Applied Multivariate Analysis

Prof. Amit Mitra

Prof. Sharmishtha Mitra

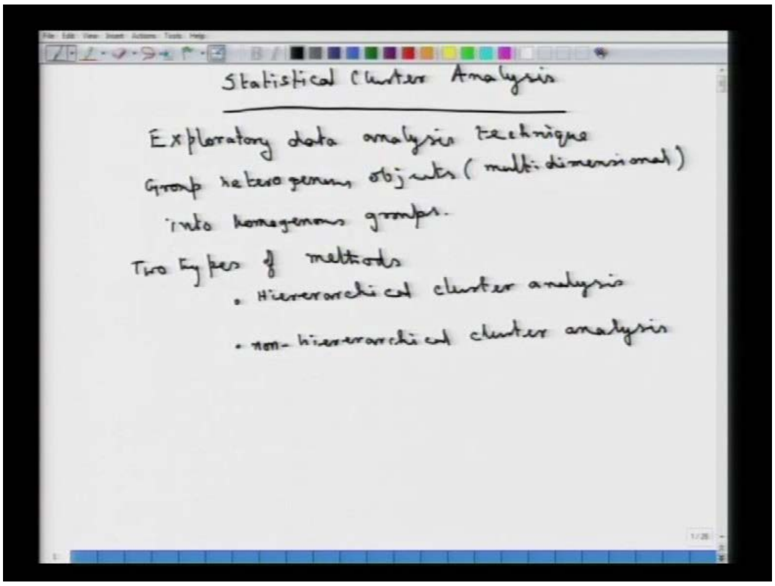
Department of Mathematics and Statistics

Indian Institute of Technology, Kanpur

Lecture No. #26

Cluster Analysis

(Refer Slide Time: 00:24)



Statistical Cluster Analysis

Exploratory data analysis technique

Group heterogeneous objects (multi-dimensional)

into homogeneous groups.

Two types of methods

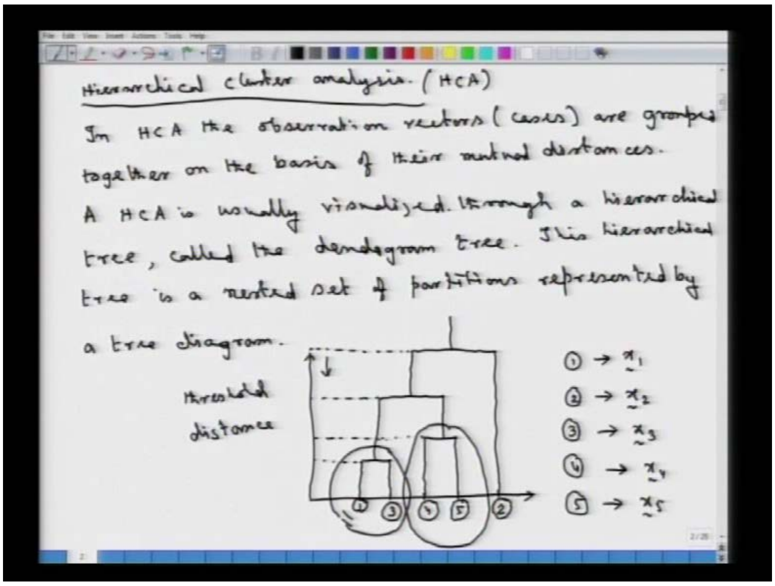
- Hierarchical cluster analysis
- non-hierarchical cluster analysis

In this lecture, we start looking at statistical cluster analysis analysis techniques. So, statistical cluster analysis technique is basically trying to do the following job. Statistical cluster analysis: So, in statistical cluster analysis, what we try to do is we try to group heterogeneous multi-dimensional objects into homogeneous groups. So, that is a basic objective of statistical cluster analysis, and what we have here is the following that this statistical cluster analysis technique can be defined as a way of exploratory data analysis. So, it is basically exploratory data analysis technique, wherein we try to group heterogeneous objects heterogeneous objects or cases; these are multi-dimensional these are multi-dimensional objects into homogeneous groups. So, that is a basic objective of statistical cluster analysis.

Now, broadly speaking there are two types of such methods two types of methods. The first one is what we call hierarchical cluster analysis hierarchical cluster analysis technique, and the second one is a non-hierarchical cluster analysis. So, in the first group of methods, which we term as hierarchical cluster analysis; they basically form clusters wherein the clusters have an inherent hierarchical structure in the formation and in the interpretation of the clusters. Now, when we talk about the other way which is the non-hierarchical cluster analysis technique; there we form clusters of homogeneous objects which are as similar as possible within each cluster and as different as possible when we look at from inter cluster distances.

So, they are as far as possible, when we talk about objects in two different clusters. However in non-hierarchical cluster analysis method base method, we do not have any hierarchy in the formations of the clusters. So, let us look at what type of how this hierarchical cluster clustering techniques can actually be performed and how this non-hierarchical cluster analysis technique also can be put forward. Now, let me give some definitions; then I will explain.

(Refer Slide Time: 03:30)



Hierarchical cluster analysis. (HCA)

In HCA the observation vectors (cases) are grouped together on the basis of their mutual distances. A HCA is usually visualized through a hierarchical tree, called the dendogram tree. This hierarchical tree is a nested set of partitions represented by a tree diagram.

$1 \rightarrow \underset{\sim}{x}_1$

$2 \rightarrow \underset{\sim}{x}_2$

$3 \rightarrow \underset{\sim}{x}_3$

$4 \rightarrow \underset{\sim}{x}_4$

$5 \rightarrow \underset{\sim}{x}_5$

Hierarchical cluster analysis: Now, let me give some basic points in hierarchical; let me call this as HCA. In hierarchical cluster analysis technique, the observation vectors observation vectors which basically are cases are grouped together on the basis of self-similarity or on the basis of their mutual distances. A hierarchical cluster analysis technique is well when we talk about that observation vectors are grouped together on the basis of their mutual distances that ofcourse is true, even if we look at non-hierarchical clustering. However, when we look at hierarchical clustering method, the following is going to be true.

