Approximation Algorithm

Prof. Palash Dey

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

Week – 02

Lecture 10

Lecture 10 : The k-Center Problem

Welcome. So, in the last class we have seen how greedy algorithms can be used for designing approximation algorithms with the example of scheduling jobs with deadlines and release times in a single machine. There we have seen that checking whether all the jobs can be scheduled and finished before their deadline is NP complete. And this implies that there is no row factor approximation algorithm for this problem of computing the schedule with minimum lateness minimum maximum lateness of any job for any computable function row. And, then we have assumed that if all the deadlines are negative which in turn implies that optimal is positive we have seen a two factor approximation algorithm using greedy techniques. So, in today's class we will see a very important problem in clustering which is called k center problem and we will design a two factor approximation algorithm for the k center problem.

So today's topic is the K-Center problem. So, the problem of finding similarities or dissimilarities in a very large amount of data is ubiquitous. For example, a seller or or any anyone who wants to group their customers among based on their similarities and they can show the ads and to specific customers or to specific group of customers they can show various recommendations which are tailor-made to various kinds of customers and so on.

So, clustering data is an important problem. to group similar items ok ah. So, we assume that the items or objects are points in a matrix space. This implies that the points satisfy triangle inequality that is for any 3 points i any i j k distance of i between i and j. So, distances are symmetric the distance from i to j is same as distance from j to i.

So, $d_{ij}+d_{jk}\geq d_{ik}$. So, this is triangle inequality So, let us not use k because k will we will use in the problem definition for k center. So, suppose let us call it I j l and what is the objective? So, the idea is to group together similar objects each group is called a cluster. Each cluster has a center we want to partition the input set of points. into k clusters ok.

And what is the objective what we want to minimize for that we want to define. So, goal objective function So, let S subset of input data points V, where V is the input data points. S be the set of centres, $|S|=k$, each data point each centre defines a cluster naturally how. So, let $S=\{a_1,\ldots,a_k\}$, then cluster the ith cluster C_i is defined as all those data points $j \in V$ such that $d_{ij} \leq d_{i'j}$ for all $i' \in S$ ok, breaking ties arbitrarily. So, each data point belongs to one cluster.

So, if it happens that one data point is equidistant to more than one cluster centers, then then we will assign it to one cluster that this is what we mean by breaking ties arbitrarily. And the goal is to goal is to minimize the maximum distance from any data point to the cluster center which can succinctly written as $max_{j \in V} d_{j,S}$ where S is a set how does $d_{j,S}$ is defined? $d_{j,S}=min_{i \in S} d_{ij}$. in S. So, here is a set of points S and here is point j.

So, I look at the distance from j to all points in S and whichever has the minimum distance that distance we call the distance between j and the set S. The goal is to minimize this distance and we will design a very natural greedy algorithm So, we first pick any say any data point arbitrarily and define that is my first centre. pick any $i \in V$ and in this set S I am maintaining my cluster centers. Now, while k cluster centers have not been picked while this then I pick the next cluster center which is as far as from S. So, j is $argmax_{J' \in V} d_{j',S}$.

So, once I pick a pick once I start with any data point and put it in my set of centers, the next point the next cluster center is as is the farthest point from that point and so on. Every iteration I am picking the farthest point from the set of centers that we have chosen till now and that is it. to $S \cup \{j\}$ return S. So, clearly this is a polynomial time algorithm this while loop runs for k iterations or $(k-1)$ iterations and every execution of while loop runs in polynomial time. Now, what we will show next is that the approximation ratio of our algorithm is 2 theorem.

ah the above greedy algorithm has an approximation factor of 2 proof ok. So, let us pick let us look at any optimal k centers. So, let $S^*=\{j_1,\ldots,j_k\}$. be any optimal set of k centers ok. these centers naturally partitions V into k clusters.

