Applied Multivariate Analysis

Prof. Amit Mitra
Prof. Sharmishtha Mitra
Department of Mathematics and Statistics

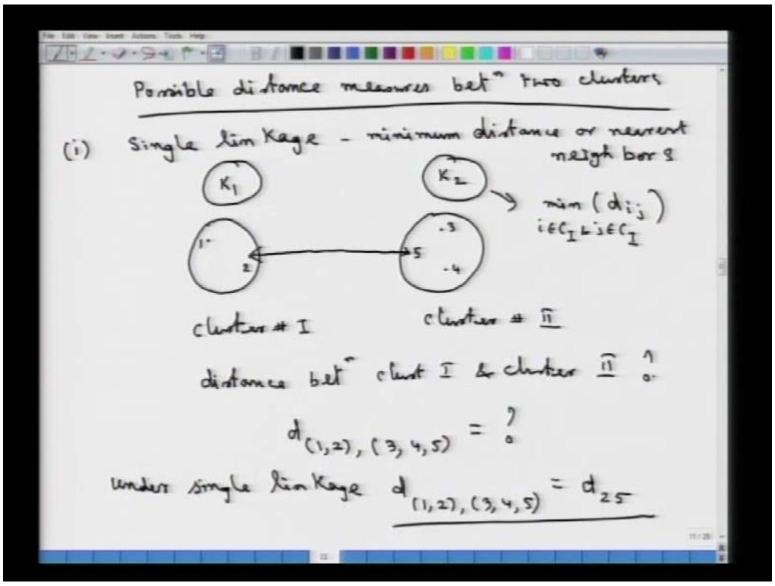
Indian Institute of Technology, Kanpur

Lecture No. # 27

Cluster Analysis

In the last lecture we had started discussing about statistical cluster analysis techniques, and we had introduced the various types of clustering techniques, that are usually applied hierarchical or non nonhierarchical clustering. And we were discussing the hierarchical clustering in more detail. And to do that, we were looking at the type of a distance measures that are usually used; because as we had discussed that once we form new clusters, we need to find out the distance between the new cluster that is formed at a particular step of iteration and the existing clusters. And in order to do that we need to have some sort of approach that we had started discussing.

(Refer Slide Time: 00:59)



Possible distance measures bet^n two clusters

(i) Single linKage - minimum distance or nearest neighbors

K_1 K_2 → $\min_{i \in C_I \,\&\, j \in C_{II}} (d_{ij})$

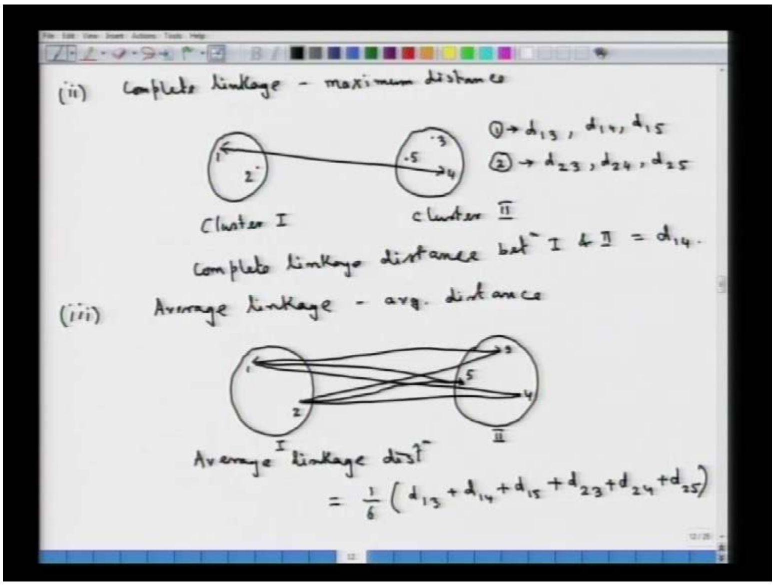
distance bet^n clust I & cluster II ?

$d_{(1,2),(3,4,5)} = ?$

under single linKage $d_{(1,2),(3,4,5)} = d_{25}$

The first type of approach that we say it was that of the single linkage, wherein we look at the minimum distance between the two clusters. One may be newly formed and other one may be an existing cluster. So, we try to find out what is the mutual distance in terms of the single linkage and that is basically looking at the nearest neighbors among these two clusters. There are other ways of defining this distance between clusters which we are going to see first of all today.

(Refer Slide Time: 01:28)



(ii) Complete linkage - maximum distance

① → d_{13}, d_{14}, d_{15}

② → d_{23}, d_{24}, d_{25}

Complete linkage distance betⁿ I & II = d_{14}.

(iii) Average linkage - avg. distance

Average linkage distⁿ

$$= \frac{1}{6}\left(d_{13}+d_{14}+d_{15}+d_{23}+d_{24}+d_{25}\right)$$

So, the second one is, what we call as a complete linkage. So, the complete linkage distance in contrast to the single linkage, it looks at the maximum distance as the way of looking at two different clusters and then, finding out the distance between the two clusters. Let us look at the type of things that we were looking at for the single linkage. So, suppose these two are two clusters. So, this is cluster number 1 and this is cluster number 2. Then, in the previous class we had this 1 and 2 in this cluster 1 and, 3 4 5 in this order in cluster number 2. So, it was a 5 here, a 3 here and we have the first case here 1 and the second case here 2. So, this was the structure of the 2 clusters; this visualized on a two dimension.

If it is on a higher dimensional than 3, we ofcourse cannot visualize such clusters. But basically, what we are trying to see is that among the mutual distances between any case that is belonging to cluster number 1 and any case that is belonging to cluster number 2. So, the distances will be d 13 that is the distance from case number 1 to case number 3, which is belonging to the second cluster; distance between 1 and 4 and distance between 1 and 5. So, these are the distance starting from this one node which is sitting here. Similarly, from the second case node number 2, they will have the distances as d 23, d 24 and d 25.These are the 6 possible distances, which are when we are looking at one case from cluster number 1 and the second case from cluster number 2.

So, these are the distances. Now, when we look at complete linkage in contrast, in the single linkage we had looked at this d 25 which is the nearest neighbor; this is the farthest neighbors that we are looking at. So, what will be having for the farthest neighbors are these 2 points. The distance between 1 and 4, 1 belonging to cluster number 1 and 4 belonging to cluster number 4 cluster number 2 appears to be the maximum. In any case, if any other distance is maximum that would be the complete linkage distance. So, here once we have that interpretation, this complete linkage distance complete linkage distance between cluster 1 and cluster 2 would thus be given by d 14.