This is usually visualized almost almost actually usually visualized through a hierarchical tree which is called the dendogram or the dendogram tree; wherein, actually this dendogram tree or the hierarchical tree this hierarchical tree is a nested set of partitions of the data of partitions, which are going to be represented by a tree diagram. Now, how does that tree diagram look like? The tree diagram looks like the following, suppose this Y axis is basically something which we going to discuss its threshold distance. So, if we have here we have a tree like the following form; suppose I have case number 1 here, case number 3 here, 4, 5. Let us consider 6 cases.

So, we might see in such a diagram of this form; say given by this type of tree structure. Now, what is an interpretation of a tree structure that we have drawn out here; it basically looks at cases. So, there are 6 cases, we are trying to form clusters among there are 5 at the moment. Let me write that as, this 1 as 2. So, these 5 cases are going to be put into various clusters using a hierarchical cluster analysis technique. A tree diagram, which is the output of such an hierarchical clustering method looks like the following. Now, it has the interpretation that case number 1 and case number 2 or all these cases are multidimensional.

So, suppose this case number 1 has got a characteristic vector which is x 1; case number 2 similarly case number 2 has got a characteristic vector which is x 2; similarly, all the cases have their characteristic vectors; x 4 is the characteristic vector corresponding to case 5 and x 5 is the characteristic vector say p dimensional vector. Now, we say that this number case number 1 and case number 3 actually come together to form the first cluster among all these 5 cases. So, they are merged at this particular level here. Then case number 4 and case number 5 are the next two cases that can be merged to form one single cluster and that at this particular distance out here.

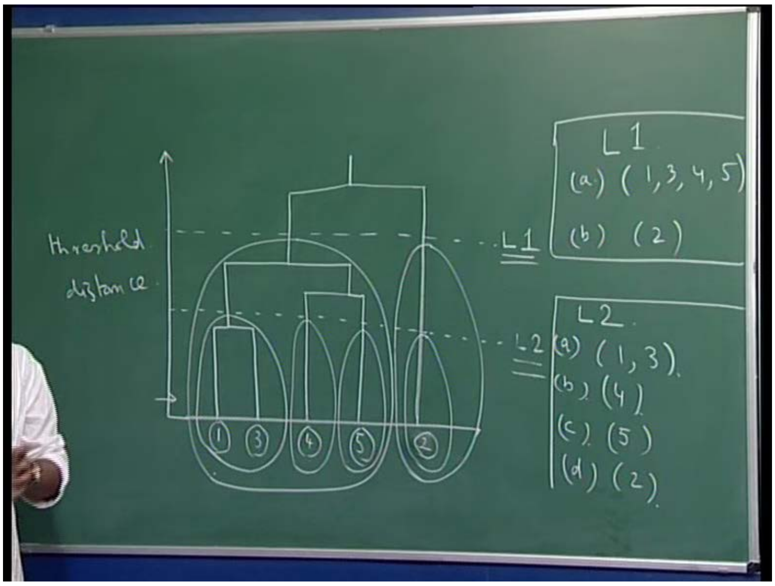
So, the distance level if you increase the distance level, then we will have these two forming one cluster. If we go higher up through the threshold distance, we will see we can see that this 1, 3, 4 and 5 come together at this particular level out here which is the level at which 1, 2 which are forming a cluster out here. So, this is one cluster that has formed first of all. Then there is a cluster which is formed between the cases 4 and 5.

These two clusters are different clusters merged at different levels. They come together at this level and then they merge to form one big cluster at this particular threshold distance. And then if we still move up by increasing the threshold distance, then what we see is that case number 2 also merges with that cluster which was formed from the cases 1, 3, 4, 5 and 2 coming together at this particular level.

So, we will have at this level out here of the threshold distance, all the cases forming in one cluster. So, it is basically trying to see it is the basically level of resolution with which we are going to look at these cases look at these multidimensional cases. Now, if we look at heterogeneous objects from a very far away distance, we will see that all the cases almost look alike. So, all the cases would appear as if they are in one single cluster and that is basically this particular level if we are looking at a higher distance, then all the cases 1, 2, 3, 4 and 5 look as if they form one cluster. If we make our resolution our resolution of looking at the data finer, if we come down in the threshold distance here, we will see more and more clusters emerging out from that particular data.

And thus, if we come down to this level and fix our frame of reference at this particular point, we will see that there are two distinct clusters in the data. As we go down in the distance metric level, we will see more and more clusters. And as a result of which, if we are looking at the objects at a very fine level of resolution, then all the objects will appear as if they are different and they do not form into one cluster. A nice thing actually which can be said about such a dendogram tree are the following that if we look at such a dendogram tree, it actually gives us a following interpretation. Let me go to the board and then try to explain this particular thing, the type of figure that we had we have drawn there can be represented here.

(Refer Slide Time: 11:18)



So, we have got these cases; we have got the x axis out here which is the case axis and what we had there was this case number 5 out here. Then we had two branches; the two branches, one of them was continuing 1 and 3 perhaps there, which was the first two get formed into one cluster. So, this is the case number 1; this was case number 3 if that is there. So, this is 1, 3, 4 and 5 here and we had this as 2. This axis is basically the threshold distance axis. Now, if this is what the dendogram tree looks like, then if we fix a reference point at one distance here. So, if our reference point is fixed at this threshold distance and if we are looking at from such a distance, we will see at for any threshold distance, we will see that there are k groups or g groups that would come out corresponding to such a partition.

Now, if we have our reference point here, what are the two clusters? There are two clusters. Clusters basically are the cases which fall under the lines which pass through that. So, we will have one cluster here and we will have one cluster here. This is cluster number 1 which has cases. So, suppose this is level one. So, corresponding to level 1, we will have two clusters. Cluster number a which is containing the cases 1, 3, 4 and 5 and the second cluster which is having case number 2. So, this is what we obtained for the level of resolution of the threshold distance at the point L1. Now, if on the other hand, now these two are going to be two disjoint subsets of the data; that is the data cases 1, 2, 3, 4, 5.

So, there is no intersection between the cases that we have within the first cluster and the cases which we have within the second cluster. What more important is that, if we look at another level of resolution, say this one. So, this is another threshold distance that we are looking at now. So, say this is a level which is L2. So, if we fix our resolution level at L2, then what are the clusters we get below this particular threshold distance line? We see that there is one branch coming out here, which is this one. So, corresponding to now my L2 level of resolution, we will have the first sector to give us the two cases 1 and 3, then we come down to the other point. This is the single branch here, which is containing this case 4.

