Applied Multivariate Analysis

Prof. Amit Mitra

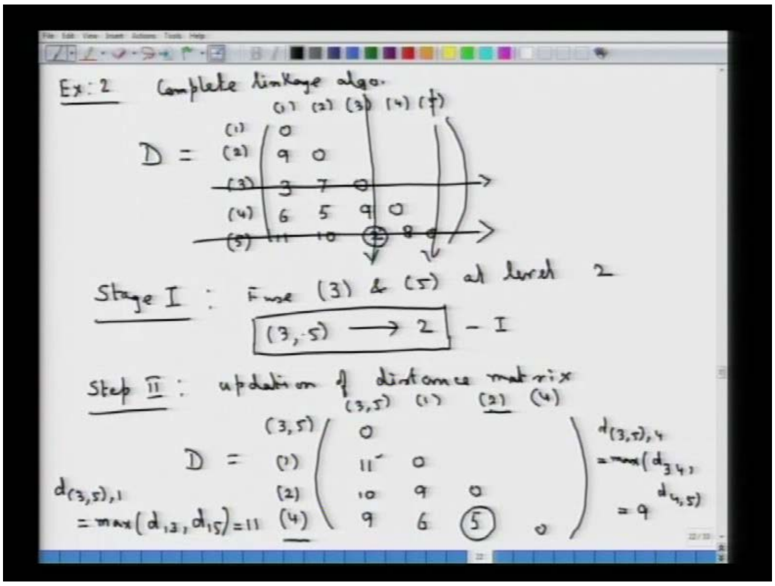
Prof. Sharmishtha Mitra

Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture No. # 28

Cluster Analysis

(Refer Slide Time: 00:27)



Ex: 2 Complete linkage algo.

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

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

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

Step II : updation of distance matrix

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

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

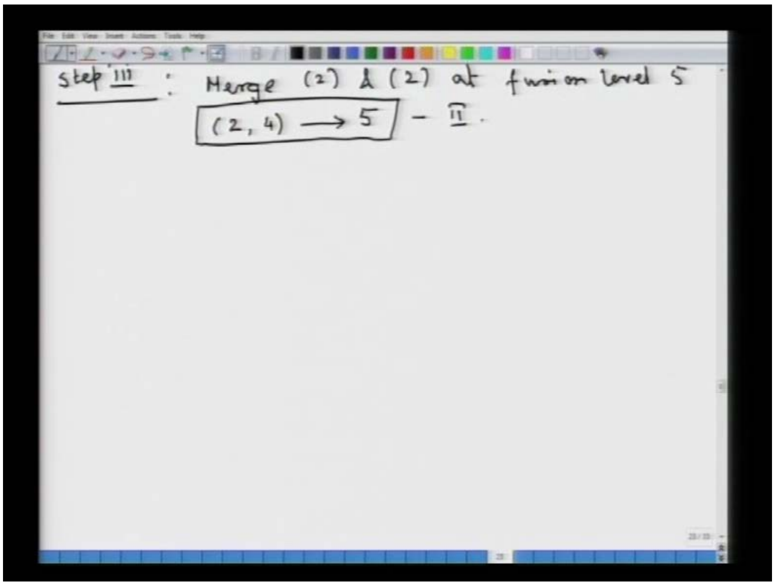
$d_{(3,5),4} = \max(d_{34}, d_{4,5}) = 9$

In the last lecture, what we will looking at we will looking at this complete linkage algorithm implementation for a data set, which a distance matrix given by the following. So, previously previous to this complete linkage clustering algorithm, we had looked at the single linkage clustering algorithm, and we had starting from a distance matrix. We had constructed this single, using the single linkage clustering algorithm; the dendrogram clustering tree and its interpretations were discussed.

And we have started this complete linkage algorithm implementation with a distance matrix, which is which was the same distance matrix as what we had use for the simple single linkage algorithm. When come up to this state number 2, wherein at the first step we had seen that, the cases 3 and 5 were merged at a diffusion level of 2. And then, we were require to do the updation of the distance matrix, which we had done in the last lecture, and had come up with this modified or rather the updated distance matrix.