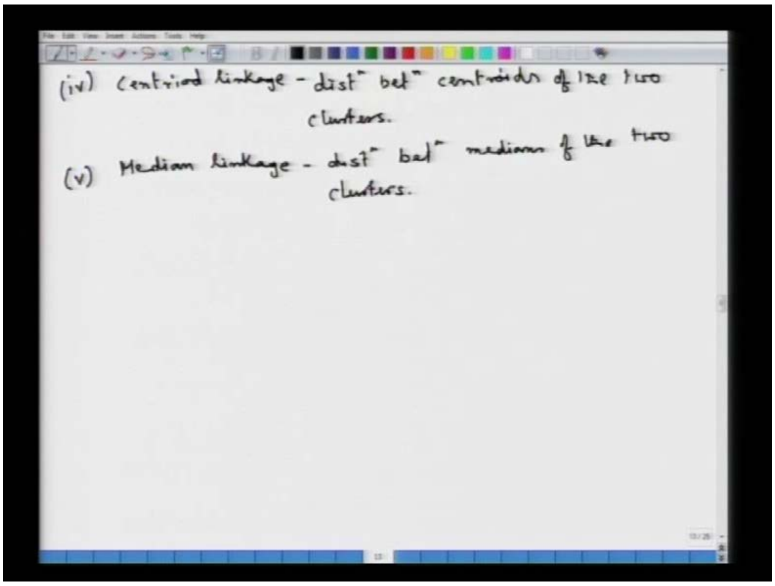
So, that now gives us the distances between these two clusters. When we have higher dimensions, we compute all such possible distances between the multidimensional points belonging to 1 cluster and the multidimensional points belonging to the other cluster and then, find out which of them is maximum. And then the maximum distance is used in order to measure the distance between the 2 stated clusters, cluster 1 and cluster 2; that is simple. In case we have k 1 cases belonging to cluster 1 and k 2 cases belonging to cluster 2, we find out all such possible distances and then look at what is which distance is maximum. And hence then use that, in order to give the complete linkage distance between the two clusters.

Now, there is a third type of distance measure which is called the average linkage which is called the average linkage. So, this looks at the average distance average distance in what sense. So, we still have these two clusters, cluster 1 and cluster 2; 1 and 2 sitting here and we have another cluster here which is having the cases which is 5; then we have a case 3 and a case 4 sitting here. Now, we will be looking at the average of the distances. The distances are 1 unit belonging to one cluster and the other unit belonging to the other cluster. So, we look at all possible such d ijs ; i belonging to cluster number 1 and j belonging to cluster number 2.

And then find out the average over all i j ; i belonging to cluster number 1; j belonging to cluster number 2. So, what will be having here is that from 1 we have 1 distance here; from 1 we have a distance to 5; from 1 we have another distance to 4; from 2 we have a distance to 3; from 2 we also have a distance to 5 and we have a distance to 4. So, the average linkage would be the average of all these 6 distances that we have come come up with. So, the average linkage distance between the two clusters, cluster number 1 here and cluster number 2 here is going to be given by…

There are 6 such distances and thus, this is going to be just the sum of d 13 plus d 14 plus d 15 this plus d 23 plus d 24 plus d 25. So, these are all these 6 possible distances distances computed, when one object from one cluster is taken to be compared with another object in the other cluster and that 1 upon 6. So, in general as I said that, this average distance is going to be 1 upon the number of such distances. If this as k 1, if this as k 2, then this number would be k 1 times k 2. And then we will have a summation double summation over i and j; i belonging to cluster number 1 and j belonging to cluster number 2.

(Refer Slide Time: 07:50)

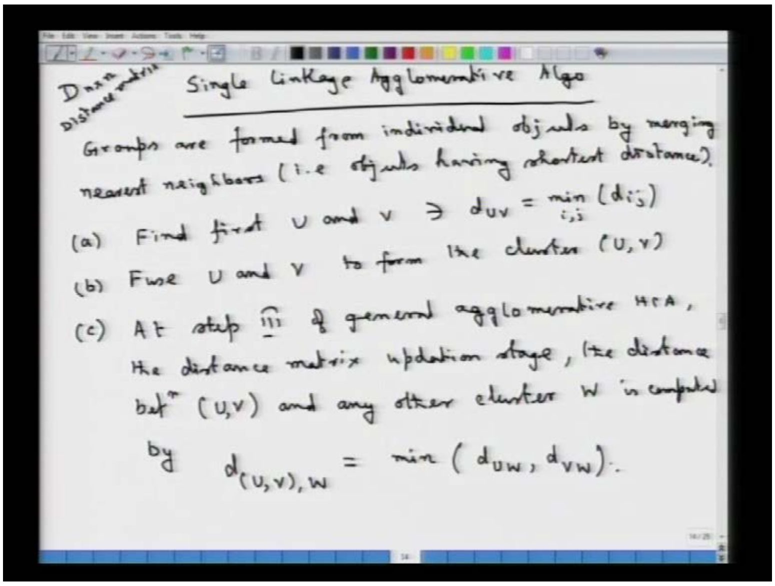


(iv) Centriod linkage - dist^n bet^n centroids of the two clusters.

(v) Median linkage - dist^n bet^n medians of the two clusters.

So, these three are the most widely used used distance measures, in order to compute distance between two clusters. There are other types of measures which one is called centroid linkage. This basically is looking at the distance between the two centroids of the two clusters. So, this is the distance between centroids of the two clusters. So, one can compute what are the centroids of the respective clusters and then find out what is a distance with respect to whatever distance matrix we are considering between the two centroids. Similar to the centroids, where wherein we are looking at the mean of these points, central point one can look at median also median linkage. So, this is going to be the distance between medians of the two formed clusters. So, this is these are some various ways of looking at how to measure distance between two separate clusters.

(Refer Slide Time: 09:11)



$D_{n \times n}$ distance matrix

Single Linkage Agglomerative Algo

Groups are formed from individual objects by merging nearest neighbors (i.e objects having shortest distance).

(a) Find first U and V ∋ $d_{UV} = \min_{i,j}(d_{ij})$

(b) Fuse U and V to form the cluster (U, V)

(c) At step iii of general agglomerative HCA, the distance matrix updation stage, the distance betn (U,V) and any other cluster W is computed by $d_{(U,V),W} = \min(d_{UW}, d_{VW})$.

Now, we look at the agglomerative single linkage algorithm in detail. Let me start fresh here and then try to understand how these distance measure are going to play a role. When we actually form this agglomerative clustering with where wherein we are going to end up finally with the dendogram tree tree structure which is going to give us clusters in a hierarchical form. So, let us look at this single linkage agglomerative algorithm. Now, how is this going to go about this is going to go go about in the following way that groups are formed from individual objects by merging the cases which has got the shortest distance.

