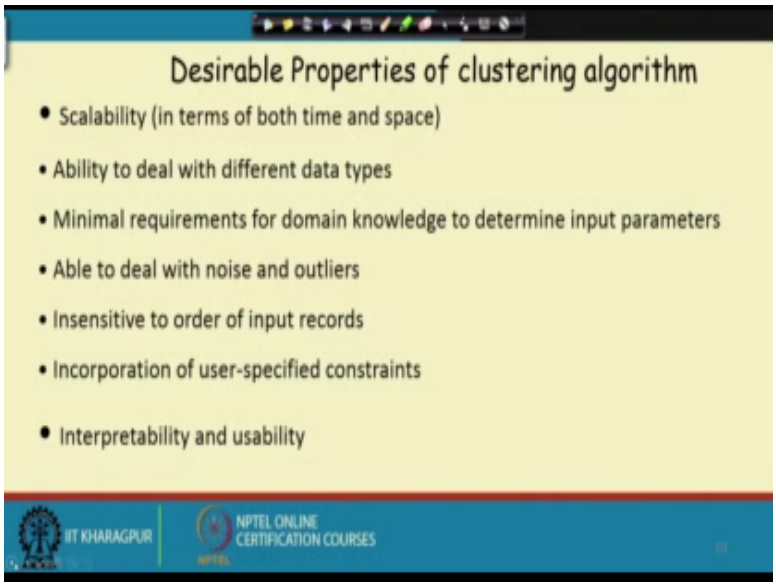
Data Mining
Prof. Pabitra Mitra
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

Lecture - 33
Clustering – II

We continue the discussion on Clustering. As we had mentioned in the previous lecture that we want to group points into homogeneous groups and the homogeneity is measured by the ratio of the average distance between the points which lie in the same cluster to the points which lie in different clusters, the scatter coefficient. And we find, we discussed that there are two ways of doing this one is partitional where we try out all possible groupings of the point and select the one having the best, and the other is hierarchical where we either split the point starting with all points in the same cluster and then splitting them based on some criteria or we do the reverse bottom up that is initially considered every point to be a separate cluster and then group them one by one.

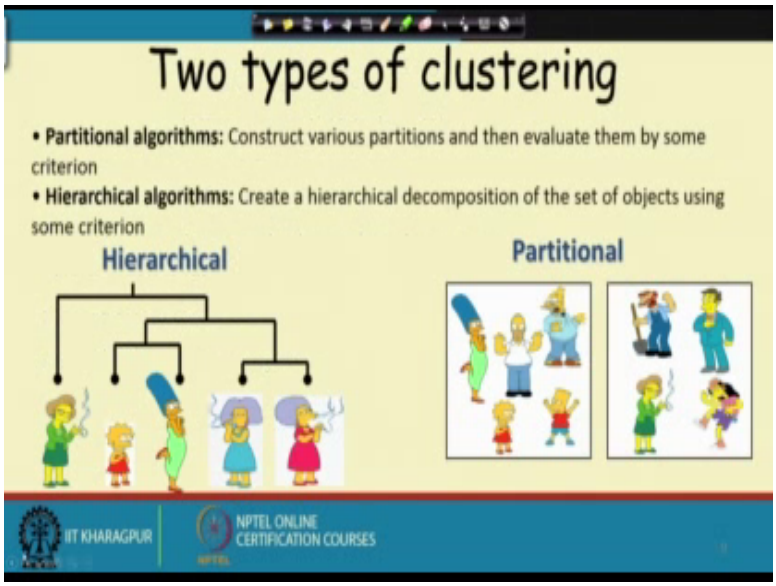
So, and we found, we saw that, so this is what we had already discussed. We need some kind of the basic assumption, we need some kind of similarity before we start the clustering. So, that can be distance or anything.

(Refer Slide Time: 01:36)



So, we first discuss the methodology of this kind of hierarchical clustering where we merge them and then and get different clusters.