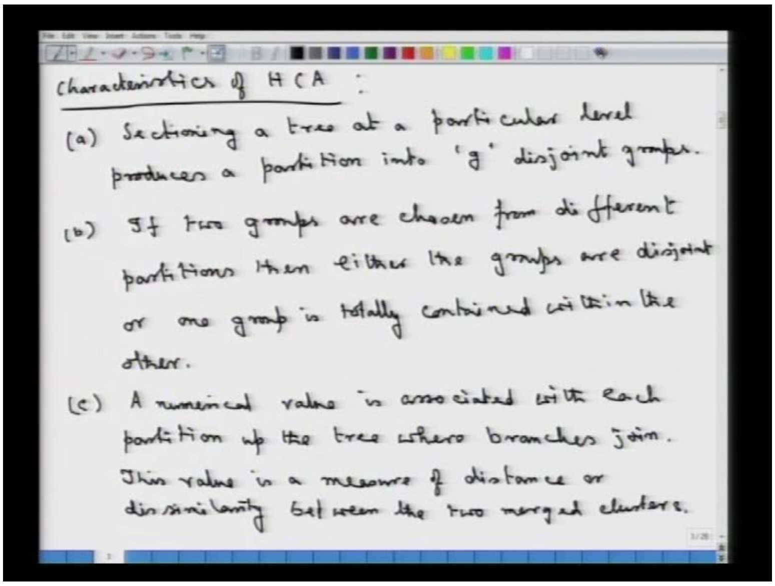
So, we will have this case number 4 within that particular branch. Then there is another branch which intersects with that L2 level, which is having a single branch; single element actually that is 5. So, the third cluster is containing the case 5 and then from here, there is once again a single case branch which is giving us the fourth cluster, which is cluster number 2. Now, if we thus look at various levels of resolution, if this is the level of resolution, we will say that all the cases belong to one single cluster; that is what I was referring to that if we look at heterogeneous group of objects from a far away distance, all the objects would look as if they are similar.

So, they all of them form one cluster. As you go down, as you make the resolution level finer and finer, you will have more and more clusters. If you come down to this particular level here of resolution, so that threshold distance is very low ; you will accordingly have a similar line to what we were having L1 and L2. Then you will see that all the cases are separate clusters. So, if you have the resolution level so fine, then all the cases would look different and what is interesting to note from these two clusters that have been formed from the one L1 level and L2 level.

If you choose any two clusters from the two sets, the clusters formed by L1 and the clusters formed by L2, you will see that either the cluster a particular cluster here would be completely contained in the other or it will be disjoint with that; because if we just look at 1,3 cluster which is from L2, so we will see that this 1,3 cluster which you get here is the proper subset of this particular cluster, that is coming from L1. If we choose two clusters say this and this cluster here, you will see that the two are disjoint. So, either they are going to be disjoint or one would be completely contained in other. So, if we look at this cluster and this cluster they are going to be disjoint; because we are looking at two different clusters.

Or if we look at this and this, they are identically the same. So, the two sets are exactly the equal. So, that is how and why it is called after all a hierarchical clustering? Because there is a there is an hierarchy in the formation of the clusters in this particular data; because you will see that these two form one cluster. So, it looks like a tree structure; because if this is the main branch of the tree, you see that the main branch is out into two parts and there are sub branches within those branches and thus, it has got an interpretation of a tree like structure and that is why such a diagram is called a dendogram tree diagram. So, that is the hierarchical structure of this particular point hierarchical clustering.

(Refer Slide Time: 17:41)



Characteristics of HCA :

(a) Sectioning a tree at a particular level produces a partition into 'g' disjoint groups.

(b) If two groups are chosen from different partitions then either the groups are disjoint or one group is totally contained within the other.

(c) A numerical value is associated with each partition up the tree where branches join. This value is a measure of distance or dissimilarity between the two merged clusters.

Let me try to put forward the type of things that I try to explain. So, these are basically the characteristics characteristics of this hierarchical cluster analysis technique. The things that I explained let me just put forward; point number a is that if we are going to section sectioning a tree at a particular level at a particular level, produces a partition into g is the number that is related to actually the type of partitions that we are going to have there at a particular threshold distance level. So, these are these are going to be organized in to g disjoint groups that is what we have seen from the figure.

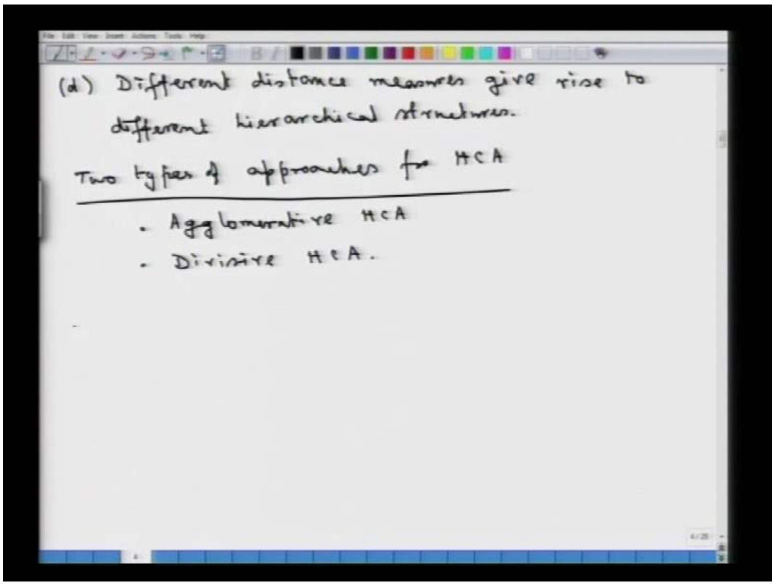
Number two: The second point to be noted is that if two groups are chosen from different sections different partitions, then either the groups are disjoint that is what we have seen. So, either we will have the groups to be disjoint or one group is totally contained within the other contained within the other. So, these two points basically are the points that we were trying to explained through this particular figure that if now you choose a particular threshold level here, this produces two clusters. If you go down to L1 L2 here, this is going to produce 1, 2, 3, 4 such clusters. If you come down here, you will have 5 such clusters.