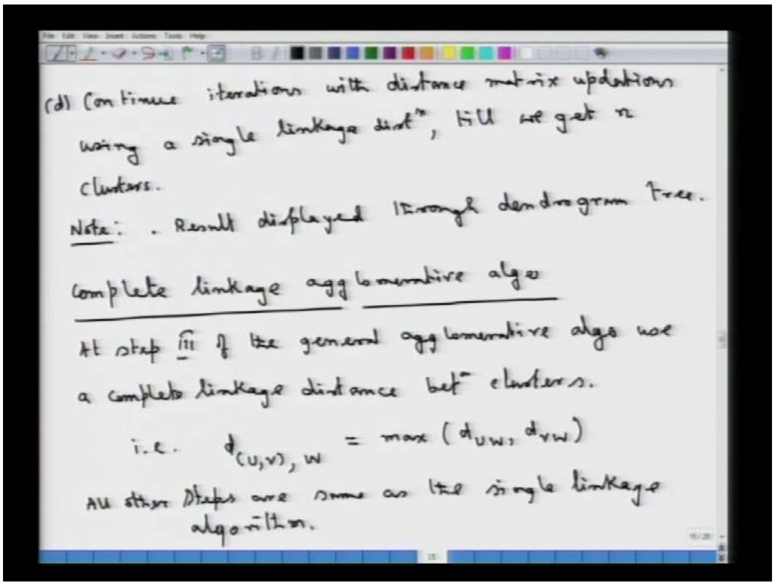
By merging, I can say which are nearest neighbors the nearest neighbor that is cases having cases or objects having shortest distance shortest or minimum distance. So, once you have that, we can follow these steps here that given this D matrix which is the starting point. Find first the smallest; find first U and V such that this is smallest. I will say find first U and V such that d UV is basically the minimum among all d ij minimum over all i j here. So, we are looking at the distance matrix. So, we ofcourse have to start with a matrix D which is n by n corresponding to the n objects. So, this is the distance matrix.

So, corresponding to all those n cases, we have this D to give us the distance matrix and these d ijs are elements of that distance matrix. So, we first find U and V such that d UV the distance between u and v is the minimum possible then, we will merge these two. So, we will merge a fuse U and V to form the cluster U V. Now, comes the fact that once U and V have been merged, we have these two forming one single cluster. And then from the remaining n minus 2 cases, we will have n minus 2 clusters and this will make up the n minus 1th cluster. Now, we need to find out what is the distance between U V cluster and the cluster which we have the original existing n minus 2 clusters.

Now, we will follow the general agglomerative steps here. So, at step 3 of general agglomerative agglomerative hierarchical cluster analysis, the distance matrix needs to be updated. So, the distance matrix updating stage, the distance between between U V, this newly formed cluster and any of the other and any other cluster say W that there will be n minus 2 of them at this first step. The distance between this and this is computed by is going to be computed by the following form that, distance between U V; this is the newly formed cluster and W. Note that, we are now using a single linkage. So, we have one cluster wherein, there are two cases and the other cluster which is having only single case, which is having the identity as W.

So, what we are going to do, if we are looking at a single linkage agglomerative algorithm is the following that we will look at the distance between U and W and V and W. And then, find out with which is minimum and that minimum would then be the distance between this newly formed cluster U V and the existing cluster which is W. So, what will be having is that distance between U V and W to be given by minimum of these two quantities; distance between U and W and distance between V and W. Simple. So, once we can do that, we can do it for all subsequent iteration.

(Refer Slide Time: 14:35)



(d) Continue iterations with distance matrix updations using a single linkage distⁿ, till we get n clusters.

Note: . Result displayed through dendrogram tree.

Complete linkage agglomerative algo

At step iii of the general agglomerative algo use a complete linkage distance betⁿ clusters.

i.e. $d_{(U,V),W} = \max(d_{UW}, d_{VW})$

All other steps are same as the single linkage algorithm.

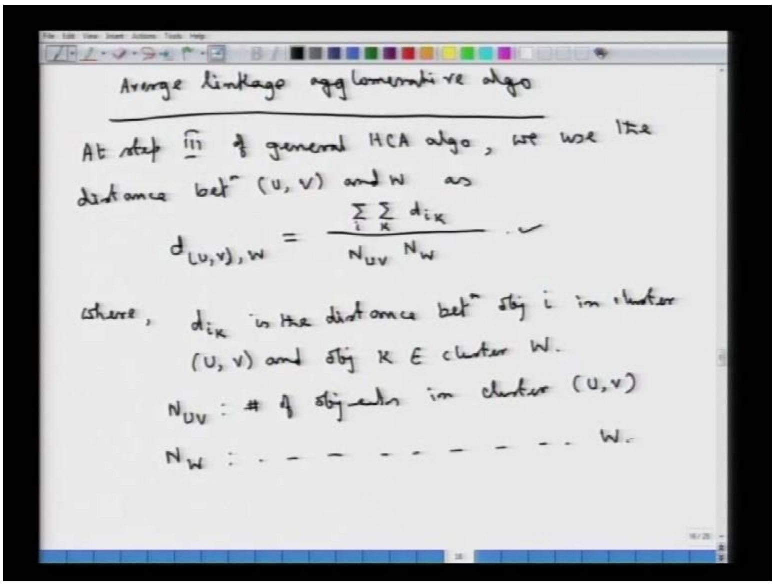
So, one can actually look at that, continue the steps of iteration we had number see here. At this d, continue iterations with distance matrix upgradations or rather updations updations using a single linkage distance. This is to be continued till we get n clusters; that is, we come down to the last level of the finest level of the resolution. In the finest level of resolutions, in division cases are going to be members of single clusters. So, a case will be sitting actually on one single cluster. Now, just to remind you that this is what we are going to have; results are going to be displayed using a dendogram tree displayed through Dendogram tree. And that branching type of structure is going to lead us to the hierarchical clustering that is what is desired here.

Now, one can use the similar type of approach, when we have a complete link linkage or an average linkage. For the complete linkage, I am not going to write the entire algorithm once again. We will just say at what point that is going to differ, it is going to differ this complete let me first write it complete linkage agglomerative algorithm. So, how is this going to behave? This is going to go along exactly in the same line. Up to this particular stage, wherein we require updation of the distance matrix. When we are then trying to have updation of the distance matrix, we need to once again find out the distance between U V and the cluster W.

And that is, now going to be given through a complete linkage, if you have a complete linkage algorithm agglomerative algorithm. And there, instead of minimum which is going to look at the nearest neighbors. In a complete linkage, we are going to look at the farthest neighbors and hence, this minimum would get just replaced by maximum. So, we will just say that, at this step 3 here; at step 3 of the general agglomerative algorithm use a complete linkage distance linkage distance between clusters distance between clusters. That is, now this distance between U V, the newly formed cluster and an existing cluster W is going to be given by the maximum over the same two distances d UW and d VW.

All other steps are same as the single linkage algorithm. So, it is very simple, you have the single linkage agglomerative algorithm in this following ways. You will have that complete linkage first, you look at the minimum which two objects are closest by finding the minimum over i j of all these d ij entries. Then fuse them, form the clusters and then come to the updation stage. And there, the distance between U V from cluster U V and existing case W existing case at the first step. Now, in the second stages onwards, this also will be clusters. So, we will have that being computed through a complete linkage and wherein will be using this maximum here.

