,8

CMPUT 675: Approximation Algorithms **Fall 2015**

Lecture 7,8 (Sep 22 & 24, 2015):Facility Location, K-Center

Lecturer: Mohammad R. Salavatipour *Scribe: Based on older notes*

7.1 Uncapacitated Facility Location Problem

Given a metric graph $G = (V, E)$. There are a set of clients $D \subseteq V$, each having a demand to be served and a set of facilities $F \subseteq V$, each having an opening cost f_i. Note that G is a weighted graph and the edges c_{ij} denote the cost of going from j to i. i.e: if the client at j wants to get service at a facility located at i. The cost functions c_{ij} are known to satisfy triangle inequality. Now we would like to open facilities at a set of locations $F' \subseteq F$ to meet the demands of the clients, keeping in mind that the total cost, $\sum\limits_{i \in F'} f_i + \sum\limits_{j \in D} (min_{i \in F'} c_{ij})$ has to be minimized.

We are going to develop an IP/LP formulation for the problem of metric uncapacitated facility location. Let us declare the variables $y_i \in \{0, 1\}$ for each facility $i \in F$ that denotes if any facility i has been opened. Another binary variable x_{ij} indicates if client j is served by facility i. The following will be the LP formulation for the primal.

$$\begin{aligned} minimize \sum_i f_i y_i + \sum_{i,j} c_j x_{ij} & \\ subject\ to \sum_i x_{ij} = 1 & \qquad \forall j \in D, \\ y_i - x_{ij} \geq 0 & \qquad \forall j \in D, i \in F \\ x_{ij}, y_i \geq 0 & \qquad \forall j \in D, i \in F \end{aligned} \tag{7.1}$$

The dual for this problem can be formulated as:

$$\begin{aligned} maximize \sum_j v_j & \\ subject\ to \sum_j w_{ij} \geq f_i & \qquad \forall i \in F, \\ v_j - w_{ij} \leq c_{ij} & \qquad \forall j \in D, i \in F \\ w_{ij}, v_j \geq 0 & \qquad \forall j \in D, i \in F \end{aligned} \tag{7.2}$$

Here, v_j can be assumed to be the total cost the client j is charged, $j \in D$, which includes w_{ij}, the cost contributed by client j to open the facility i, $i \in F, j \in F$.

Lemma 7.1 *If (x^*, y^*) and (v^*, w^*) are the optimal solutions to the primal and dual problems respectively, then $x^*_{ij} > 0$ implies $c_{ij} \leq vj$.*

Proof: The proof can be derived from the complementary slackness condition: $x^*_{ij} > 0$ *implies* $v_j - w_{ij} = c_{ij}$. Since $w_{ij} \geq 0$, we have $c_{ij} \leq vj$. ■

Let (x^*, y^*) be an optimal solution to the primal LP. For each client $j \in D$, we define the first neighborhood of j as $N(j) = \{i \in F | x^*_{ij} > 0\}$ and the second neighborhood of j as $N^2(j) = \{k \in D | \exists i \in N(j); x^*_{ik} > 0\}$. We also define $C_j = \sum\limits_{i,j} c_j x^*_{ij}$ as the total cost for serving the client j in the optimal solution of the LP. We are going to show that the following algorithm is a 3-approximation for the uncapacitated metric facility location problem.

3-Approximation Algorithm for Facility Location

```
Solve the Linear Program. Let $(x^*, y^*)$ and $(v^*, w^*)$ be the optimal solutions.
$C \leftarrow D$
$K \leftarrow 0$
while $C \neq \emptyset$ do
  $K \leftarrow K + 1$
  Choose $j_k \in C$ which mimimizes $v^*_j + C_j$
  Choose $i \in N(j_k)$ with probability $x^*_{ij_k}$
  Assign $j_k$ and all unassigned clients in $N^2(j_k)$ to $i$
  $C \leftarrow C \setminus \{j_k \cup N^2(j_k)\}$
```

Theorem 7.2 *The above algorithm is a 3-approximation algorithm for uncapacitated facility location.*

Proof: Consider an arbitrary iteration k of the above algorithm. The expected cost of opening a facility at this instant is $\sum\limits_{i \in N(j_k)} f_i x^*_{ij_k} \leq \sum\limits_{i \in N(j_k)} f_i y_i$ because of the LP constraint $x_{ij} \leq y_i$.

Moreover, when we pick the client j_k and assign all unassigned clients in $N^2(j_k)$ to a facility $i \in N(j_k)$, we form a partition of the facilities and purge all clients connected to facility i from being parsed in the $(k+1)^{th}$ iteration. Thus, any facility i is chosen at most once in the lifetime of the algorithm. The total expected cost of opening facilities over all iterations is at most:

$$\sum_k \sum_{i \in N(j_k)} f_i y^*_i \leq \sum_{i \in F} f_i y^*_i$$

Now that we have bounded the cost of opening new facilities, let us try to bound the connection costs. Consider an iteration k. The expected cost of assigning j_k to a facility i is $\sum\limits_{i \in N(j_k)} c_{ij_k} x^*_{ij_k} = C_{j_k}$.