So, the number of clusters that you are going to get, when you are looking at the threshold distance to decide what is the partition of the data; that is what we will be having the g number of groups; that is what I had written there. And the second point is that if you look at two different levels, then the clusters that are formed in this L1; corresponding to this, there are two clusters and corresponding to L2, there are four clusters. These are either going to be disjoint or one would be totally contained within the other. Now, there are other characteristics which are easy to understand.

I will just write that a numerical value is associated is associated with each partition up the tree. So, the structure is the tree structure; up the tree where branches join together. Now, this value is a measure of distance or dissimilarity dissimilarity between the two merged clusters. Because that is what we have seen that if I said that from this particular figure that if we are looking at this level of resolution, then here here at this particular point all these four cases are merged. So, at a higher distance all these cases are merged; at this distance so on the y axis here, the distances are measured.

So, corresponding to each merger, we will have a distance that signifies at what level the corresponding cases are joined together here. So, at a much lower distance, these two cases are joined; at a bit higher distance, these two cases 4 and 5 are joined; even higher distance 1, 3, 4, 5 are joined and at such a distance, all the cases are joined together to form one single cluster; that is what is written in this c point here that a numerical value is associated with each partition up the tree, where branches joined. This value is the measure of the distance or the dissimilarity between the two merged clusters.

(Refer Slide Time: 22:56)



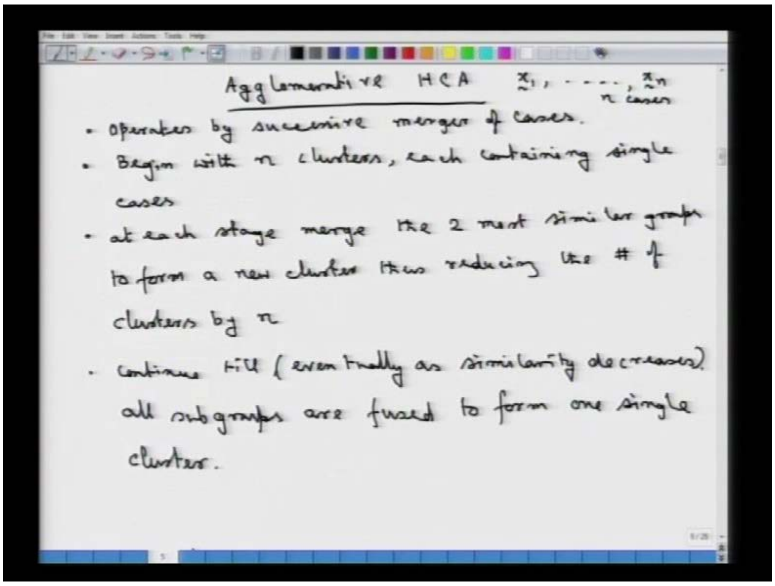
(d) Different distance measures give rise to different hierarchical structures.

Two types of approaches for HCA

- Agglomerative HCA
- Divisive HCA.

Now, the characteristic number 4 is the following that different distance measures ofcourse are going to give us different clusters; different distance measures give rise to different hierarchical structures. Now, that is the basic type of characteristics of any hierarchical clustering method. Now, let us discuss about the two types of hierarchical or rather two types of algorithms that are usually used in order to frame such hierarchical clustering. Now, the two types of two types of approaches for these hierarchical cluster analysis formations are the first one is called agglomerative hierarchical clustering method and the second one is called the divisive hierarchical clustering method. So, the first one is agglomerative hierarchical cluster analysis algorithm and this is the second one is what is called the divisive hierarchical clustering analysis technique. Now, let us discuss one by one these cases.

(Refer Slide Time: 24:28)



Agglomerative HCA $\underset{\sim}{x}_1, \ldots, \underset{\sim}{x}_n$
n cases

- operates by successive merger of cases.
- Begin with n clusters, each containing single cases
- at each stage merge the 2 most similar groups to form a new cluster thus reducing the # of clusters by n
- continue till (eventually as similarity decreases) all subgroups are fused to form one single cluster.

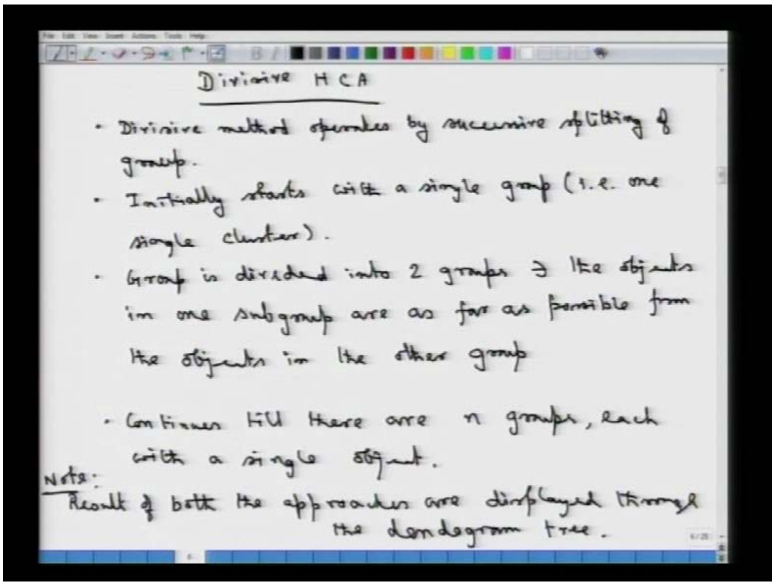
Let us now concentrate on first, agglomerative hierarchical cluster analysis algorithm. Now, it is going to be based on the following points that the first of all this operates by successive merger of cases operates by successive merger of cases that is the first point; because, it is agglomerative in nature. So that, what we do in the agglomerative clustering method is we start with n cases; suppose I have got n cases, n cases are put in all different clusters. So, for n number of data points, n number of cases, we will have n clusters to start with and then we keep on joining keep on merging cases after cases, where the first step of the iteration we will have the two most similar cases joined together to form a one cluster.