(Refer Slide Time: 19:18)



Average linkage agglomerative algo

At step $\underline{\text{iii}}$ of general HCA algo, we use the distance bet^n (U, V) and W as

$$d_{(U,V),W} = \frac{\sum_i \sum_k d_{ik}}{N_{UV} N_W} . ✓$$

where, d_{ik} is the distance bet^n obj i in cluster (U, V) and obj $k \in$ cluster W.

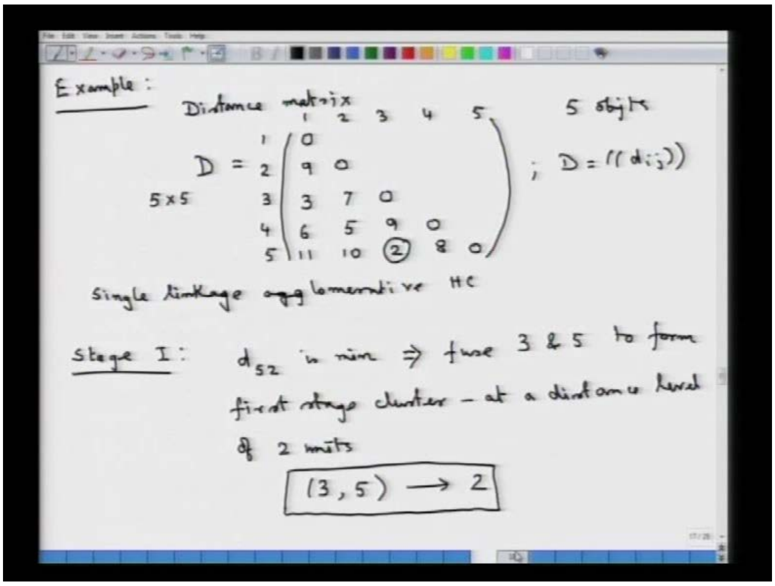
N_{UV} : # of objects in cluster (U, V)

N_W : - - - - - - - - - W.

Now, when we have an average linkage, the steps also are similar. So, average linkage agglomerative agglomerative algorithm will follow exactly in the same way. We will have at the third step only. These algorithms are going to differ at step 3 of general hierarchical cluster analysis algorithm .We use the distance between U V form the cluster new cluster and W. As this d UV, this is the first cluster that is formed and w that is going to be given by 1 upon… Let me write this fraction bit larger; this is the number of cases which are there in u v and the number of cases which may be there in w. At the first step, this N W will be equal to 1 and N UV will be equal to 2. But at the subsequent steps, when we are looking at the other iterations in order to get to the N clusters, this will these will be different. We will see that in data examples.

Now, this in the numerator, we have summation over i summation over k and the distance is given by d ik. Wherein, we have got this following interpretation that, this d ik is the distance between object i in cluster U V in cluster U V and object k belonging to the cluster which is denoted by W; this is an existing cluster. So, what we are looking at is we are looking at all the d ik such that i is belonging to the U V cluster and k is belonging to the W cluster. And then, finding out all such distances and then finally, that is going to be given by the average linkage. The average linkage would find out the average of all such distances. So, this is the total of all such distances and these are the number of such distances and hence, we will have this to give us the average of the all such distances. This N U V is the number of objects in cluster U V and similarly, this N W is the number of objects in the cluster which is given by W.

(Refer Slide Time: 22:21)



Example :

Distance matrix

5 objts

$$5\times5 \qquad D = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & & & & \\ 2 & 9 & 0 & & & \\ 3 & 3 & 7 & 0 & & \\ 4 & 6 & 5 & 9 & 0 & \\ 5 & 11 & 10 & \textcircled{2} & 8 & 0 \end{array} \; ; \quad D = ((d_{ij}))$$

Single linkage agglomerative HC

Stage I : d_{52} is min $\Rightarrow$ fuse 3 & 5 to form first stage cluster – at a distance level of 2 units

$$\boxed{(3,5) \longrightarrow 2}$$

Let us look at the numerical example, in order to see how these things actually work. Let us look at this numerical example. We have got this distance matrix to start with. So, the row data has been transformed into the following distance matrix which is given by say D which is of the following form. It is a symmetric matrix as we discussed yesterday that we will be having this 0 along the diagonals. Because this is going to measure the distance between 1 and itself and that is 0. This is going to measure the distance between 2 and 2 that is going to be 0. So, distance from itself is taken to be 0 naturally. These are the other entries. Let me plug in these entries 6 5 9 0 and the last row is giving the distances as 11 10 2 8 and 0.

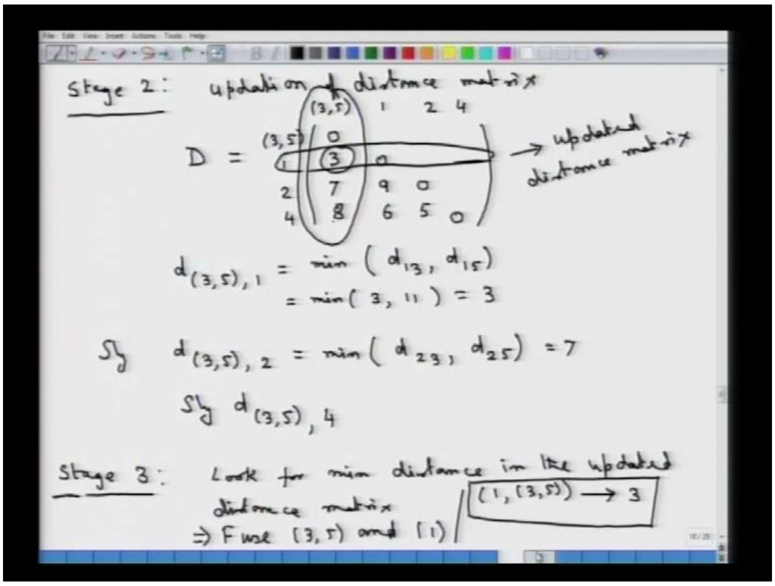
So, this has got a 5 by 5 dimension. So, they are there are naturally 5 cases. So, we have 5 objects or cases which need to be cluster according to hierarchical clustering algorithm. First of all, let us look at a single linkage agglomerative hierarchical clustering. So, this distance matrix has got the interpretation that, these are basically the objects, object identifications. So, 1 2 3 4 5 this along the columns 1 2 3 4 and 5. These are all the d ij entries. So, this matrix is basically the matrix of distances. So, we will have that to be given by theses d ij s. For example, this particular term here is the distance between object 2 and object 5. Now, suppose we are trying to implement single linkage agglomerative hierarchical cluster.

So, we are trying to implement the algorithm that we have just now learned in order to get clusters, hierarchical cluster out of clusters rather. From this particular data which has been represented through a distance matrix. If we are going to do that, what is the first thing that we will be doing. So, Stage 1 from this distance matrix, we are going to see which of the two case cases are closest. Now, if we look at the mutual distances, we find that this is a smallest. So, remember what we were doing here, when we were discussing these algorithms that will be looking at, find first two cases U and V such that d UV is minimum among all such d ij. So, for this given data what is that? These are d ij minimum of all these d ij ofcourse excluding 0s.