Now, we start from this particular point, now if you look at this distance matrix now, we will have to look at which two clusters can now be merged. In order to do that, we will be looking at which of these this mutual distance that is given in the updated distance matrix is the minimum possible. So, as we see here that this 5 is the minimum among all this off diagonal entries, and hence we will be merging case number 2 with case number 4 and form a new cluster with these 2 cases.

(Refer Slide Time: 01:55)

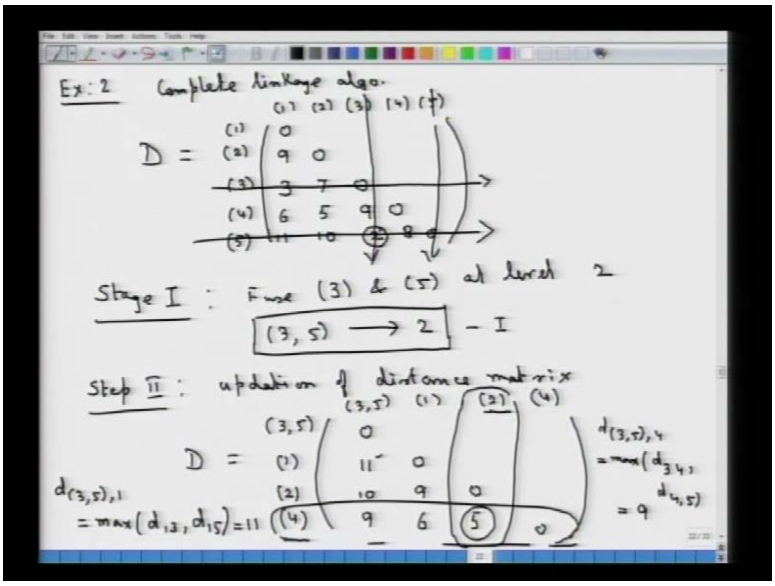


step III : Merge (2) & (2) at fusion level 5

$\boxed{(2, 4) \longrightarrow 5}$ – II .

So, we move on to that particular step here. Step 3 is to merge 2 and 4 at this fusion level 5, if you remember correctly that is at a distance level of 5. So, we will be keeping this particular information, similar to the previous information that we had kept that. We are forming this new cluster previously it was 3 5, now its 2 4. So, this 2 4 forms a new cluster at a fusion level of 5. So, that is information that we need to retain and carry forward, because we will be using this information. This is that second bit of information, which will be requiring in order to form the dendrogram tree.

(Refer Slide Time: 02:57)



Ex: 2 Complete linkage algo.

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

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

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

Step II : updation of distance matrix

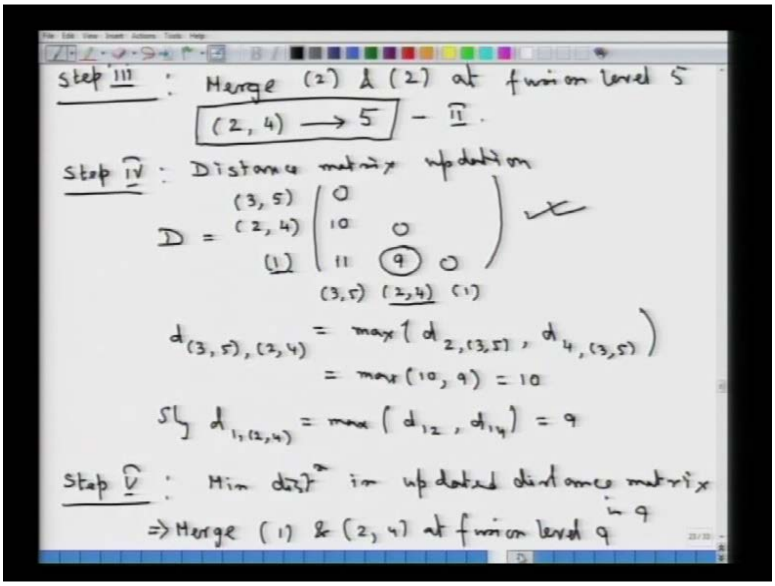
$$D = \begin{array}{c|cccc} & (3,5) & (1) & (2) & (4) \\ \hline (3,5) & 0 & & & \\ (1) & 11 & 0 & & \\ (2) & 10 & 9 & 0 & \\ (4) & 9 & 6 & \textcircled{5} & 0 \end{array}$$

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

$d_{(3,5),4} = \max(d_{34}, d_{45}) = 9$

Now, once we look at this particular this distance matrix now, neither 2 is a separate identity nor is 4 a separate identity and hence, the rows and the columns corresponding to this would vanish. So, we will not be having any entry corresponding to this 2, this 4. Similarly, corresponding to this 2 row here and the column 4 here will not be present there. What will be present there is the number of clusters. Now are 3 in a numbers. What are the clusters? The clusters now are 3 5 2 4 and 1.

