601.435/635 Approximation Algorithms **Lecturer:** Michael Dinitz
Topic: Max Coverage, k-Center **Date:** 2/1/24
Scribe: Michael Dinitz

4.1 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 4.1.1 *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 4.1.2 $x_{i+1} \geq \frac{z_i}{k}$.

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 4.1.3 $z_i \leq (1 - \frac{1}{k})^i OPT$.

Proof: We prove the claim by induction. 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

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

as claimed. ■

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. ■

4.1.1 Extensions

It turns out that Max k-Cover is a special case of a more general problem of maximizing a submodular function subject to a cardinality constraint. There has been a huge amount of work on submodular optimization, which we won't really have time to get into in this course. But if you're interested, let me know and I can point you in the right direction. There are many reasonable options for course projects here.

4.1.2 Minimum k-Union

You might notice that while Maximum k-Cover is the natural maximization variant of Set Cover, there is a natural minimization variant of Maximum k-Cover *other* than Set Cover: the Minimum k-Union problem, where our goal is to choose k sets in order to *minimize* the size of their union (rather than maximize). It might not be obvious, but this turns out to be a radically different problem, which is significantly more complicated. It is a bit too advanced for this course (or at least the first few weeks of this course), but I am very interested in this problem, and the best known algorithm is due to Eden Chlamtáč, me, and Yury Makarychev from a few years ago:

Theorem 4.1.4 ([CDM17]) *There is an $O(m^{1/4+\epsilon})$-approximation to Minimum k-Union for every constant $\epsilon > 0$, and under plausible (but nonstandard) complexity assumptions there is no $o(m^{1/4})$-approximation.*

4.2 k-Center

Definition 4.2.1 *Given a metric space (V, d) and natural number k, the k-center problem is to select a subset $F \subseteq V$ with $|F| = k$ that minimizes $\max_{u \in V} d(u, F)$.*

Note: In the above, $d(u, F)$ is taken to be $\min_{v \in F} d(u, v)$.

k-Center has applications in operations research and military planning, and admits several variants, including the following:

- k-Median: Input and feasible sets are as above, but uses objective function $\min_{F \subseteq V} \sum_{u \in V} d(u, F)$
- k-Means: Input and feasible sets are as above, but uses objective function $\min_{F \subseteq V} \sum_{u \in V} (d(u, F))^2$
- Facility Location: Feasible sets no longer carry the size restriction $|F| = k$, but each 'center' (element included in F) must be paid for, introducing a tradeoff.

Algorithm 1 A greedy algorithm for k-CENTER

Input: Metric space $(V, d), k \in \mathbb{N}$.
Output: $F \subseteq V, |F| = k$, with minimum max distance to elements of V.

```
F ← {u}, for u ∈ V arbitrary
while |F| < k do
  Let u ∈ V \ F be the element maximizing d(u, F).
  F ← F ∪ {u}
end while
return F
```

Claim 4.2.2 *F is feasible.*

Proof: Clear; the algorithm increases $|F|$ by 1 on each iteration and ends when $|F| = k$. ■

Theorem 4.2.3 *Algorithm 1 is a 2-approximation.*

(Note: Intuitively, we might expect this result because we expect to be able to apply the triangle inequality when working with max distances, and the triangle inequality produces factors of 2.)

Proof of Theorem 4.2.3: Let F denote the output of the greedy algorithm, and let F^* denote the OPT solution. We will prove that for all $u \in V$, $d(u, F) \leq 2 \cdot \max_{v \in V} d(v, F^*) = 2 \cdot OPT$, from which it follows that $\max_{u \in V} d(u, F) \leq 2 \cdot \max_{u \in V} d(u, F^*) = 2 \cdot OPT$, and that we have a 2-approximation.

Definition 4.2.4 *For each $v \in F^*$, let the cluster of v be given by $C(v) := \{u \in V : d(u, v) = d(u, F^*)\}$, where tie cases of the form $d(u, v_1) = d(u, v_2) = d(u, F^*)$ for $v_1, v_2 \in F^*$ are decided by placing u into one of the tied clusters arbitrarily.*

Lemma 4.2.5 *Let $x, y \in C(v)$. Then $d(x, y) \leq 2 \cdot OPT$.*

Proof: By the triangle inequality, we have that $d(x, y) \leq d(x, v) + d(y, v)$; by the definition of $C(v)$ we have that this is equal to $d(x, F^*) + d(y, F^*) \leq 2 \cdot OPT$. ■

Returning to the proof of Theorem 3.1.3, we have two cases:

1. **Case 1:** For all $v \in F^*$, $C(v) \cap F \neq \emptyset$. Let $u \in V$, say with $u \in C(v)$ for $v \in F^*$. Then $F \cap C(v) \neq \emptyset$, so let $w \in C(v) \cap F$. Then $w \in F$, so $d(u, F) \leq d(u, w)$, and $u, w \in C(v)$ gives that $d(u, w) \leq 2 \cdot OPT$ by the lemma. Hence $d(u, F) \leq 2 \cdot OPT$. Note that this case does not use any properties specific to the greedy algorithm.