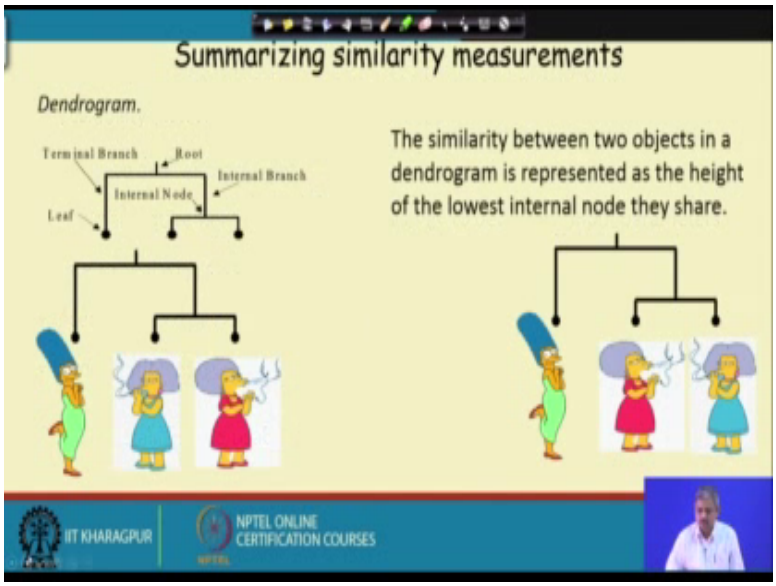
(Refer Slide Time: 01:40)



And there are two kinds as I have told agglomerative which is bottom up and divisive which is top down. And we all we have also shown that what we construct basically is a tree like structure explained here which we called a dendogram.

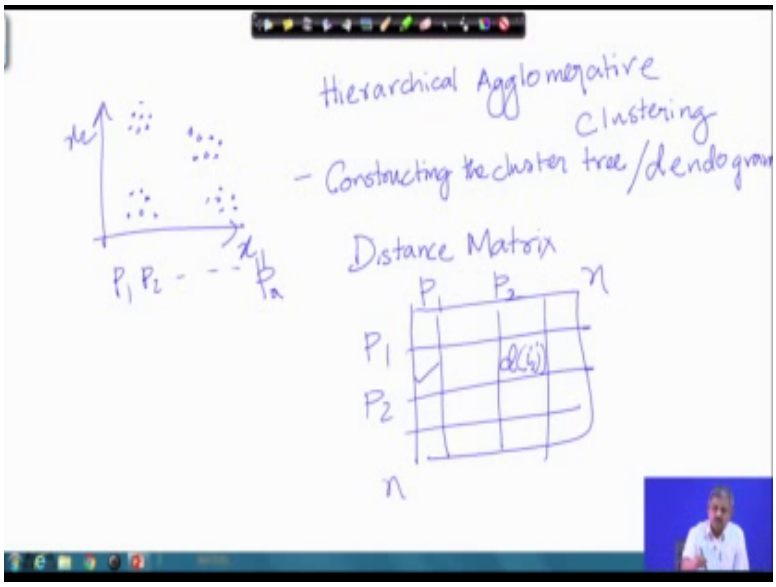
Before I go into the method, so one of the I want to just quickly mention some of the properties, we will discuss this algorithm before, we discuss them we tell that some of the properties we desire of the clustering algorithms is that it should be fast both in time and space it should be fast, it should be deal with not just numerical, but nominal ordinal attributes also it should be, it should be able to deal with noise and outliers, it should be independent of the order of the input records and if necessary you should specify certain contents constants and finally, interpret. So, these are some characteristics we need.

(Refer Slide Time: 02:58)



So, what I will do is that I will now explain the hierarchical clustering starting with the bottom up that is the agglomerative approach. So, let me explain you with the help of points as I have all always done.

(Refer Slide Time: 03:17)