(Refer Slide Time: 03:24)



Step III : Merge (2) & (2) at fusion level 5

$\boxed{(2, 4) \longrightarrow 5}$ - II.

Step IV : Distance matrix updation

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

$d_{(3,5),(2,4)} = \max(d_{2,(3,5)}, d_{4,(3,5)})$
$= \max(10, 9) = 10$

Sly $d_{1,(2,4)} = \max(d_{12}, d_{14}) = 9$

Step V : Min dist^n in updated distance matrix is 9

$\Rightarrow$ Merge (1) & (2, 4) at fusion level 9

So, we will require distance matrix updation. So, that is the next step of this implementation of the complete linkage algorithm. This is the distance matrix updation. Now, the new distance matrix is going to be 3 by 3 as we have discussed. So, that will be having this now has the clusters. Now, 3 5 was an existing clusters in the previous step. Now, we have formed a new cluster which is having cases 2 and 4 and there is one more case, which is 1 which is already in the existing list of clusters. So, we will have this 3 by 3 matrix filled up. Now, this is corresponding to that 3, 5 cluster.

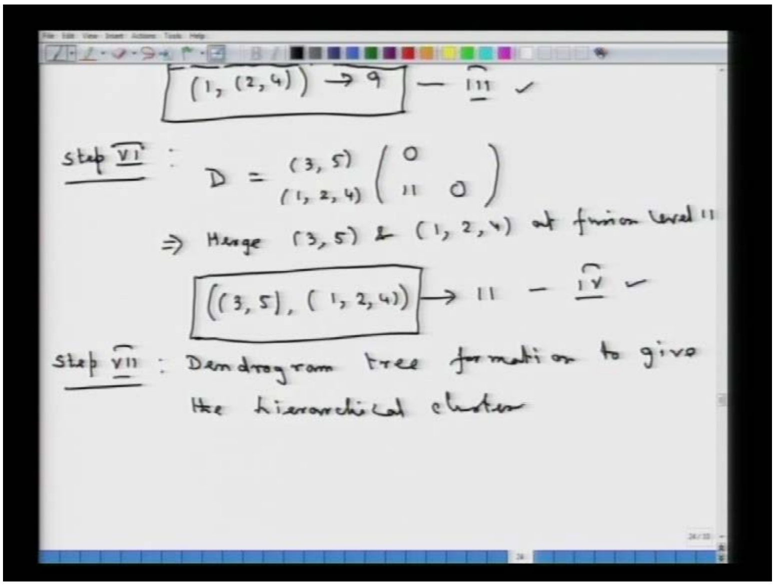
This column is representing from this cluster here; the second columns is from 2, 4 cluster and third column is from this 1 cluster. Now, note that there are some entries from the previous distance matrix here. Because the distance between 1 and 3 5, which is 11 will be an entry here; which will be carried forward from the previous table previous distance matrix itself. However, when we are trying to look at the distance between 1 and 2, 4 or 2, 4, and 3, which would come out here this is to be calculated. So, for the distance matrix updation, the distance between the cluster 3 5 and the cluster 2 4 needs to be computed; remember, we are looking at a complete linkage distance.

And hence, the distance between cluster 3 5, when cluster 2, 4 would be given through that maximum of that mutual distances; that is what would lead us to a complete linkage among this. So, the distance between 2 and cluster 3 5 and the distance between 4 and 3 5 would now be computed. What are these two distances? From the previous table, we will be getting this d 2 3 5 d 2 3 5 is 10. So, this is going to be maximum of that 10 entry and distance between 4 and 3 5 similarly, distance between 4 and 3 5 is 9. So, what we will be having is maximum of these two that is 10 to represent. Now, the distance between these two clusters which is 3 5 and 2 4.