Because there is no point fusing 1 with itself; it is already setting in one cluster. So, we have to look at all such d ij s such that, i is not equal to j ofcourse. So, we will have this 2 which is the minimum here. So, this d 52 is minimum minimum among all these d i js. So, this would imply that we will fuse 2 and 5 to form first stage cluster first stage cluster at a distance level of 2 at a distance level of 2 at a distance level of 2 units. We will have to preserve this particular information; because we would require these informations. Later on, in order to frame the dendogram diagram. So, this is an input at the Stage 1 which we need to keep track of. Now, the next thing that we are going to do is that, now this let me see, it is not 2 and 5, this is corresponding to 3 5.

So, we are going to fuse case number 3, 2 5 is here; we are going to fuse case number 3 from here and case number 5. This is going to be replaced by 3 here. So, we are fusing case number 3 and case number 5 at level 2. Now, since 3 and 5 are no longer members of single unit clusters, the distance matrix when that needs to be updated. What are the existing clusters? The existing clusters will be single turn point clusters, which is having case 1 as 1.1 cluster. Second cluster which is having 2 the second unit and then, 4 will also be a member. Because that is an existing cluster and the new cluster is 3 and 5. So, you will have to find the next distance matrix which is going to be a 4 by 4 distance matrix, wherein the 4 clusters are 1 2 4 and 1 2 4 and 3 5.

(Refer Slide Time: 27:53)



Stage 2: updation of distance matrix

$$D = \begin{array}{c|cccc} & (3,5) & 1 & 2 & 4 \\ \hline (3,5) & 0 & & & \\ 1 & 3 & 0 & & \\ 2 & 7 & 9 & 0 & \\ 4 & 8 & 6 & 5 & 0 \end{array}$$

→ updated distance matrix

$$d_{(3,5),1} = \min(d_{13}, d_{15})$$
$$= \min(3, 11) = 3$$

Sly $d_{(3,5),2} = \min(d_{23}, d_{25}) = 7$

Sly $d_{(3,5),4}$

Stage 3: Look for min distance in the updated distance matrix

⇒ Fuse (3, 5) and (1)

$\boxed{(1, (3,5)) \rightarrow 3}$

So, this is Stage 2 of this clustering algorithm, updation of distance matrix. Now, the distance matrix D is now going to be a 4 by 4 matrix, wherein the distances are the distance is corresponding to the previous terms there. So, we will have from the previous expression only this one case here, one cluster here which is new to us. These 1 2 and 4 are existing clusters. So, there is one new entry here, and these are 1 2 4 which are existing clusters. So, we will have along the diagonal similarly 0s. Now, these distances note that the distance between 2 and 1, between 4 and 1, between 4 and 2 are already present in the previous distance matrix. Distance between 1 and 2 or the clusters which are not yet merged. Distance between 4 and 2 which is going to be given by this; they are already present in the previous cluster.

So, what will be doing is, just to copy them from the previous expression which is the following that, we will have this distance between 1 and 2. Let me just see once again distance between 1 and 2 is 9. So, we just plug in 9 out here; distance between 1 and 4 is an entry which is required 1 and 4 is 6. So, we will have this entry as 6 and similarly this is the distance between 2 and 4 which is 5. Now, these entries here these 3 entries what is this? This is the distance between the newly formed cluster 3 5 and 1, which is not present in the previous diagram there.

From the previous distance matrix, what we had done was since these two were the merged units, we will have to delete all these entries from here; that is what we have to do because those two have been merged. Now, we will have to compute this, we will have to compute this and we will have to compute this. So, what are those things we will require the distance between the cluster 3 5 and 1. Now we are on a single linkage and hence, we would look at the minimum distance between 1 and 3 and the distance between 1 and 5; 1 belonging to this cluster here; 3 belonging to the point here and 5 belonging to the fused cluster. So, from the previous distance matrix, we will actually look at what is d 13? And what is d 15?

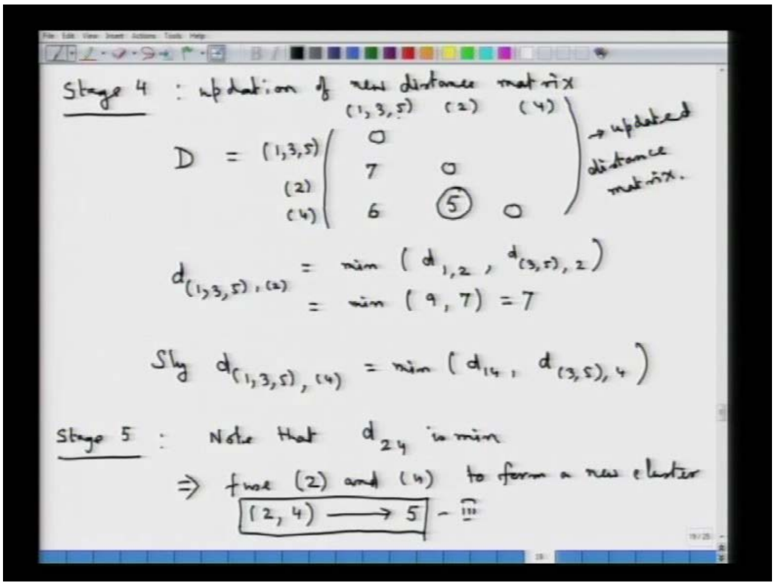
From the given data, they can easily be found out what those quantities are are 3 and 11 and the minimum of that is equal to 3. So, the distance between the newly formed cluster 3 5 and 1 is thus going to be 3. So, we will have the entry here as 3. Similarly, the distance between 3 5, the newly formed cluster and 2 would be the minimum of the two distances, distance between 2 and 3 and the distance between 2 and 5. So, we can find out that also from the distance matrix, the original distance matrix and hence, we will be able to fill up this particular table which would turn out to be 7. Similarly, this distance between the 3 5 new cluster and the cluster which is 4 which is obtained as 8. So, this distance matrix is the updated which is the updated distance matrix.

So, what we said is, if we have an n by n distance matrix to start with at the first step, two cases are going to be fused. We will be having an updated distance matrix which is going to be an n minus 1 cross n minus 1. So, we had here 5 cases. We had started with the distance matrix 5 by 5. Now, we have come down to 4 by 4. So, that completes the Stage 2 here. Now, Stage 3 Stage 3 is basically the second step of iteration, wherein we are now going to form two new cluster or rather we are going to fuse 2 new cluster two old cluster to form a new cluster. Now, we will look at this distance matrix, at the benchmark distance matrix and then, try to find out which of these units are closest to one another.