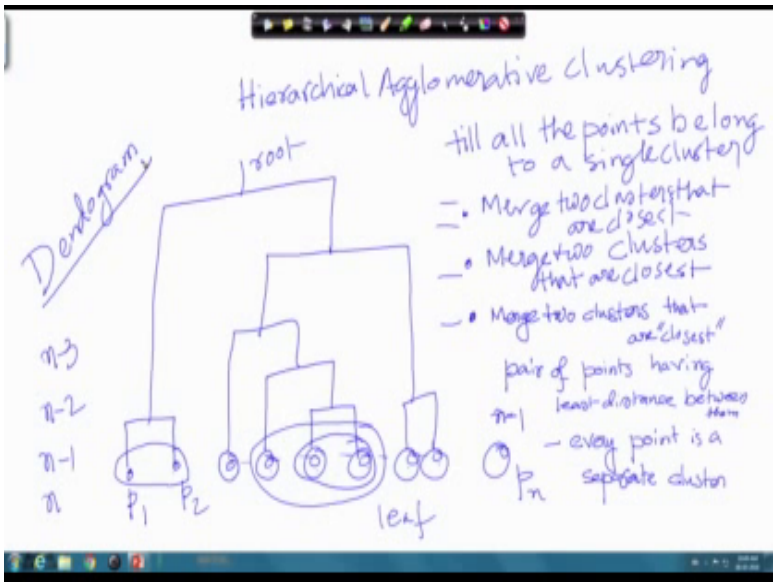


So, your points are say two dimensional points given say like this these are some points note that we do not have any class information here, we only have the similarity between some distance between the points and you have to find the natural clusters, the natural group groupings. So, I explained hierarchical agglomerative, agglomerative cluster and I

will tell you way of constructing the cluster tree as I had shown before or the also known as the dendogram.

Let me explain. So, suppose I have n points P 1, P 2 to P n in two dimensional part. It may be in higher dimension also. Actually they need not be points like this, all I need is a distance matrix between the points. So, if there are n points all I will need for this is a kind of distance matrix, which is a for n points a n by n matrix, n by n matrix which tells you every entry tells you the distance between point i and point j. So, the distance between P 1 and P 2 will be this it tells you. So, every distance matrix is d i j the distance between i and j part. So, all I need for hierarch; I need not know their coordinate values also all I need needs their pairwise distances distance between every pair of points. So, let me explain.

(Refer Slide Time: 05:37)



(Refer Time: 05:42) I am writing down again that you can check the note. So the idea is like this. So, in the leaf of the dendogram you consider all the points as all the points, so p 1, p 2 up to p n.

Assume every point is a separate cluster, so every point is a separate cluster. Of course, they will be singleton clusters because there will be only one element per cluster, but we start with that, every point is a separate cluster. So, there are n clusters if there are n points to a cluster.

Now what you do check the pair of points which have the least distance between them. So, you consider pair of points having least distance between them, between them. Suppose I mean I am just explaining p 1, p 2 are the closest pair of points. So, what I do earlier there are n clusters p 1, p 2, p n every point a cluster every point a cluster, now if p 1 p 2 are closest they get merged, these two clusters get merged, these two small clusters get merged to a slightly larger cluster containing two points, earlier a cluster at one point now the cluster has two points. Other points are single point clusters. So, in the next level of this tree I have n minus 1 clusters, in the next level I have n minus 1 clusters because two points got, two clusters got merged.

So, now I can in the, to further build the tree I can visualize it like this. So, this is a two point cluster and these are single point clusters, but there are clusters. Now I will merge two clusters which are closest, merge two clusters. I have not yet defined what is closest I have defined two points how they are closest have defined by their distance, but how two clusters are closest I have not yet defined I will define soon.

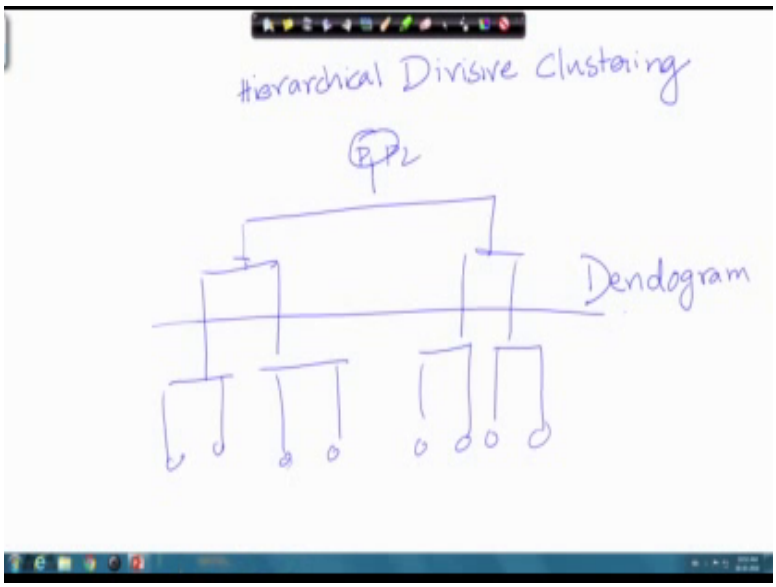
So in the next level merge two clusters that are closest. I am not yet told what is closest means ok, but suppose I have some measure that what is closest. So, what I will do? I will merge. So, it is possible that these two clusters are closest or these two single one clusters are closest or this and this is closest whatever, whichever is closer let us say this two is closest, I merge it.