Now, similarly one can, one have to obtain actually the distance between 1 and cluster 2 4 1 and 2 4. So, for that we will be requiring the maximum, because we are on complete linkage platform. We will be requiring the distance between 1 and 2 and the distance between 1 and 4, which can be looked at from the previous table itself and which is equal to 9. One can verify that that it is equal to 9 actually. So, we come up with this as the modified or the updated distance matrix at step 4, when we have three clusters in place.

Now, the next step, once we have an updated distance matrix that is, step number 5 is to look at this updated distance matrix and then, find the minimum distance minimum distance in updated matrix updated distance matrix updated distance matrix is worth, we have as we can see is 9. So, this is 9. This would imply that, we will merge the cases accordingly. So, now this 9 is the distance between a single turn cluster 1 and a cluster which is having two cases 2 and 4. And hence, we will be merging 1 and 2 4 at a fusion level of 9. 1 and 2 4 at fusion level 9.

(Refer Slide Time: 07:23)



$\boxed{(1, (2, 4)) \rightarrow 9}$ — $\overset{\frown}{\underline{\text{III}}}$ ✓

step $\overline{\text{VI}}$:

$$D = \begin{array}{c} (3,5) \\ (1,2,4) \end{array} \begin{pmatrix} 0 & \\ 11 & 0 \end{pmatrix}$$

$\Rightarrow$ Merge (3, 5) & (1, 2, 4) at fusion level 11

$\boxed{((3, 5), (1, 2, 4))} \rightarrow 11$ — $\overset{\frown}{\underline{\text{IV}}}$ ✓

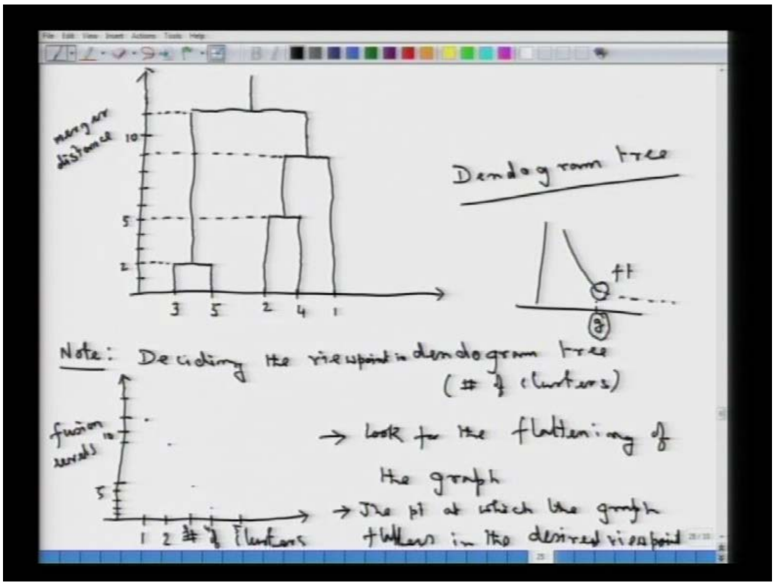
step $\overset{\frown}{\underline{\text{VII}}}$: Dendrogram tree formation to give the hierarchical cluster

And we would require actually this information to be preserved, because 1 with 2 4 is now merged 1 and 2 4 and Merged 1 and 2 4 are Merged at a fusion level of 9 and that is the information which we need to keep track of. This is the third information, when we are adding the information one after the other. After this 2, we will be having this third information which would be information, which would be required actually to form the dendogram tree. Now, we come to the next step that is, step number 6. So, at Step 6 we will look at updating this distance matrix, previous this distance matrix was 3 by 3, because we have three clusters. Now, we have got two clusters now.

One cluster is an existing cluster 3 5 and the other cluster is the cluster which is now formed, which is having 3 cases 1 2 and 4. And hence, we will be having the distance matrix which is just a 2 by 2 matrix. 3 5 is an existing cluster and 1 2 4 is a cluster that is newly formed. So, this 2 by 2 matrix will have 0's in the diagonal. This is the distance between this cluster and this cluster. Using a complete linkage (( )) distance philosophy, one can similarly find out, what is the distance between these two clusters. It turns out that, this particular distance is equal to 11. So, this would imply that, the last fusion is merging the cluster 3 5 and the cluster 1 2 4, at fusion level 11.