For any other client $l \in N^2(j_k)$ that is assigned to the facility i, the expected cost of serving l is at most:

$$c_{hl} + c_{hj_k} + \sum_i + \sum_i c_{ij_k} x^*_{ij_k}$$

Using results from Lemma 7.1, $c_{hl} \leq v^*_l$ and $c_{hj_k} \leq v^*_{j_k}$. Consequently, the cost of serving l,

$$c_{il} \leq v^*_l + v^*_{j_k} + C_{j_k}$$

Since we pick j_k as the minimizer of $v_p^* + C_p$ in Step-6 of the algorithm, $v_{j_k}^* + C_{j_k} \leq v_l^* + C_l$.

Therefore, the expected cost of serving the client l,

$$c_{il} \leq 2v_l^* + C_l$$

Hence, the total expected costs over all such connections is at most:

$$\sum_{i \in F} f_i y_i^* + \sum_{j \in D} (2v_l^* + C_j)$$

Since the value of optimal primal LP is $\sum\limits_{i \in F} f_i y_i^* + \sum\limits_{j \in D} C_j$ and the optimal dual LP has an optimal solution of $\sum\limits_{j \in D} 2v_l^*$, the total expected cost over all connections can be bounded by 3 times OPT.

■

There are many different clustering problems one can consider. Some of the well studied and classical problems are the following. We will focus on the first two and present approximation algorithms for them.

k**-center**:

- **Input:** A metric graph, $G = (V, E)$ with edge costs c_{ij} satisfying triangle inequality and a positive integer $k \geq 1$.
- **Goal:** To find a subset $S \subseteq V$ of k centers to be open ($|S| = k$) and assign each node to the nearest center in S such that minimizes $\max\limits_{v \in V} d(v, S)$, where $d(u, S) = \min\limits_{s \in S} d(u, s)$.

k**-median**:

- **Input:** A metric graph, $G = (V, E)$ with edge costs c_{ij} satisfying triangle inequality and a positive integer $k \geq 1$.
- **Goal:** To find a subset $S \subseteq V$ of k centers to open and assign each node to the neares open center whil minimizing $\sum\limits_{v \in V} d(v, S)$, where $d(u, S) = \min\limits_{s \in S} d(u, s)$.

k**-min-sum-radii**:

- **Input:** A metric graph, $G = (V, E)$ with edge costs c_{ij} satisfying triangle inequality and a positive integer $k \geq 1$.
- **Goal:** To find a subset $S \subseteq V$ of k centers and assign a radius r_i for each center $c_i \in S$ to form a cluster (containing nodes within distance r_i from c_i) which minimizes the sum of radiis of the clusters. Here, radii is defined as the maximum distance between the cluster center and any other vertex in the cluster.

7.2 Greedy Algorithm for k-center

A 2-approximation greedy solution to the k-center problem is as follows:

2-Approximation Algorithm for k-Center

```
1. Start from an arbitrary vertex $v \subset V$; $S \leftarrow \{v\}$
2. while $|S| < k$ do
3.     $u \leftarrow \max_{u} d(u, S)$ – pick the vertex farthest from all vertices in S
4.     $S \leftarrow S \cup \{u\}$
5. return $S$
```

Theorem 7.3 *The above algorithm is a 2-approximation for k-center.*

Proof: Suppose S^* is an optimal k-center for a given graph $G = (V, E)$ and let $V_1^*, V_1^* \cdots V_k^*$ be the clusters associated with this solution. Let r^* be the maximum radius of the k clusters.

Claim 7.4 $\forall u, v \in V_i^*; d(u, v) \leq 2r^*$.

Suppose the solution returned by our algorithm, S, has one vertex from each cluster V_i^*, then it clearly has a cost that is less than $2r^*$. If S does not have one vertex from each V_i^*, then by pegion hole principle, there exists atleast one cluster V_j^* that contains two or more elements of S. Let us call these elements p and p'. Since the maximum radius in the optimum solution is r^* and the vertices p, p' are contained in the same cluster, d(p, p'), the minimum distance between p and p' is at most $2r^*$.

Without loss of generality, let us assume that our algorithm picked p' after p. Since our solution picked p' after p, p' would have been the farthest vertex from any of the cluster centres during the iteration it was picked. Furthermore, as $d(p, p') \leq 2r^*$, the maximum radius of the k cluster, $\max_{v \in V} d(v, S)$ can be at most $2r^*$.