Now, if you look at closest, we will have this 3 as the minimum that is now coming in this modified or updated distance matrix. So, we look for minimum distance minimum distance in the updated distance matrix and that would lead us to fusing this 3 5 cluster and the single turn cluster which was having case number 1 as the entry. Now, what we have to do is to take a note of this particular fact that, this 1 3 5 are getting merged into one single cluster at what distance. If 1 is very particular, one should actually write it in this way that, 1 is getting fused with the cluster which was containing 3 5; two units at what level, the level is 3. So, this is the second information that, one needs to store the first information.

We had stored that 3 5 merged at level 2 and now 1 and 3 5 were merged at level 3. Now, note that from the previous distance matrix now, neither 1 would be present in the next step of updated distance matrix nor will 3 5 be present. So, we cannot have anything corresponding to this row here 1. This also would vanish and the corresponding column corresponding to this one also will vanish; because we are not going to have one entry there. So, we will have to update the distance matrix. So, that brings us to stage number 4 Stage number 4 updation of new distance matrix. Now, what is going to be the dimension of that particular distance matrix? The previous distance matrix had a dimension 4 by 4.

(Refer Slide Time: 34:58)



Stage 4 : updation of new distance matrix

$$D = \begin{array}{c|ccc} & (1,3,5) & (2) & (4) \\ \hline (1,3,5) & 0 & & \\ (2) & 7 & 0 & \\ (4) & 6 & \textcircled{5} & 0 \end{array} \rightarrow \text{updated distance matrix.}$$

$$d_{(1,3,5),(2)} = \min(d_{1,2}, d_{(3,5),2})$$
$$= \min(9, 7) = 7$$

$$\text{Sly } d_{(1,3,5),(4)} = \min(d_{14}, d_{(3,5),4})$$

Stage 5 : Note that d_{24} is min

$\Rightarrow$ fuse (2) and (4) to form a new cluster

$\boxed{(2,4) \longrightarrow 5}$ - $\overline{\text{iii}}$

So, the new distance matrix which would now be having this 1 3 5 as a standing one cluster and 3 and 4, 2 and 4 to be the other two existing clusters. Can it forward from the previous step of iteration? And hence, we will have in total 3 clusters. Hence, the updated distance matrix is going to be a 3 by 3 matrix; wherein, we will have entries corresponding to now this case number 2 is an existing cluster. So, this would remain as it is and we have a new cluster now which is 1 3 5; this is 1 3 5; this is 2 and this is 4; this two are existing cluster. So, we will have a 3 by 3 distance matrix, wherein we need to compute something and something will be carried forward. What are the things that would be carried forward?

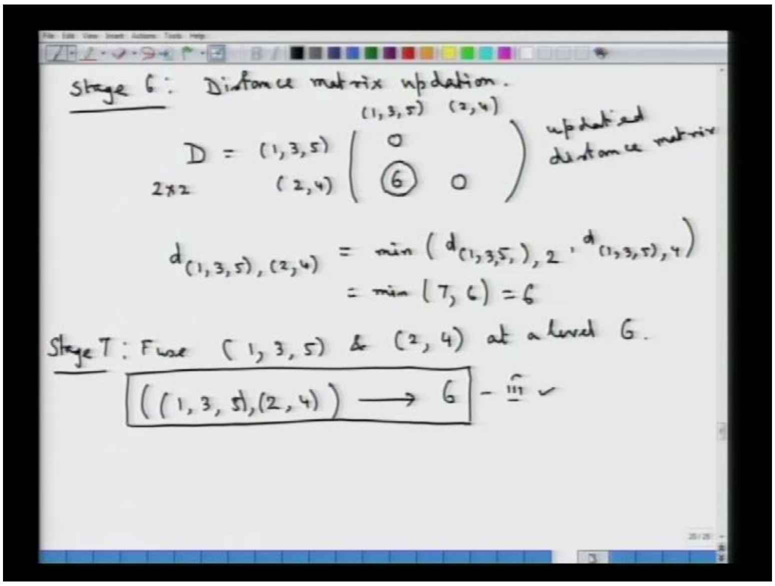
Now, this is an entry which is going to give us the distance between case 2 and case 4 which we will be having from the previous table itself. So, the distance between 2 and 4 or the distance between 4 and 2 are same. So, that would be 5. From the previous table, we will have one entry out here, which is going to be 5. What is this entry? This entry these needs to be computed, this is the distance between the cluster 1 3 5 and the cluster 2, which is an existing cluster. This is a new cluster and this is a distance between 1 3 5 cluster and the cluster which is 4. Now, we are going to compute that, exactly in the same way as what we had done previously. So, this is going to be the minimum of the distance between…

Now remember 1 and 3 5 have been fused. So, one can find the distance between 1 and 2 and the distance between 3 5. Because 3 5 is an identity, which is carried forward from the previous step. So, we can find out what is the distance d 1 2 and the distance d (3, 5), 2. And then, find out what is the minimum of these two distances from the given data? What it turns out is that, this is d 1 2 is given by 9 and d (3, 5), 2 which is coming from the previous table d (3, 5), 2 is 7; this is 7. So, the minimum is 7 and hence, the distance between this newly formed cluster and the old cluster 2 is 7. Similarly, this distance between 1 3 5 and 4 is going to be given by minimum of d 14 and the distance of 3 5 and 4 from the tables, this from the previous table one can similarly compute this as 6.

So, this is now the updated distance matrix at this second stage of iteration updated distance matrix. So, once we have this updated distance matrix, we move on to the third step of iteration. At stage 5, which is basically the third step of iteration, we look at this updated distance matrix and find out, which distance is minimum. Now, we note that this is the minimum, if i is not equal to j from that distance matrix and hence this distance between 2 and 4 is minimum. So, that that would imply that, we are required to fuse this case number 2 and case number 4 to form a new cluster to form a new cluster. So, once we have these to form a new cluster, the information about this fusion of cases and the mutual distance needs to be preserved. Because that is what is going to be required, when we are going to construct the dendogram diagram.

So, we will say that this 2 and 4 has been merged at a level which is at a level of distance 5. So, at the third step of iteration, we have this information to be retained. Look back at the other information, this is the first information that we have retained. This is the second information that we have retained at the second step of iteration; the previous one at the first step of iteration and this at the third step of iteration. Now, what is the situation at the end of this third step of iteration? At the end of the third step of iteration, we have got 1 3 5 to be a one cluster unit and 2 4 to be another cluster unit. So, we have two clusters now. Now, we will have to fuse them together; the only point of interest is to see at what distance level.