So, the last information that is, what we will be having is the following that, this 3 5 cluster merges with a cluster 1 2 4 to form a single 5 unit cluster at a fusion level or a merger level of 11. So, this is the 4th information that will be keeping. So, we have all the 4 information. This is the first fusion level; this is the second fusion level; this is the third fusion level and this is the forth fusion level. Then the last step of this algorithm is to drawn the dendogram diagram. Step 7 is the dendogram diagram or the dendogram tree formation to give the hierarchical clusters. to give this hierarchical clusters

(Refer Slide Time: 10:32)


Dendogram tree

Note: Deciding the viewpoint in dendogram tree (# of clusters)

→ look for the flattening of the graph

→ The pt at which the graph flattens in the desired viewpoint

Let us do that on the next slide here. So, what will be having similar to the single linkage distance here, the maximum merger level is 11. This is the merger distance or the threshold distance that is on the y axis and we will have this 1 2 3 4 5; say, this is fusion level 5; 6 7 8 9 10 11. So, this is the 10 level here now. What is the information that will be putting here? First we will put in the information that 3 and 5 have fussed at level 2. 3 and 5. So, there are cases which is 3 here, this is 5 here; they have merged at fusion level of just equal to 2. So, they are the nearest among all the pairs of objects. So, that is the first information.

So, this we have taken care of this one, will next look at the second information, then 2 and 4 have merged at a level 5. Let us see so, its 2 and 4 had been merged at a level 5.

So, this is that level 2 merger; this is that level 5 merger; case number 2 and case number 4 are merged to form a new cluster. So, we have taken care of this one. Now, what we are next going to have is input number 3. 1 2 and 4 merged at level number 9. So, the case 1 comes in here. So, there is a singles seek here, which is coming up to this level of 9; that is a merger level 9. And then, this two cases 2 and 4 and 1 now come together to form a new cluster, that is what this input had given. So, we have taken care of this 2. So, the last thing that we need to do is that, all the cases now merged at level number 5 4.

So, we have this one cluster, which is having these three cases 1 2 and 4; we were another cluster 3 and 5. And then, they come together, these two clusters come together at fusion level distance of 11 and that is the dendogram tree. So, we will have this as the dendogram tree, wherein the clusters are in hierarchical form. They with this we have formed from that distance matrix D by implementation of a complete linkage algorithm. Now, an important point to be noted in this type of hierarchical cluster analysis is that, where to how to decide on the view point of looking at the clusters being formed.

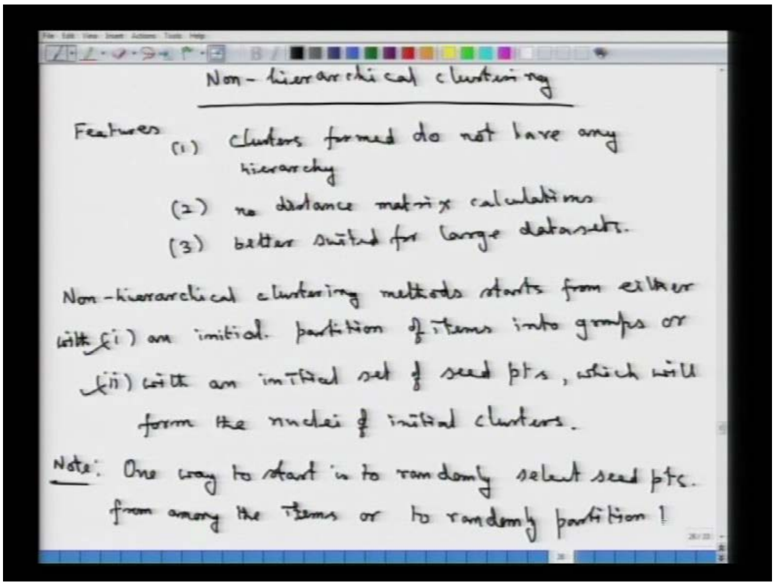
So, when we are looking at deciding the view point in dendogram tree now, if you look at this particular point, the deciding the view a view point actually. Let me correct it view point or the reference point actually; from which we are looking at. Whether to look at the dendogram because if we look at this level 11, we have all the five cases in one cluster, if we look at this 9 level here. We will have then two clusters. One having 2 5 3 5 and the other having 1 2 4 which distance is desirable. There are various ways actually at looking at this deciding upon the view point, is simple approach is to looking the following diagram which looks at the following on x axis, we have the number of clusters.