So, now, what happens? Now I have n minus two clusters one cluster two point another cluster two point and then this single point clusters ok. So, in the next level I have, so here I add n, here I add n minus two clusters. Now, repeat this again merge, again merge two clusters among the among this after this step merge two clusters that are closest. So, what may happen maybe now this cluster and this cluster is closest. So, they get merged into a 3 point cluster, so 3 point clusters. So, you see I am gradually building the tree and repeat this clusters that are closest. Just go on repeating this, just take two clusters that are closest let us say now these two get merged maybe after one more step this is to get merged, then maybe after one more step this two get merged and maybe after one more step these two get merged ok. So, this is repeated till all the points belong to a single cluster the root, ok. So, this tree is that dendogram. So, I repeat once again what I did.

Initially if I had to cluster n points each and every point is a cluster. And then I go on merging two clusters which are closest which are closest again, I have not yet told again what is closest. I keep on merging the closest cluster two clusters which are closest they merge together from one cluster I keep on doing this. I keep on doing this till everything get merged to a single cluster and I get the dendogram tree ok.

So, now, you will say you have just got a tree. So, how do I get my groups? How do I get? Suppose I want to divide the point into 3 clusters how do I get 3 clusters ? So, what I do, I look at the level of the tree where there are 3 branches. So, this level there are 3 branches, I cut the tree there, I cut that you there and each of the branch form a cluster. So, now, there are 3 clusters these two one cluster, these four one cluster, this two one cluster ok. So, I cut the dendogram, dendogram at k minus 1 height, k minus 1 if I had to get k clusters right, ok. So, this is agglomerative bottom up. The reverse of this is divisive reverse of this is divisive.

(Refer Slide Time: 13:43)

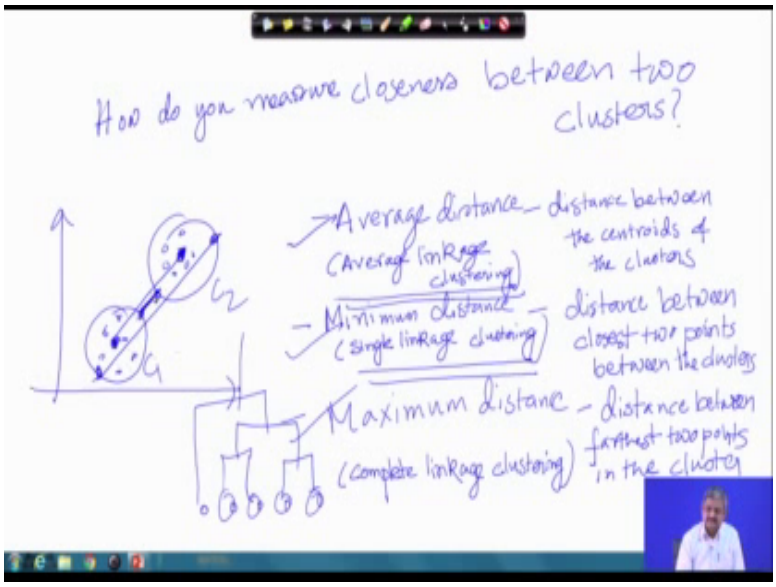


So, what happens there? There are initially all the end points found in one cluster p 1 to p n, and then they are divided into two clusters based on two sets of point which are furthest then they are further subdivided till they are furthest and go on doing this till every point is a separate cluster ok, is a separate cluster. So, same dendogram, but now constructed the other way round. And again if one k clusters you cut at the k minus 1

height ok. So, this is clear. So, both hierarchical divisive and agglomerative this is clear ah.