And then the two cases merged at that particular distance level, we will have a single cluster and the rest of the n minus 2 cases will form n minus 2 clusters. So, from n clusters in the data, we will move on to n minus 1 clusters in the data and then at the third step of iteration, we will consider these n minus 1 cases to be n minus 1 clusters. And then we will have to form different we will have to form the distance matrix between all these n minus 1 cluster and then we will proceed to the next step to see which of these clusters are closest actually. And then we will merge two next two set of clusters and then this process is going to proceed until we have all the cases in one single cluster.

So, it goes on along the line that it starts with n cases goes on merging cases one after the other. So, it is on a mode of successive merges until we come to a point that all the n cases are put together in one single cluster. So, I just write the steps here that this ofcourse operates by successive merger of cases that is what I try to explain and begin with n clusters, each containing single points each containing single cases. So, to start with, we have got so these are the data x 1, x 2, x n. These are the n cases and then if we on a agglomerative HCA, then we start with n clusters each of these objects in one single cluster. Now, what we do here is that at each stage, merge the two most similar the most similar nature ofcourse is going to be defined shortly two most similar groups to form a new cluster; thus reducing the number of clusters by 1 reducing the number of clusters by 1.

So, from n clusters we will come down to n minus 1 cluster; (( )) because, I have already explained that two most similar groups are merged. And then we will have one cluster being reduced from these n clusters and thus we will have n minus 1 cluster at that point. Continue this process, continue till eventually as similarity decreases eventually as similarity decreases, all sub groups are fused together to form one single cluster. So, start with n such clusters go on merging, then the type or rather the algorithm for merging and finding out means the steps associated we will see in that algorithmic formulation. So, this is how an agglomerative hierarchical clustering method is going to be implemented.

(Refer Slide Time: 29:32)



Divisive HCA

- Divisive method operates by successive splitting of group.
- Initially starts with a single group (i.e. one single cluster).
- Group is divided into 2 groups ∋ the objects in one subgroup are as far as possible from the objects in the other group
- Continues till there are n groups, each with a single object.

Note:
Result of both the approaches are displayed through the dendogram tree.

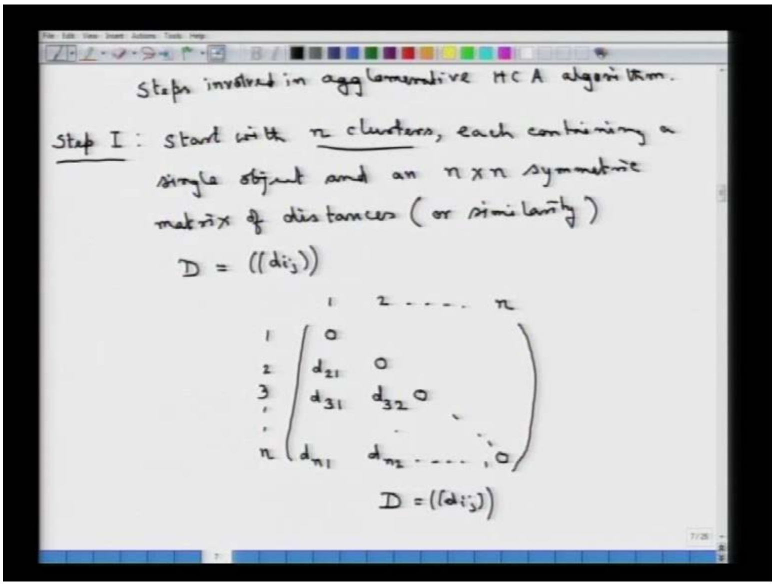
Now, what about the other type of hierarchical clustering method that I said it is what I said as divisive hierarchical clustering cluster analysis; it goes exactly in the opposite direction. When an agglomerative hierarchical method starts with n cases and then go on merging, until you have one single cluster wherein all the objects all n objects are put together. The divisive hierarchical clustering method goes in the opposite direction. So, it starts with one single cluster, wherein all the cases wherein all the n cases are put together. So, it starts with one single cluster and then goes on branching those cases or dividing those cases into homogeneous groups of objects.

And thus, we will have starting from one single cluster wherein n objects are put together by successive dividing or splitting of the cases or the groups actually. We will finally come down to n clusters, wherein n clusters will contain a single point. Each of the objects will now form will be the member of all those n clusters. So, the steps involved can be highlighted in the following way. So, let me just write that this divisive method or divisive approach operates by successive splitting of groups. So, that is the first thing first point to be noted. This initially starts with a single group with a single group; that is one cluster one single cluster, where all the objects are put together starts with one single cluster.

Then this group is divided divided in to two subgroups mutually disjoint such that the objects in one group one of these subgroups are as far as possible are as far as possible from the objects in the other group from the objects in the other group. So, at the first step we try to split the single group into two groups. So, that the objects in one group look as much different as possible from the objects, which are there in the second group of objects. So, this process is going to be continued till we will have all the objects in individual clusters; continues till there are n groups of objects each with a single object. So, that is basically the type of approach that is followed in a divisive hierarchical cluster analysis method.

Now, the output of both these agglomerative hierarchical approach and divisive hierarchical approach are represented through what we have the dendogram diagram. Results of both the approaches are displayed through the dendogram tree. So, both of them are going to be through the dendogram tree. It is said that the divisive hierarchical clustering method is computationally not as efficient as that of agglomerative clustering. An agglomerative hierarchical clustering approach is actually the type of approach that is usually more frequently adopted approach. We will look at the steps associated with such an algorithm, which eventually is going to lead us to agglomerative hierarchical clustering method in detail and the type of distance measures also that are going to play an important role in such cluster analysis formulation.

(Refer Slide Time: 34:47)



Steps involved in agglomerative HCA algorithm.