So, this this basically is to decide the number of clusters and on y axis here, we will have this as the fusion levels or the merger levels. Now, for this particular dendogram, if we look at suppose this is say, I divided here; at 5 here; this is 10; this is say 11. There is a number of clusters is 1 2 3 4 5. So, there are 5 cases. So, at the most we will have five clusters. Now, if you look at fusion level number 11 fusion level of 11 out here, then we will have one cluster here. Now, if you look at a fusion level of 9. So this is say fusion level 9 if you look at fusion level 9 we will have two clusters. So, we have the second point here 9.

If you look at a fusion level of 5, which is this one; we will have 3 clusters in the data. If we look at a fusion level which is 2 here… So, if this is 5, then let say this is 4 3 2 1 say. So, the fusion level of 2 we will have this as 4 clusters and then, this graph is up to this particular point. Because number of clusters 5, that would be at the 0 level of fusion. So, we have a graph like this and it is basically from such a graph look for the flattening actually of the graph. The point at which, the graph flattens the point at which the graph flattens is the desired view point.

So, we will say that, from this point onwards well this is very small data set and hence, the flattening of this particular graph is not observed here. If you have huge number of cases, then what it will turn out is that, we will have such a graph with many numbers of clusters. We will find that, if we have for example, a graph of this following nature that it is of this nature then, this is the point which we associate with the flattening point. So, this is what we associate as the flattening point. And there will be a number of cluster say g for this particular point here. This g usually taken as the view point for looking at the number of hierarchical clusters that, are formed from this particular data set.

(Refer Slide Time: 17:45)



Non-hierarchical clustering

Features (1) clusters formed do not have any hierarchy

(2) no distance matrix calculations

(3) better suited for large datasets.

Non-hierarchical clustering methods starts from either with (i) an initial partition of items into groups or (ii) with an initial set of seed pts, which will form the nuclei of initial clusters.

Note: One way to start is to randomly select seed pts. from among the items or to randomly partition !

Now, we move on to the other type of clustering algorithm, which is non-hierarchical clustering. non-hierarchical clustering. We will discuss one important in non-hierarchical clustering method, which is called the k means clustering or the method of iterative we location method. So, what are the features of such a non-hierarchical clustering method?

So, the features the salient features would be that, the clusters are naught clusters formed do not have any hierarchy. Do not have any clusters correct this 1 clusters formed do not have any hierarchy. Number 2, there is no distance matrix calculations as contrast to the previous hierarchical clustering method. No distance matrix calculation, we start with the no data and end up with row data only. No distance matrix calculations.

Now, it is said that, it is better suited for a higher or large data set better suited for a large data sets why is that so, because if you look at the output an hierarchical clustering. If you have a huge data set actually then, the number of cases may run into hundreds and thousands. And in a such situation, we will have also such merges a cases, this x axis will be so crowded with all those cases for a high dimension for a large data set. Then, it will be very difficult actually, to make out which clusters are formed at which particular level. Hence, the output will be so cumbersome for a hierarchical cluster analysis output.

It is better suited we have a non-hierarchical clustering and we do not talk about any hierarchy in the formation of the clusters. Now, how what is the type of algorithm or what is type of behavior of such non-hierarchical clustering method? Non-hierarchical clustering method starts either clustering methods starts from either number 1, an initial partition initial random partition actually initial partition of items or objects into groups or its start with an initial set off starts either with and initial partition of items into groups or with and initial set of seed points. I will talk about explain what I mean seed points. Now, these seed points will actually act as will form the nuclei of initial clusters.