You basically construct a tree and then cut the tree. And how do you construct a tree? In bottom up you merge the in initially every point is a cluster then merge the closest ones and divisive everything all everything is a single cluster and you divide the furthest one ok.

(Refer Slide Time: 15:10)



So, now, I will define distance between two because I have told you that you have to merge the closest clusters. So, how do you measure closeness doing two clusters? How do you measure the closeness between two clusters?

So, let me check here and say this is another. So, I have say two cluster c 1 and c 2. So, what is a measure of distance between c 1 and c 3? There are 3 options, one is called, one is the average distance. You take the centroid of both the clusters and distance between them is the distance between two clusters, not that distance between two points are previously defined either Euclidean or Manhattan or something I have defined. Using the distance between two points I am defining distance between two clusters is the distance one possibility, average distance.

Mean distance, minimum distance, so you define distance between two cluster at the distance between the closest two points in the cluster, closest two points in the cluster,

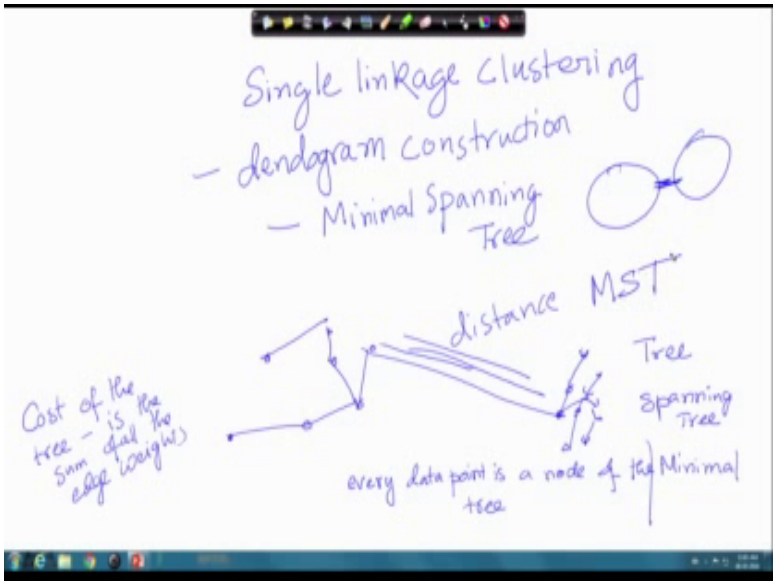
closest two points between the clusters. Similarly you can take maximum distance where the distance is the distance between farthest two points in the cluster. So, points in the in the cluster ok. So, 3 way, 3 different ways you can measure it.

Depending on which way you measure you have a different, so you remember as what I will do you remember the hierarchical agglomerative divisive I was telling that merge two clusters have been closest which are closest. So, now, closest will be measured in terms of either this or this or this ok. That is how closest will be measured ok.

So, depending on which of these 3 you use, you have 3 names for this hierarchical algorithms. If you use the minimum, if you use the average distance you call it a average linkage clustering, remember the name average linkage clustering. If you have the minimum distance as the closeness you have the single linkage clustering, one of the most popular algorithm single linkage cluster. If you have the maximum distance you call it a complete linkage clustering complete linkage clustering ok, single linkage, complete linkage, average linkage. Method is same, same dendogram, but closeness of two clusters is differently defined.

So, this single linkage clustering also has a graph theoretic implication. See, how I have told you merge two clusters having closest, same point closest merge two clusters having closest nearest neighbours and closest two points that is the single linkage.