Step I : start with n clusters, each containing a single object and an n x n symmetric matrix of distances (or similarity)

$D = ((d_{ij}))$

$$\begin{array}{c|cccc} & 1 & 2 & \cdots & n \\ \hline 1 & 0 & & & \\ 2 & d_{21} & 0 & & \\ 3 & d_{31} & d_{32} & 0 & \\ \vdots & & & \ddots & \\ n & d_{n1} & d_{n2} & \cdots & 0 \end{array}$$

$D = ((d_{ij}))$

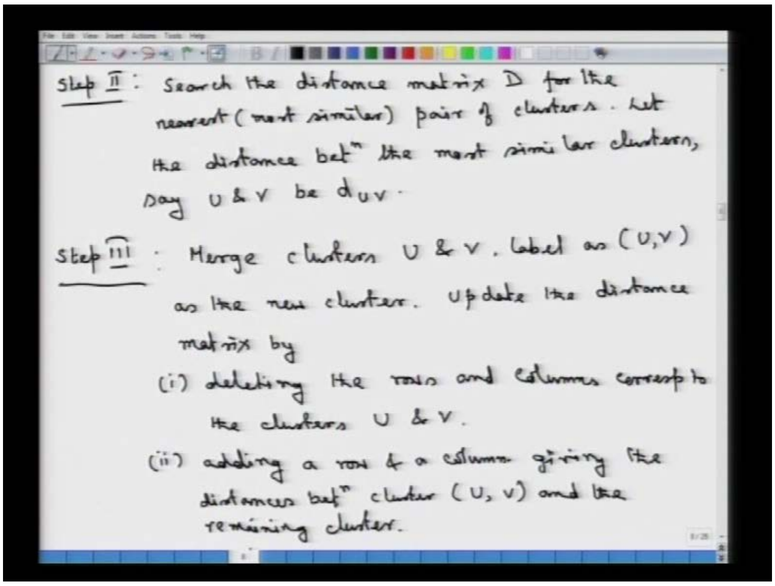
Let us now look at agglomerative clustering in more detail. Steps involved in agglomerative hierarchical cluster analysis algorithm: So, we look at the computational steps; these basically these are the computational steps. Now, the basic philosophy of the agglomerative hierarchical clustering method is that we will be looking at the clusters or rather to start with we will be looking at the n objects being members of n different clusters. So, that is the step 1 and we need to have at the starting point, some measure of the distances the mutual distances between these n cases.

Now, that distance can be Euclidean distance; it can be any other distance measure that one can introduce. So, at the step 1, we do the following that we start with n clusters each containing single object each containing a single object and an n by n symmetric ofcourse that is going to be symmetric symmetric matrix of distances. One can have a distance matrix or one can have a similarity matrix depends on the problem or similarity distances, say that is given by a matrix D which contains the entries d i j. Now, what is this D matrix here? D matrix is the following that it gives the mutual distance between the objects that are present.

So, there are n objects. So, suppose these objects are 1, 2, 3 up to n. So, we form this particular distance matrix D. So, this is the distance matrix D, which is holding this d i j entries here. This is the symmetric matrix wherein, if you look at the diagonal element, it is the distance between 1 and 1. So, that should be logically zero. So, all the diagonal entries of this distance matrix is going to be 0 and then this measures this gives us the distance between the first and the second object. So, this I can say it is d 2 1. This is going to be for the third object.

So, this is going to be the distance between the third object and the first object; this is between the third object and the second object; and ofcourse, the distance between the third object and itself would be 0; because it is the mutual distance of that object from itself; like that you will have all the entries out here. So, this is the distance between the n th object and the first object d n 2 and so on; the this last entry would be d n n minus 1. So, it is the distance between the n th and the n minus 1 th object. This is the symmetric matrix. So, exactly these entries find the place here; because the distance between 2 and 1 is the same as the distance between 1 and 2 the two objects.

(Refer Slide Time: 38:45)



Step II : Search the distance matrix D for the nearest (most similar) pair of clusters. Let the distance bet^n the most similar clusters, say U & V be d_{UV}.

Step III : Merge clusters U & V. Label as (U,V) as the new cluster. Update the distance matrix by

(i) deleting the rows and columns corresp to the clusters U & V.

(ii) adding a row & a column giving the distances bet^n cluster (U, V) and the remaining cluster.

So, we start with this particular n clusters, wherein each containing a single object and then a distance matrix that we are going to compute from the multidimensional data. Once we have the distance matrix, we move on to the second step of this agglomerative hierarchical clustering method. From the distance matrix from this distance matrix, we will look at which of these n objects which two of these n objects are closest to one another. In other words, we will look at from these distances from these mutual distances, which give us distance between any two objects in this particular set and then the minimum is what is sort after.

So, if we see that, this d 32 is the minimum among all these entries here, we will see or rather we will inferred that the object number 3 and object number 2 are closest to one another in terms of all these mutual distances. And hence, we will merge those two cases for which the mutual distance is the minimum. So, that is the second step. So, at step 2, we will search the distance matrix search the distance matrix D for the nearest for the nearest which is most similar; pair of objects or pair of clusters. Remember that, at the first step we are started with n clusters, so we are looking at 2 points at the first step points of clusters. Let the distance between the most similar clusters say U and V two points be denoted by d u v.

So, this d u v is the distance which is the smallest and hence, we will merge the clusters U and V together to form a new cluster. Since we have merged U and V, we will now be having from n clusters we will have n minus 1 cluster now. So, suppose that is this; so when we move on from this step 2, what do we have? We have n minus 1 clusters. Now the previous distance matrix that is what we started with D; these needs to be updated, when we have n minus 1 clusters; because this was the distance matrix corresponding to n clusters. Now, we have merged two cases and we will now be having n minus 1 cluster and we will have an n minus 1 cross n minus 1 distance matrix which has to be computed and that is the updation step of the distance matrix; that is what is step three.