(Refer Slide Time: 40:39)



Stage 6: Distance matrix updation.

$$\underset{2\times 2}{D} = \begin{array}{c|cc} & (1,3,5) & (2,4) \\ \hline (1,3,5) & 0 & \\ (2,4) & \boxed{6} & 0 \end{array}$$

updated distance matrix

$$d_{(1,3,5),(2,4)} = \min\left(d_{(1,3,5,),2},\ d_{(1,3,5),4}\right)$$
$$= \min(7, 6) = 6$$

Stage 7: Fuse (1, 3, 5) & (2, 4) at a level 6.

$$\boxed{((1,3,5),(2,4)) \longrightarrow 6} - \underline{iii}\ ✓$$

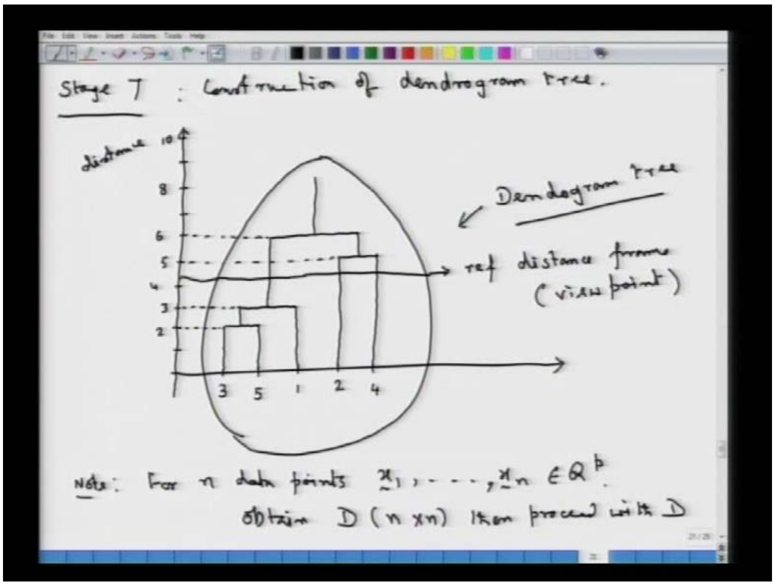
So, we come to this stage 6 now, once again it is after the fusion, we have to look at the distance matrix updation. So, when we look at this distance matrix updation, we will now be having… In the previous distance matrix, it was a 3 by 3, 5 by 4, 4 by 4, 3 by 3. And now, we will be having 2 by 2 distance matrix and what are the entries? The first entry is 1 3 5, that is the existing cluster and we have 2 4 to be the newly formed cluster. So, we will have this 2 by 2 distance matrix. Only one element is important, because the other entries are 0s. Because that is, the mutual distance between 1 3 5 cluster and itself and the distance between 2 4 cluster and itself. So, this is what we need to find out and what is that?

This is the distance between the cluster 1 3 5 and the cluster which is 2 4. Now, how is that going to be obtained from the previous table? We are trying to look at the distance between 1 3 5 and the cluster 2 4. Now, we will be looking at 1 3 5 cluster and an unit taken from the second cluster that is 2 and then, we look at the distance between 1 3 5 and an unit taken from the other unit taken from the other cluster which is unit number 4. So that, this would actually turn out to be, that will be required to find out the minimum of 1 3 5; this is an existing cluster and newly formed cluster unit, that is 2 and the distance between 1 3 5, this cluster and the unit 4 which is now fused with 2.

Now, the entries corresponding to this, can be obtained from the previous table 1 3 5 and 2 that is 7 and 1 3 5 and 4 that is 6. So, we will have the minimum of 1 3 4… Let me go back once again, 1 3 5 and 2 is 7 and the next is 6. So, we will have minimum of this 7 and 6 which is 6, thus this is equal to 6. So, this is at the last step of iteration. This is updated distance matrix, after the fusion has taken place. So, from this updated distance matrix, we will look at this and then, we will any way have to fuse. Because there are two clusters and we have to have one single cluster at the end of the day, at the end of the agglomerative hierarchical cluster analysis. And hence, we see that this natural is the only entry which is also the minimum.

So, we will say that at the last stage which is stage 7, we will fuse this 1 3 5 cluster and the cluster which is containing these two cases 2 4 at a level level of distance 6. So, the last bit of information that we need to keep is that 1 3 5 and this 2 4 all coming into one cluster. Let me still put a back at here. So, that this cluster and this cluster are merged at a fusion level of 6. So, this is at the end of the 4th step. Now, we look back at the output of this particular algorithm. This is one that we are going to take; this is the second input that we are going to take; this is the third input that we are going to take and this is the last input that we are going to take. Now, the thing that remains is the last stage. Stage 7, which is construction of dendogram tree from this output of the algorithm, dendogram tree.

(Refer Slide Time: 44:48)



Stage 7 : Construction of dendrogram tree.

Note: For n data points $x_1, \dots, x_n \in R^p$.
obtain D (n x n) then proceed with D

Now, in order to do that, as I said that we need to collect this entire information. This line is not particularly straight. So, this on this y axis, we have distance being measured and here we will have the case. Let me just mark them, say this is 1 2 3 4 5 6 7 8 9 and 10; suppose these are the distances. So, this is a distance 2; this is a distance 4; this is the distance 6 8 and 10. Now, we look back at the output of the system, this is what we are first going to consider. So, we will fuse to a 3 and 5 at a level 2. At a level 2, this case number 3 and case number 5 are the ones to be fused at this particular level which is 5 level 2. So, we will have this is the first input basically, that is what we have from this algorithm; 3 5 merging at distance level 2.

What is the next? 1 3 and 5 merged at level 3. So, how to represent that? 3 and 5 are already there. So, you will have this as case number 1. Now, we will have this being fused at what level? I just need to look at that, at level 3. So, this is the distance. So that, we will have corresponding to this distance 3, these 3 cases one is getting merged with 3 5. So, this is that at level 3. So, this was if this was a first step of iteration being represented in the dendogram; this is the second step of iteration. Now, what is the third step iteration? This is the second step. At the third step of iteration, two new clusters 2 and 4 are merged at a level 5. So, there are two new cases 2 and 4; these are two new cases.

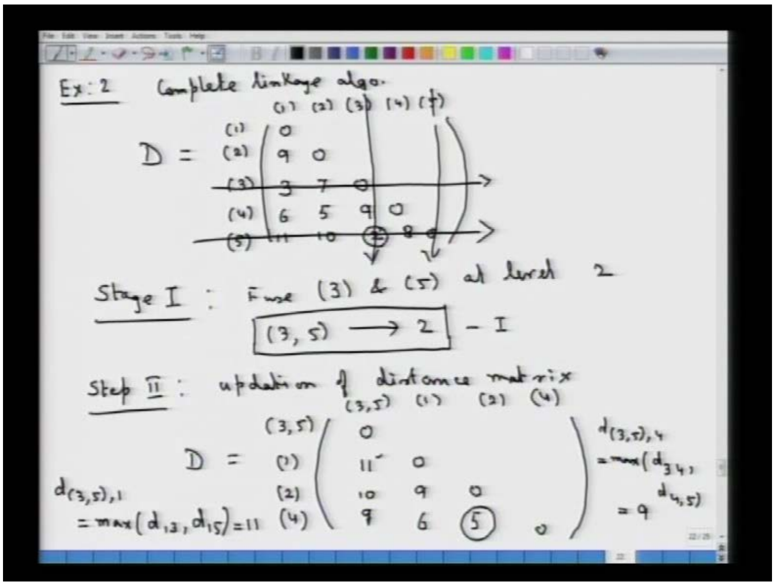
They are going to be merged at this level, which is distance level 5. So, we will have the two branches corresponding to these two cases 2 and 4 getting merged at this level 5. So, that is the third iteration input that is, what we have got and this is 3; this is not 3; once again this has to be the 4th step of iteration. So, this is the 4th step of iteration. We will see that, 1 3 5 an existing cluster and 2 4 another existing cluster is now getting merged at a level 6. So, this is an existing cluster, 2 4 this is another existing cluster. They are different up to the distance level which is 6 here and then getting merged to form the single cluster. So, this is the distance level 6 at which they are getting merged.