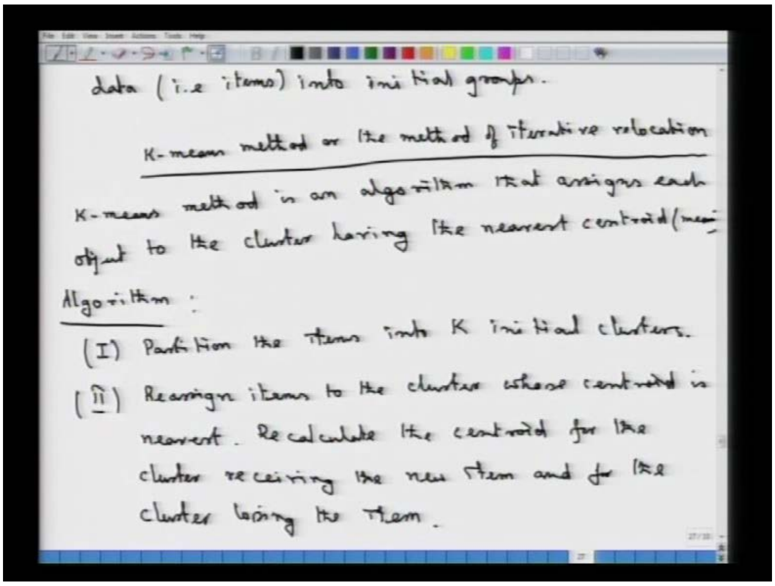
So, what are we trying to do here? We are trying to do in the non-hierarchical clustering is the following that, if we have n cases with us, we are basically trying to put this n cases into k number of means. k number of means in the sense that, k number of clusters. So, each of these k means are identification levels for each of this clusters. And then, naturally there is not going to be any hierarchy between these means; all of these means are different. So, we will have such k clusters in the data, as the final output of this particular system. They do not have any hierarchical structure in the formation of the clusters.

There will be no distance matrix calculation as we will see and this better suited we have already discussed. This non-hierarchical clustering method, how do they start? Well When we have a hierarchical clustering method, we start with what? We start with a distance matrix. When we have a non-hierarchical clustering method, it starts basically with either an initial partition of items into groups. So, if we see that, there are k clusters in the data, we will start with an initial random which was on k clusters; which are basically the partition of the data set either with that or we start with k seed points, k multidimensional points randomly chosen.

It may be chosen from among the existing data or may be just k dimensional p dimensional k such seed points, which now act as the centroid or the nuclei of the initial cluster. So, these are the two ways in which a non-hierarchical clustering method actually starts. I will just put it as a note that, when I have said that, I start with either this or with this second initial set of seed points. How one can actually do that? One way to start is to randomly select seed points from among the data itself, from among the items, because each of these items are the p dimensional p dimensional in nature.

When we talk about seed points, which are going to form as the nuclei of the initial clusters, which are going to change iteratively. Those points also need to be of the same dimension as that of the data. And thus choosing initials set of seed points say k in number would be to look at randomly choosing k points, from among the n possible items. Now, this is one way, which would take care of this particular approach or what we can do is to randomly partition the data to randomly partition the data.

(Refer Slide Time: 24:29)



data (i.e items) into initial groups.

K-means method or the method of iterative relocation

K-means method is an algorithm that assigns each object to the cluster having the nearest centroid (mean)

Algorithm :

(I) Partition the items into K initial clusters.

(II) Reassign items to the cluster whose centroid is nearest. Recalculate the centroid for the cluster receiving the new item and for the cluster losing the item.