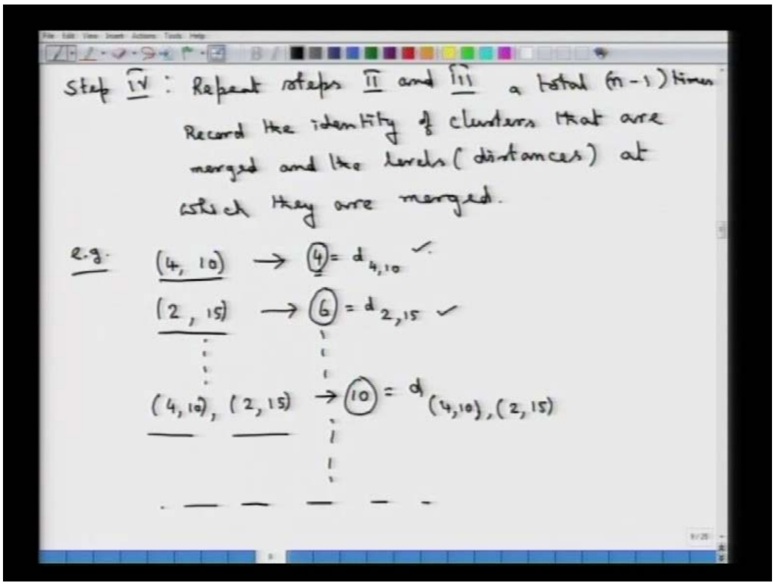
So, this merge clusters U and V label as say (U, V) whatever as the new cluster and then we will have to do this distance matrix updation. So, update the distance matrix by doing the two following things. Number 1: note that, when U and V have been merged together U and V well when we had that n by n distance matrix D, U and V were two separate identities. Now U and V have been merged together and hence, the rows and columns corresponding to U and V have to be deleted from the original distance matrix. So, one has to delete rows and columns, wherein U and V are present and then one has to actually add this (U, V) as the new case. And then find out the distance of this newly formed cluster with the other single point clusters that is present in the data.

So, the first step is that, we will have to delete that. So, this updation is done by deleting the rows and columns, corresponding to the clusters U and V. So, that is the first thing one has to do and the second thing is done by adding a row and a column adding a row and a column giving the distances giving the distances between the newly formed clusters (U,V) and the remaining clusters in that distance matrix. So, this is what I have already explained that those do not have any presents U V; because U V now have been merged. So, the distance is corresponding to U and V separate identities have to be deleted and then another row and column needs to be added; because we now have a new identity which is the merged identity of the U and the V points.

So, once we have that updation done, what we will be doing here? Now at this particular step here when we are at this particular point, we will have a distance matrix which is going to be an n minus 1. So, this will produce a distance matrix, which is going to be n minus 1 cross n minus 1; because error n minus 1 clusters. So, what we will do now? We will go back to step 2 and then see which two cases can now be merged now be merged in terms of those two cases coming together to form one single cluster. Now, that cluster can be joined with (U, V) or it can be two different cases, which are single term clusters. So, that that depends ofcourse on this particular updated distance matrix.

Once we have the updated distance matrix, we go back to step number 2 and then once again search for the minimum distance between the clusters n minus 1 now in numbers. And then we will have to come to step 3; because we will have to merge those two cases and then the number of cases will get further reduced by 1 and we will have n minus 2 cases. And then we will also require updation of the distance matrix by deleting rows and columns, corresponding to those two merged cases and then adding row and a column corresponding to the distance between the newly formed clusters and the previous clusters that were remaining in the data.

(Refer Slide Time: 46:12)



Step IV : Repeat steps II and III a total (n-1) times

Record the identity of clusters that are merged and the levels (distances) at which they are merged.

e.g. (4, 10) → $\boxed{4} = d_{4,10}$ ✓

(2, 15) → $\boxed{6} = d_{2,15}$ ✓

⋮

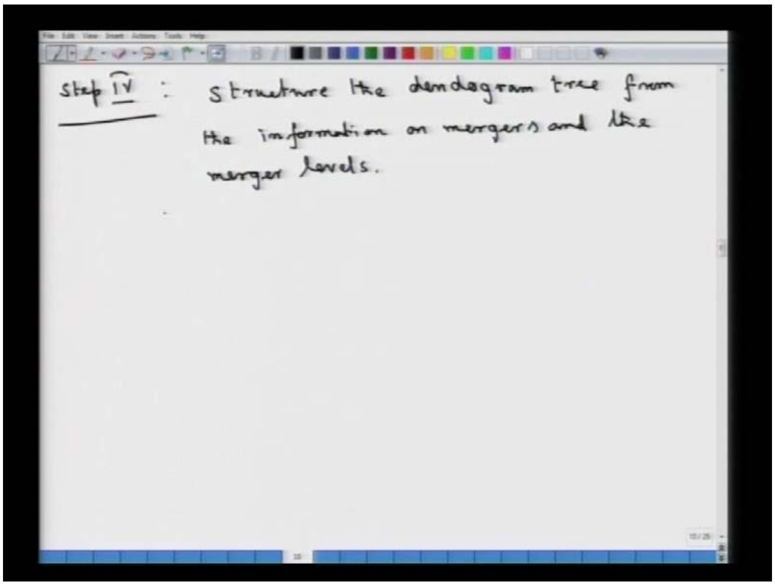
(4,10), (2,15) → $\boxed{10} = d_{(4,10),(2,15)}$

So, we just say that in step 4 in step 4 what we do is to repeat steps 2 and 3 one after the other a total n minus 1 times; because at the end of n minus 1 times, you will have n clusters in the data; I am sorry 1 cluster in the data n minus 1 times. We will have to record the identity record the identity of clusters that are merged at different levels that are merged and the levels at which they are merged. Now, these levels are nothing but the distances the minimum distances that we are going to find out distances or similarities. We are not talking about similarities at the moment. We are looking at distance matrix only, levels at which they are merged.

So, at the end of these n minus 1 steps, we will have say information of the type that say case number 4 is merged with case number 10 at a distance level of say 4. So, this is just a numerical example. I will go through a numerical example to see how this step actually works. So, suppose I have case number 4 and case number 10 is the first to be merged at a distance level 4, now this would be the distance between case number 4 and case number 10 surely. Now, say at the second step, we have case number 2 to be merged with case number 15 at a distance level of 6. This would then correspond to this case number 2 distances between case number 2 and 15.

So, these are the identities that we are going to keep at a particular point of time. Say at some other level, we will find that this cluster 4 10 gets merged with cluster 2 15 at a level distance level say 10, which is going to be the distance between these two cluster, which would come from the distance matrix of the previous step. So, this may come till we actually reach the point that we only have one single cluster. Now, these information of these informations are to be recorded; because these two clusters the clusters that are merged and the levels these are the levels, at which they are merged. So, these are the levels and these are the clusters which are merged at those points of time and once we have all these information, the last step is to use this information to construct the dendogram tree.

(Refer Slide Time: 49:34)



step IV : structure the dendogram tree from the information on mergers and the merger levels.