call them $C_1,C_2,\ldots,C_k$ ok. And the let the value of the optimal objective function be $r^*=max_{i \in V} d_{i,S}$. ok. So, first observe that you first observe that in any cluster the distance between 2 points in that cluster is at most $2r^*$ this follows from triangle inequality. So, suppose this is one cluster center C_i and here is the center j_i. Now, if I take any two points now their distance is at most the distance between let us call it i_1 and i_2, the distance between i_1 and i_2 is at most the sum of distances between i_1 and j_i and i_2 and j_i

each of them is at most r^* which implies that the distance between i_1 and i_2 because of triangle inequality is $2r^*$ ok. So, it follows from triangle inequality that the distance between any two points in a cluster is at most $2r^*$ ok. Now, let $S \subseteq V$ be the set of cluster centers chosen by the greedy algorithm. Now, we break it into two cases case 1. $S \cap C_i \neq \varnothing$ for all $i \in \{1,\ldots,k\}$.

So, if it happens that our greedy algorithm picks exactly one point from each of the clusters the $C_1,\ldots,C_n$ these clusters are defined using the optimal cluster centers. So, if this happens then as we have argued that the distance between any two points in a cluster is at most $2r^*$, then for every point $i \in V$ So, that point belongs to some plus some C_i and in that C_i there is one center which is at most $2r^*$ distance away and each point is assigned to the closest cluster center. So, we have for every point distance between i and S is less than equal to $2r^*$. So, in this case we have shown that it is a two factor approximation algorithm. The other case is there exists an $i \in [k]$ such that $|S \cap C_i| \geq 2$. There exist one cluster C_i from which the our greedy algorithm picks 2 centers. Now, here also you see the first time our algorithm picks 2 centers from a C_i the distance $d_{i,S}$ from the distance of any point i to the set of centers is chosen till now is at most $2r^*$. So, here is suppose C_i and there are 2 centers picked say suppose this is $j_1^{'}$, this is $j_2^{'}$. Now, suppose $j_1^{'}$ is picked first and then in the iteration when $j_2^{'}$ is picked the distance of all points to to the centre chosen so far is already less than equal to $2r^*$ and this distances can only monotonically decrease it may not decrease it is monotonically non increasing. So, hence in this case also for every point $i \in V$, we have $d_{i,S} \leq 2r^*$.

Hence, our greedy algorithm has an approximation ratio of at most 2. Can we improve it? It turns out that if there exist any row factor approximation algorithm for any row strictly less than 2, then we have $P = NP$. So, theorem there is no ρ factor row approximation algorithm for the k center problem. for any $\rho < 2$ unless $P = NP$. And what is the proof? The proof is using a reduction from dominating set.

What is the dominating set? So, it is a reduction from dominating set. A dominating set of a graph G is a subset of vertices, so that every other vertex dominating set, so that every other vertex is neighbor of it. Okay. So, let us see how we can solve dominating set if we have a row factor better than two factor approximation algorithm. in the dominating set problem we are given a graph G and an integer k and the question is does there exist a dominating set of size k this problem is known to be NP complete.

So, what we define the we reduce this problem to the k centre problem the points set in the k centre problem is v and how do we define the distance? Distance between i and j is 1 is 1 if this edge if there is an edge between i and j. and 2 otherwise. Now, do you see that the optimum value of the k center problem is 1 if and only if there is a dominating set of size k. Observe that r^* of the reduced k center problem is 1 if and only if there is a dominating set of G of size k ok. You can prove it formally that you take it as a homework it is easy.

But here you see if the optimum is 1 and the approximation ratio is rho which is strictly less than 2, then any rho factor approximation algorithm has to output the optimal solution. That means, with they have to solve the dominating set problem which is NP complete problem and hence if there is a better than 2 factor approximation algorithm. even if 2 minus say 1 power 1 2 minus 1 by say 2 to the power n or anything anything better than 2 factor approximation algorithm for k center problem using that algorithm we can solve the dominating set problem in polynomial time thereby showing p equal to n p. So, under $P \neq NP$ assumption, this is the best approximation ratio that we can obtain for the case center problem ok.

So, let us stop here. Thank you.