(Refer Slide Time: 21:15)

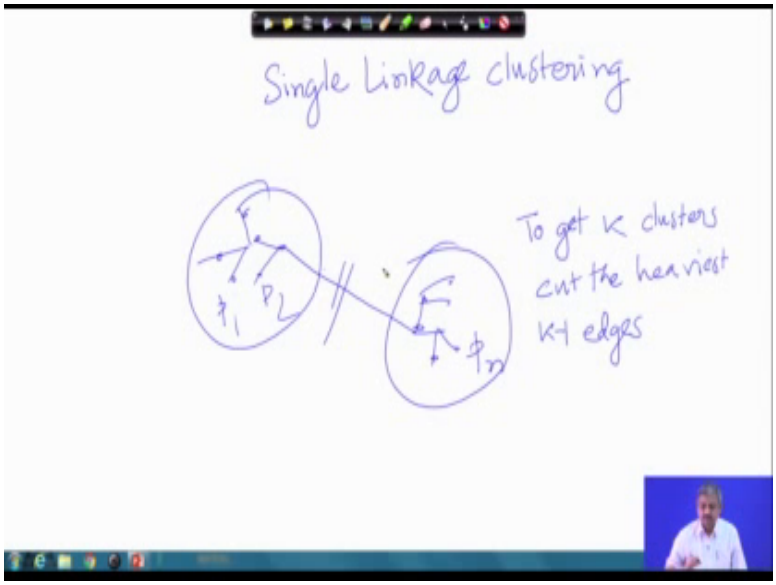


So, these dendogram construction, note that single linkage means the closeness is the distance between the closest pair of point, construction actually produces a minimal spanning tree I will explain what it is. You might have those who have done a graph theory course might have heard this term ok.

So, what you do is the following. Take every point, this is one, this is one, so two groups of point. You make a tree; that means, you connect a tree like structure. So, two points there is only one path between two points ok, similarly here only one path between two points ok. So, it is a tree there is no cycle basically there is a tree and it is a spanning tree; that means, every point is a node, tree, it is a spanning tree ; that means, every node is there is a path no board node is disconnected. And if we take this edge weights to be the distance between these two points the total cost of the tree is the sum of all these edges.

So, edge weights and among all possibility this is the minimal, minimum. So, that is why it is called a MST, minimal spanning tree. So, actually the dendogram construction is same as the Prim's algorithm of minimal senile c construction those who know.

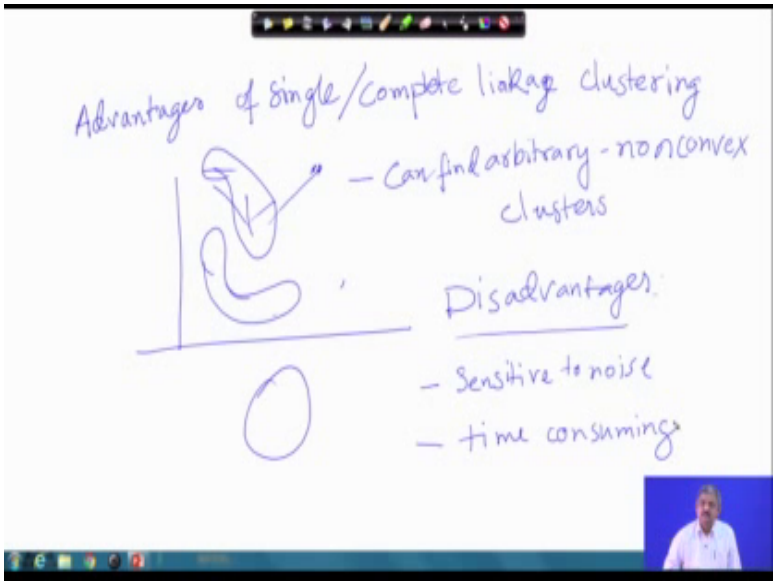
(Refer Slide Time: 24:23)



So, what you do in single linkage clustering? Take the points p n, construct a MST. Now, to get k clusters cut the heaviest having the highest weight. So, if you want to get two cluster cut the heaviest point, you want to get three cluster cut two heaviest weights and

that will partition the points in sense is a tree till position the point into two groups those are your clusters.

(Refer Slide Time: 25:34)



What are the advantages of single and complete linkage clustering? See, if you have arbitrary separate clusters the algorithm will still find them non convex; that means, convex means like this closed it is non convex clusters.

Some disadvantages, if you have a noise point then this algorithm fails, so sensitive to noise. So, imagine like this it will be completely wrong, noise time consuming slow ok. These are the disadvantages ok.

So, this covers my discussion on these two algorithms, In the next lecture we will discuss the partitional clustering algorithms which will like as I mentioned it evaluates the partitions.

Thank you.