2. **Case 2:** There exists a $v \in F^*$ for which $C(v) \cap F = \emptyset$. By the pigeonhole principle (using that $|F| = |F^*| = k$), there exists $v' \in F^*$ s.t. $|C(v') \cap F| \geq 2$. So, suppose that $a, b \in C(v') \cap F$, and that a is added to F before b. Let F' give the set of elements added to F up to but not

including b. Now let $u \in V$. Then we have the following series of inequalities:

$$\begin{aligned} d(u,F) &\le d(u,F') && \text{(since } F' \subset F\text{)} \\ &\le d(b,F') \\ &\le d(b,a) && \text{(definition of } F', a\text{)} \\ &\le 2 \cdot OPT && \text{(Lemma 4.2.5)} \end{aligned}$$

The key inequality $d(u,F') \le d(b,F')$ follows from the fact that if u is further from F' than b, the greedy algorithm would select u on the next iteration instead of b.

This exhausts all cases and completes the proof. ■

This leaves the question of whether the analysis above is tight, which may be answered via example:

Claim 4.2.6 *There are metric spaces for which the greedy algorithm returns a solution of value* $2 \cdot OPT$

Proof: Consider a set of 5 collinear vertices spaced at increments of 1 unit of distance, with $k = 2$. The optimal solution selects the second and fourth vertices, which have max distance 1 to all other vertices, but the greedy solution will always leave a vertex at distance 2 from F. ■

To conclude our analysis of k-Center, we answer the question of whether we can beat the constant factor of 2 incurred by the greedy algorithm with a hardness of approximation proof.

Theorem 4.2.7 *If there exists a c-approximation for k-Center for* $c < 2$*, then* $P = NP$.

Proof: In class we will show a reduction from Dominating Set, which is also in the textbook. Here we give an alternate proof via a reduction from Vertex Cover. Recall that in Vertex Cover (decision version), for the input $(G = (V,E), k)$ we output 'Yes' if there exists a vertex cover of size at most k, and otherwise output 'No'. Note that this is an NP-hard problem.

Let $[G = (V,E), k]$ be a VC instance, and let $V' := \{v_e \mid e \in E\}$. We reduce to k-Center on the set $V \cup V'$, with k as provided in the instance, and the metric $d(\cdot,\cdot)$ with distances:

- $d(u,v) = 1$ if $u, v \in V$ and $\{u,v\} \in E$.
- $d(u,v_e) = 1$ if $e = \{u,w\}$ for some w.
- $d(u,\cdot) = 2$ otherwise

To see why the above is a metric, notice in particular that every distance is either 1 or 2, triangle inequality cannot be violated.

Lemma 4.2.8 *G has a vertex cover of size k iff* $(V \cup V', d)$ *has a k-Center solution of value 1.*

Proof: [⇒] Let S be a VC of G, $|S| = k$; we would like to show that S is a solution to k-Center of value 1. To see this, let $u \in V$. Note that in solving Vertex Cover we need never consider isolated vertices, so we can suppose wlog that there exists $v \in V$ such that $\{u,v\} \in E$. Because S is a VC, either u or v must be covered by S. If $u \in S$, $d(u,S) = d(u,u) = 0$. Else if $v \in S$, $d(u,S) \le d(u,v) = 1$. Now let $v_e \in V'$, with $e = \{u,v\} \in E$. Then again either u or v is in S, and thus $d(v_e, S) \leq \min\{d(v_e, u), d(v_e, v)\} = 1$. So every vertex of $V \cup V'$ is within distance 1 from a node in S.

[$\Leftarrow$] Let S be a k-Center solution of value 1. If there exists $v_e \in S \cap V'$, replace it in S by one of its endpoints (i.e. if v_e has $e = \{u, v\}$, add u or v to S and remove v_e), forming a new set $S' \subseteq V$. We would now like to show that S' is a VC with $|S'| \leq k$. To see this, let $e = \{u, v\}$ be an edge. Then $d(v_e, S) \leq 1$, because S was a k-Center solution of value 1. It follows that either $v_e \in S, u \in S,$ or $v \in S$. In all cases, the replacement process above ensures that either u or v is in S', and S' is a VC of G. ■

With this lemma in hand, suppose A is an algorithm which c-approximates k-Center, for $c < 2$. Then an algorithm for VC is given by reducing to k-Center by the steps described above, running A on that instance, and returning 'Yes' if A has value less than 2, and returning 'No' otherwise. If the starting Vertex Cover instance is a YES instance (there is a vertex cover of size at most k), then Lemma 4.2.8 implies that there is a k-Center solution of cost 1, and thus A must return a solution of cost at most $c \cdot 1 = c < 2$ so we will correctly return Yes. On the other hand, if the starting Vertex Cover instance is a NO instance then Lemma 4.2.8 implies that every k-Center solution has cost larger than 1 (and thus equal to 2 since all distances in the instance are either 1 or 2). Since A must return a feasible solution, it returns a value at least 2, so we will correctly answer No.

This completes the proof of the theorem. ■

References

[CDM17] Eden Chlamtáč, Michael Dinitz, and Yury Makarychev. Minimizing the union: Tight approximations for small set bipartite vertex expansion. In *Proceedings of the 2017 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 881–899, 2017.