So, the figure that we have obtained here is the dendogram tree. Now, this dendogram tree is to be interpreted exactly in the same way as, what we had seen for the general discussion on such interpretations of dendogram tree. For Example, if we put our reference frame somewhere here, say suppose this is the reference frame, reference distance frame or the view point. So, from this dendogram tree, if we say that 4.2 is in or 4.2 or 4.3 is a distance, at which we are going to look at the hierarchical cluster. So, the clusters below those are one single turn cluster which is 4; another single turn cluster which is 2 and there is one big cluster which is having cases 1 3 and 5.

So, as you can see, this constructed dendogram now is a perfect example of a hierarchical structure; wherein, it is a tree structure in the tree structure in the sense that you can look at this particular line to represent a branch of a tree below which all the cases are sub branches. If you look at this particular level then, there are two branches of the tree. In one branch, there are two cases 2 and 4. In the other branch, there are three cases which are 1 3 and 5. If we truncate the tree at this particular level, there are three branches coming from that particular level which is having 4 2 and 1 3 5 in three different clusters. So, that is how a dendogram is constructed for a given higher dimensional multidimensional data.

We can actually replicate the algorithm, the numerical algorithm that is what I try to illustrate in order to get to such a dendogram tree. With only one thing to remember that when we were looking at this particular example, we started with the distance matrix. For any practical data set, what will be having is say x 1, x 2, x n. If we have this… Let me write it in a separate page for n data points n data points x 1, x 2, x n. These are all multidimensional data say these are belonging to R to the power p. From these data, obtain the distance matrix; obtain this n by n distance matrix; there are n cases. So, n by n distance matrix and then proceed with D to update this hierarchical clustering dendogram tree structure.

(Refer Slide Time: 52:03)



Ex: 2 Complete linkage algo.

$$D = \begin{array}{c|ccccc} & (1) & (2) & (3) & (4) & (5) \\ (1) & 0 & & & & \\ (2) & 9 & 0 & & & \\ \cancel{(3)} & \cancel{3} & \cancel{7} & \cancel{0} & & \\ (4) & 6 & 5 & 9 & 0 & \\ \cancel{(5)} & \cancel{11} & \cancel{10} & \boxed{2} & \cancel{8} & \cancel{0} \end{array}$$

Stage I : Fuse (3) & (5) at level 2

$\boxed{(3, 5) \longrightarrow 2}$ – I

Step II : updation of distance matrix

$$D = \begin{array}{c|cccc} & (3,5) & (1) & (2) & (4) \\ (3,5) & 0 & & & \\ (1) & 11 & 0 & & \\ (2) & 10 & 9 & 0 & \\ (4) & 9 & 6 & \boxed{5} & 0 \end{array}$$

$d_{(3,5),1} = \max(d_{13}, d_{15}) = 11$

$d_{(3,5),4} = \max(d_{34}, d_{4,5}) = 9$

Now, what we will do is that we will look at example number 2; wherein, I will try to look at a complete linkage algorithm being implemented. Let us start with this distance matrix which I think is the same as what we started with the single linkage. So, the entries here are 0 9 0 3 7 0 and then, we have 6 5 9 and then we have 11 10 11 10 2 this has to be 0 2 8 and 0. So, these are the different objects first object, second object, third multidimensional object, forth multidimensional object and fifth multidimensional objects object. So, we will have 1 2 3 also along the columns. So, this is what the distance matrix is. So, from this distance matrix, we are going to obtain a complete linkage agglomerative hierarchical cluster dendogram tree.

Now, what we are going to do, I am going to write less as what compared to the previous example. So, at the stage 1, we look at this distance matrix and look at which of the two cases are most similar. So, we see that this d ij is minimum except those which are all diagonal; because that does not make any sense. We will have this as the minimum which is distance between 3 and 5. So, we will have this fuse 3 and 5 at level 2. So, we have the first output, which is 3 5 getting fused at level 2. Up to this particular point, the algorithm has not at all differed from the type of algorithm which is used for the single linkage.

Now, when we have this, we need to go to the second step of this algorithm which is updation of the updation of this distance matrix. Now, the new distance matrix will have an identity, which is 3 5 and then we will have the existing cases which is 1 2 and 4. So, this is that 3 5 1 2 and 4. Now, as before we will have some entry is coming directly from the previous table like the distance between 2 and 1 distance between 2 and 4. So, those are going to be exactly what we have in the previous structure which is what we have as the following that this is 9 6 and 5. So, these are what are coming from this particular table itself.

Now, in order to update the distance what we say that this cannot be present now. We have to delete this. We have to delete this particular term also and also the row wherein 3 is present. So, this also gets deleted and then this also will be deleted. So, we will have these remaining 9 6 and 5 as you can see 9 6 and 5 coming directly from there. These two are the quantities that we need to compute. What are those? That is the distance between 3 5 and 1. Now, we are on a complete linkage. So, we will find out what is the distance what is the maximum distance between any object taken from the one group there. So, d 13 and d 15. So, d 1 3 is 3 and d 1 5 is 11.

So, the maximum of 3 and 11 would just be given by 11. So, this under a complete linkage would now be given by 11 which is different from what it was for the single linkage. Similarly, this particular element here we need to find out what is the distance between 3 5 and 4. And that is going to be the maximum distance between distance 3 4 and distance 4 5 which are coming from the previous table that is 10. The maximum is going to be given by this 3 5 2 would be 10. And then, another entry is required which is distance between 3 5 and 4, which would turn out that it is 9.

So, we have these three new entries, which this is giving the distance between 3 5 and 1, which comes from the updation of the distance matrix. Distance between 3 5 and 2, distance between 3 5 and 4 which is 9; the maximum distance between 3 4 and 4 5. So, that, we will have 3 5 and 4; this is equal to 9, and the other one will be equal to 10. So, from distance this distance matrix, now would be used when we are now looking at fusing two new cases. As we will see here that this is the minimum minimum among the remaining d ij and hence the two cases are going to be fused. And then, we will construct the dendogram corresponding to this this example in the next Lecture.

Thank you