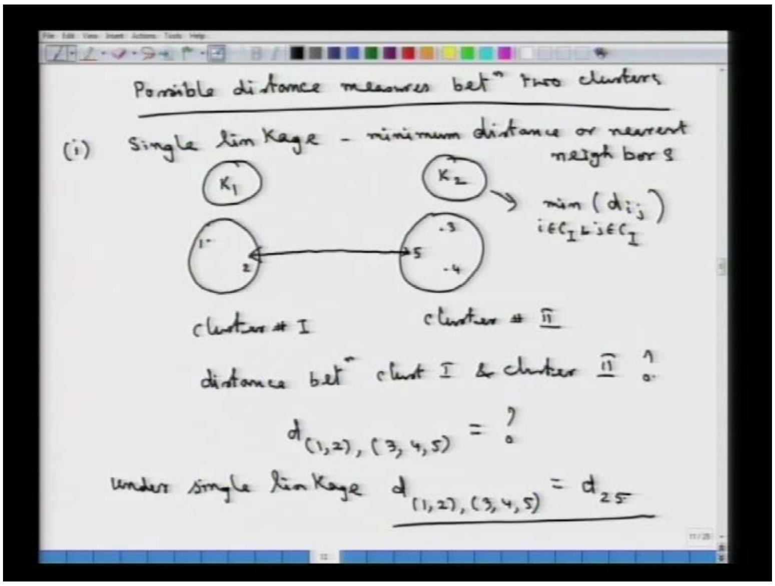
So, we will conclude with this step 5. Step 5 is that we will have to structure the dendogram tree, which is going to give us the hierarchical clustering representation of the data structure the dendogram tree from the recorded information from the information that we have collected at all these n minus 1 and n steps information on mergers, and the merger levels. So, these are the steps that are involved in constructing an agglomerative hierarchical clustering approach. Now, note that in this step 1 out here, this distance matrix needs to be computed.

So, suppose we are looking at an Euclidean distance, so this can be computed without any difficulty. At the second step, (Refer Slide Time: 38:45) we are just searching for the nearest pair of clusters from that n by n data matrix and that is basically looking at the minimum of that d i j values; that is trivial. So, one can find that. Now, then we will merge those two clusters and level them as U, V and then this updation of the distance matrix, which is the crucial point actually comes into existence. Now, this step here, that we are deleting rows and column its trivial ofcourse.

But when we are looking at this particular step here, that we are now going to add a row and column giving the distances between clusters which is newly formed and the remaining clusters. Now, how is this going to be done? Because we would require inter cluster distances. At the later stage of algorithm, when we are repeating step 2 and 3, we will have say at a particular point of time at the first step of updation, we will require the distance between a cluster which has two cases and a cluster clusters which has 1 object. At the next step or at subsequent steps, we might actually confront a particular situation that we have two clusters.

One containing four cases; other containing two, three cases. And then one needs to find out what is going to be the distance between these two clusters; because that distance is going to come into the distance matrix calculation and that distance matrix updation needs to be done, after merger at every level. Now, this particular distance matrix updation or adding the row and column basically involves some requires some thought of how the inter cluster distances are going to be measured, when we have clusters which are containing more than one objects. Now, there are various ways of doing that; I will just list the most important ones.

(Refer Slide Time: 52:32)



Possible distance measures bet^n two clusters

(i) Single linKage - minimum distance or nearest neighbors

$\min(d_{ij})$
$i \in C_I$ & $j \in C_I$

distance bet^n clust I & cluster II ?

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

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

So, let me list these things. I will explain them; possible distances distance measures possible distance measure between two clusters that is what is required, when we are going to do the distance matrix upgradation between two clusters. Now, the first type of distances that are proposed is what is called the single linkage. There are single linkage, complete linkage, then we have average linkage, centroid distances and other type of things. Let me try to tell you, what is the single linkage? This is what is going to be characterized by a minimum distance minimum distance now between clusters minimum distance or nearest neighbor approach.

Now, how is this going to be formed? Let us take this particular example. Suppose I have two clusters. These are two clusters. So, this is say cluster number 1 and this is cluster number 2. So, this is cluster number 1; this is cluster number 2. Suppose I have two cases here. This is case number 1; this is say case number 2; this is say case number 5, 3, 4. Suppose we are able to visualize in higher dimensions; ofcourse, we will not be able to visualize on such papers. Suppose it is on a two dimension and these are the location of the points 1, 2 which are now put in 1 cluster and 3, 4, 5 these are three points which are put in to another cluster.

Now in order to compute, so the point is to find out distance between this cluster 1 and cluster 2. So, what is the distance? Because, we will require this type of measure; the distance between 1, 2 cluster and distance between 3, 4, 5 cluster what is that equal to? If you are adopting single linkage, then the single linkage is basically going to look at the minimum distance that is between these clusters and that is what is going to actually be based on the nearest neighbors. Now, we see that this case number 2 and case number 5, these are the 2 points. The case number 2 from cluster number 1 and case number 5 from cluster number 2 are the nearest ones.

So, if we are adopting a single linkage, then this distance is going to be given by under single linkage approach under the single linkage approach, the distance between the cluster 1, 2 and 3, 4, 5 is going to be given by d 25. In general, we can have say k 1 objects here and k 2 objects here and then we will find out all the distances. From among these k 1 objects, we will have 1 object here join with all the objects here. Then the same is done for all the other objects that are present in k 1 with all other objects in k 2.

So, this is basically going to lead us to this type of distance. So, the distance between these cluster for a general cluster containing k 1 elements and a cluster containing k 2 elements in a multidimensional setup would be the minimum of d i j, where this i belongs to cluster number 1 and j belonging to cluster number 2. So, we will find out all possible distances like for this case here. There are how many such distances; distance from 1 to each of these 3 here; so d 13, d 15, d 14 and then from 2, d 23, 25, 24.

We will have all those 6 there and the minimum among all those is what is going to give us the single linkage distance between these two clusters. So, we will stop at this particular point. In the next lecture we will look at the other possible distance measure as I said the complete linkage, average linkage and other type of distances. And then we will actually apply these distances in order to get to single linkage hierarchical clustering, complete linkage hierarchical clustering and then look at some data and to form dendogram tree starting from the scratch, the data and then getting to the dendogram tree. Thank you