots i_l\}$ with $l = |S_j|$ such that our algorithm covers the elements in the order $i_1, i_2, \ldots, i_l$. At the moment the algorithm covered i_t for the first time, none of $i_t, i_{t+1}, \ldots, i_l$ were already covered. S_j was clearly an option to cover all of them and had cost effectiveness $\leq \frac{c_j}{l+1-t}$. The greedy algorithm picked the most cost-effective set, so it achieved cost-effectiveness $\leq \frac{c_j}{l+1-t}$ (i.e., the algorithm could have picked an even cheaper set than S_j, but not a more expensive one). Thus, $p(i_t) \leq \frac{c_j}{l+1-t}$.

$$\sum_{S_j} p(u) = \sum_{t=1}^{l} p(i_t)$$

$$\leq \sum_{t=1}^{l} \frac{c_j}{l+1-t}$$

$$= c_j \sum_{t=1}^{l} \frac{1}{l+1-t}$$

$$= c_j \sum_{t=1}^{l} \frac{1}{t}$$

$$= c_j H_l$$

□

Returning to the proof:

$$OPT = \sum_{j \in T^*} c_j$$

$$\geq \sum_{j \in T^*} \frac{1}{H_{|S_j|}} \sum_{u \in S_j} p(u) \text{ by the lemma}$$

$$\geq \frac{1}{\max_{j \in T^*} H_{|S_j|}} \sum_{j \in T^*} \sum_{u \in S_j} p(u)$$

$$\geq \frac{1}{\max_{j \in T^*} H_{|S_j|}} \sum_{u \in U} p(u) \text{ because } T^* \text{is a cover}$$

$$\geq \frac{1}{\max_{j \in T^*} H_{|S_j|}} C(T)$$

$|S_j| \leq n$, so this is at most an H_n approximation. Since $H_n \approx \ln n$, we have an $\mathcal{O}(\ln n)$ approximation. □

Theorem 3. *Unless* ***P****=****NP****, there is no polytime $\mathcal{O}((1-\mathcal{E})\ln n)$ SET COVER approximation algorithm for any $\mathcal{E} > 0$.*

This is a really long and difficult proof (it would take a semester to teach) from the 1990s according to the probabalistically checkable proofs (PCP) theorem.

2 Maximum Coverage Approximation

MAXIMUM COVERAGE: Given a universe U where each element has a value $v(u)$, subsets $S_1, \ldots, S_m \subseteq U$, and an integer k, find a $T \subseteq \{1, \ldots, m\}$ with $|T| \leq k$ maximizing $\sum_{u \in \bigcup_{j \in T} s_j} v(u)$.

Theorem 4. *This is a $(1 - \frac{1}{e})$ approximation.*

Algorithm 2 Greedy Approximation Algorithm for Maximum Coverage

Start with $R = U, T = \emptyset$
for k iterations $t = 1, 2, \ldots, k$ **do**
    Let S_{j_t} be a set minimizing $\sum_{u \in S_{j_t} \cap R} v(u)$.
    Choose S_{j_t} and set $R = R \setminus S_{j_t}, T = T \cup \{j_t\}$
end for
Return T

Proof. Let T^* be the optimal solution and T_t our solution after t rounds. Thus, $OPT = C(T^*) = \sum_{u \in \bigcup_{j \in T^*} s_j} v(u)$, $U_t = \bigcup_{j \in T_t} S_j$, and $V_t = \sum_{u \in U_t} v(u)$.

Two important observations:

(1). Monotonicity: by adding all of T^* to T_t, we would get a solution of value $\geq OPT$ (i.e., increase our value by at least $OPT - V_t$).

(2). Submodularity/diminishing returns: by adding the best set S_j for $j \in T^*$ to T_t, we increase the value by at least $\frac{1}{k}(OPT - V_t)$ because $|T^*| \leq k$.

Our greedy algorithm maximizes $V_{t+1} - V_t$, so:

$$V_{t+1} - V_t \geq \frac{1}{k}(OPT - V_t)$$

$$V_{t+1} \geq (1 - \frac{1}{k})v_t + \frac{1}{k}OPT$$

Claim 5. $V_t \geq (1 - (1 - \frac{1}{k})^t)OPT$

Proof. By induction.

$$V_0 = 0$$

$$V_{t+1} \geq (1 - \frac{1}{k})V_t + \frac{1}{k}OPT$$

$$\geq^{IH} (1 - \frac{1}{k})(1 - (1 - \frac{1}{k})^t)OPT + \frac{1}{k}OPT$$

$$= ((1 - \frac{1}{k}) - (1 - \frac{1}{k})^t)OPT + \frac{1}{k}OPT$$

$$= (1 - (1 - \frac{1}{k})^{t+1})OPT$$

□

After k iterations:

$$V_k \geq (1 - (1 - \frac{1}{k})^k)OPT$$

$$\geq (1 - \frac{1}{e})OPT$$

□

Theorem 6. *Unless* ***P***=***NP****, there is no polytime* $(1 - \frac{1}{e} - \mathcal{E})$ *MAXIMUM COVERAGE approximation algorithm for any* $\mathcal{E} > 0$.