■

Theorem 7.5 *There is no α-approximation algorithm for the k-center problem for $\alpha < 2$ unless $P = NP$.*

Proof: Consider the dominating set problem, which is NP-complete. In the dominating set problem, we are given a graph $G = (V, E)$ and an integer k, and we must decide if there exists a set $S \subseteq V$ of size k such that each vertex is either in S, or adjacent to a vertex in S. Given an instance of the dominating set problem, we can define an instance of the k-center problem by setting the distance between adjacent vertices to 1, and nonadjacent vertices to 2: there is a dominating set of size k if and only if the optimal radius for this k-center instance is 1. Furthermore, any α-approximation algorithm with $\alpha < 2$ must always produce a solution of radius 1 if such a solution exists, since any solution of radius $\alpha < 2$ must actually be of radius 1. So such algorithm would solve the dominating set problem in polytime, which is impossible, unless $P = NP$. ■

7.3 k-median problem

k-median is an important clustering problem that has similarities to both k-center and facility location problem. An instance of this problem is similar to k-center: a complete graph $G = (V, E)$ with metric edge costs $c(e)$ and a positive integer k. The difference is in the objective function: now instead of minimazing maximal distance from nodes to the nearest center ($\max_{v \in V} d(v, S)$) we're minimizing the sum of distances from nodes to the nearest open center ($\min \sum_{v \in V} d(v, S)$).

k-median problem

- Input
  - A complete graph $G = (V, E)$ with set of clients $D \subseteq V$ and facilities $F \subseteq V$
  - c_{ij} is the cost of assigning location j to a facility at location i (and this cost function is metric)
  - k - the maximal number of facilities we can open.
- Goal: find $F' \subseteq F$, where $|F'| \leq k$, to open, s.t. the assignment cost is minimized: $\min \sum_{j \in N} d(j, F') = kmed(F')$.

Without loss of generality, we can assume that $|F'| = k$. As in the k-center problem, we assume that the distance matrix is symmetric, satisfies triangle inequality, and has zeros on the diagonal. We present a local search algorithm for k-median problem with good approximation ratio. For every subset $F' \subseteq F$ we use $kmed(F')$ to denote the cost of the solution if set F' is chosen.

Local search algorithm

1. Start from an arbitrary F' with $|F'| = k$
2. On each iteration see see if swapping a facility in F' with one in $F - F'$ improves the solution
3. Iterate until no single swap yields a better solution

Figure 7.1: Local search algorithm for k-median problem

Theorem 7.6 *If F' is a local optimum and F^* is a global optimum, then* $\text{kmed}(F') \leq 5\text{kmed}(F^*)$

Proof: Our proof is based on "Simpler Analyses of Local Search Algorithms for Facility Location" by Gupta and Tangwongsan (arXiv:0809.255). The proof will focus on constructing a set of special swaps. These swaps will all be constructed by swapping into the solution location i^* in F^* and swapping out of the solution one location i' in F'. Each $i^* \in S^*$ will participate in exactly one of these k swaps, and each $i' \in F'$ will participate in at most 2 of these k swaps. We will allow the possibility that $i^* = i'$, and hence the swap move is degenerate, but clearly such a "change" would also not improve the objective function of the current solution, even if we change the corresponding assignment. For any two points i, j we use $d(i, j)$ to refer to the distance between these two points, i.e. c_{ij}. Let $\phi : F^* \to F$ be a mapping that maps each $f^* \in F^*$ to the nearest facility in F, i.e. $d(f^*, \phi(f^*)) \leq d(f^*, f)$ for all $f \in F'$.

Let $R \subseteq F'$ be those that have at most one $f^* \in F^*$ mapped to them. Now we define a set of k pairs of potential swaps: $S = \{(v, f^*) \subseteq R \times F^*\}$ such that:

1. $\forall f^* \in F^*$, it appears in exactly one pair $(v, f^*) \in S$.

2. each node $r \in R$ with $\phi^{-1}(r) =$ appears in at most two swaps.

3. each nodr $r \in R$ with $\phi^{-1}(r) = f^*$ appears only in one swap.

How to build this set S? for each $r \in R$ with in-degree 1 we add pairs $(r, \phi^{-1}(r))$ to S. Let F_1^* be those of F^* that are matched this way. Other facilities in R have in-degree zero; let us call this set R_0. Note that

$$|F^* \setminus F_1| \leq 2|R_0|.$$

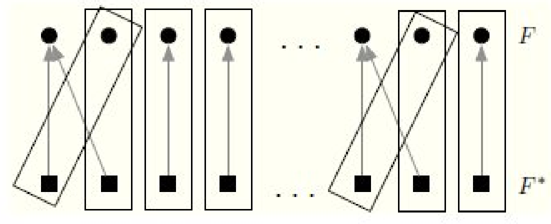

Figure 7.2: An example of mapping $\phi : F^* \to F$

Now we can add other pairs by arbitrarily matching each node of R_0 with at most two in $F^* \setminus F_1^*$.

Observation: For any pair $(r, f^*) \in S$ and $\tilde{f}^* \in F^*$ with $\tilde{f}^* \neq f$: $\phi(\tilde{f}^*) \neq r$.

We use the fact that none of these potential swaps (in S) are improving to derive a bound on the cost of local optimum. Suppose that $\sigma : D \to F'$ and $\sigma^* : D \to F^*$ are mappings of clients to facilities in the local optimum and global optimum, respectively. For each $j \in D$, let $O_j = d(j, F^*) = d(j, \sigma^*(j))$ be the cost of connecting j in the optimum solution and $A_j = d(j, F') = d(j, \sigma(j))$ be its cost in the local optimum. We use $N^*(f^*) = \{j | \sigma^*(j) = f^*, f^* \in F^*\}$ to denote those assigned to f^* in the optimum solution and $N(f) = \{j | \sigma(j) = f, f \in F'\}$ to denote those assigned to f in the local optimum.

Lemma 7.7 *For each swap $(r, f^*) \in S$:*

$$\text{kmed}(F' + f^* - r) - \text{kmed}(F') \leq \sum_{j \in N^*(f^*)} (O_j - A_j) + \sum_{j \in N(r)} 2O_j.$$

Proof: Suppose we do the swap (r, f^*) and let's see how much the cost increases (note that since we are at a local optimum, this must be the case). We can upper bound this by giving a specific assignment of clients to facilities. Clearly the optimum assignment of clients to facilities cannot cost more than this:

- each client of $N^*(f^*)$ is assigned to f^*
- each client $j \in N(r) \setminus N^*(f^*)$ is assigned by the following rule: suppose $\tilde{f}^* = \sigma(j)$; we assign j to $\tilde{f} = \phi(\tilde{f}^*)$. Note that $\tilde{f} \neq r$.
- the assignment of all other clients remain unchanged.

For each $j \in N^*(f^*)$ the change in cost is exactly $O_j - A_j$; summing this over all $j \in N^*(f^*)$ gives the first term on the RHS. For $j \in N(r) \setminus N^*(f^*)$, the change in cost is:

$$\begin{aligned}
d(j, \tilde{f}) - d(j, r) &\leq d(j, \tilde{f}^*) + d(\tilde{f}^*, \tilde{f}) - d(j, r) && \text{using triangle inequality} \\
&\leq d(j, \tilde{f}^*) + d(\tilde{f}^*, r) - d(j, r) && \text{since } \tilde{f} \text{ is closest to } \tilde{f}^* \\
&\leq d(j, \tilde{f}^*) + d(j, \tilde{f}^*) && \text{using triangle inequality} \\
&= 2O_j
\end{aligned}$$

Thus, summing up the total change for all these clients is at most: $\sum_{j \in N(r) \setminus N^*(f^*)} 2O_j \leq \sum_{j \in N(r)} 2O_j$. ■

Now we use this lemma and sum over all pairs $(r, f^*) \in S$. Note that each $f^* \in F^*$ appears exactly once and each $r \in R \subseteq F'$ appears at most twice. Therefore:

$$\begin{aligned}\sum_{(r,f^*)\in S}(\text{kmed}(F'+f^*-r)-\text{kmed}(F')) &\leq \sum_{f^*\in F^*}\sum_{j\in N^*(f^*)}(O_j-A_j)+2\sum_{r\in R}\sum_{j\in N(r)}2O_j\\ &\leq \text{cost}(F^*)-\text{cost}(F')+4\text{cost}(F^*)\end{aligned}$$

This implies that $\text{cost}(F')\leq 5\text{cost}(F^*)$.

■

Note that the running time of this algorithm is not necessarily polynomial. To get polynomial time algorithm we only consider swaps which improve the cost by a factor of at least $(1+\delta)$ for some $\delta>0$. So when the algorithm stops we are in an almost locally optimum solution, i.e. each potential swap can only improve by a factor of smaller than $1+\delta$. Then essentially the same analysis shows that the approximation ratio of the algorithm is at most $5(1+\delta)$ which is $5+\epsilon$ for sufficiently small $\epsilon>0$. If the objective value is M for the optimum solution then the algorithm takes at most $O(\log_{1+\delta}M)$ steps to arrive at a locally optimum solution which is polynomial.

Improvment using t-swaps: A similar anaylsis shows that if one considered all t-swaps (instead of just 1-swaps) for a constant value of t at each step then the local search has a ratio of $3+\frac{2}{t}$.