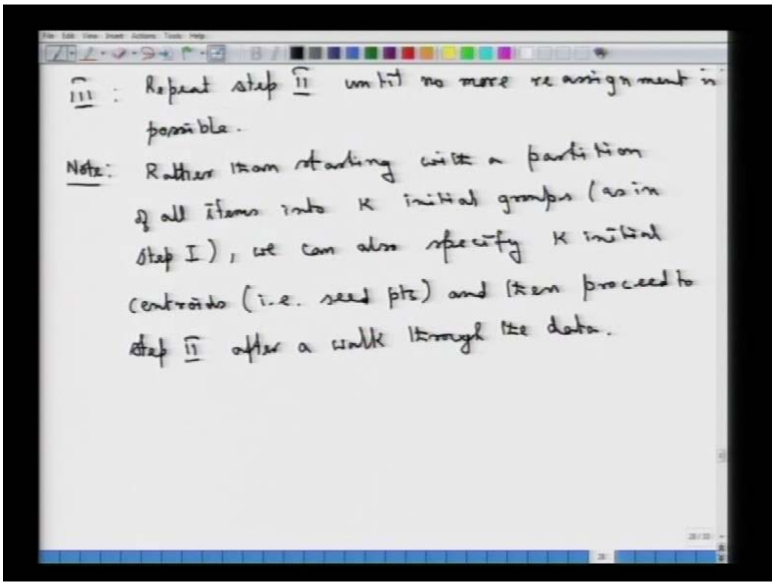
So, it is random partition of the data that is items or objects into initial groups. That is how one usually implements these two different types of approaches. We will look at one example. In the example, what we will do is we will approach the first; we will adopt the first approach that, we will have initial partition of items into groups. And then, we will see how this method of k means clustering is going to evolve? Now, I said that, the k means clustering, k means method or the method of iterative relocation or the method of iterative relocation is an important clustering approach, which actually leads us to non-hierarchical clusters that can be formed from the data.

Now, k means method, I will just explain what; how this particular method actually goes about? k means method is an algorithm k means a method is an algorithm that assigns each object that assign's each object to the cluster having the nearest nuclei or centroid. So, we will be having all such possible clusters. And then, this k means method actually will assign a particular object to a cluster, if that clusters centroid is nearest to that particular object. When that object is being compared with the centroids of all the possible clusters into which, the object can actually go to.

Now, what is the algorithm for this k means method? The algorithm goes in the following steps. So, the first step is to partition the data, partition the items into k initial clusters. This is one approach, we could have also put k centroid into that particular initial data and then, make the assignment accordingly. Now, first we partition all the possible n items in to k initial groups. And once that is being done then, we reassign will look at possible reassignment; reassign items to the clusters whose centroid is nearest in Euclidean sense. Then, once we have done that, we would recalculate the centroid for the clusters receiving the new item receiving the new item and for the cluster which is losing that item receiving the new item and for the cluster losing the item of the case.

So, what we are doing? We are looking at k initial clusters and then, we are looking at whether a particular object is closest to its own cluster. That is, in the initial cluster, whether it is nearest to that or which center to which centroid of the initial clusters k initial clusters that item is closest 2. And if that if we find that, a particular item is closer to another cluster than to the initial assigned cluster will make a reassignment. Once a reassignment is made, we will have to recalculate the centroid; the k centroid for two specific clusters. One cluster which is receiving the new item and the cluster which is losing that particular item.

(Refer Slide Time: 29:34)



III : Repeat step II until no more reassignment is possible.

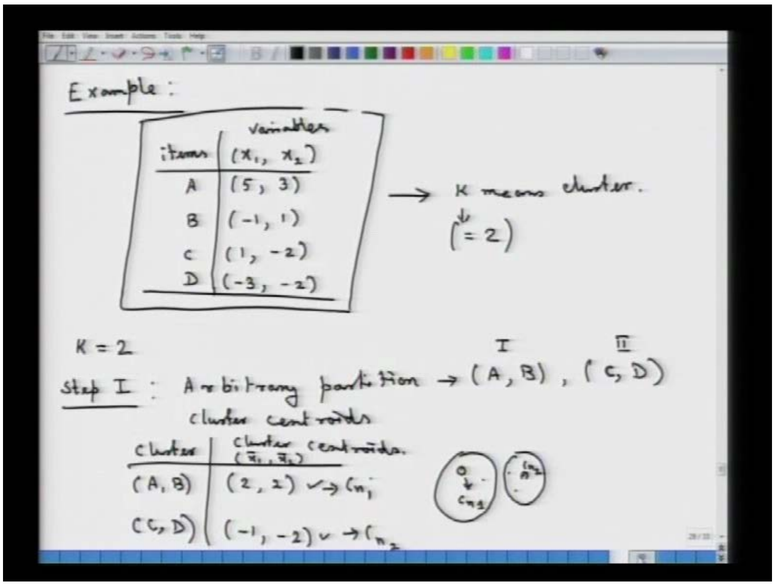
Note: Rather than starting with a partition of all items into K initial groups (as in step I), we can also specify K initial centroids (i.e. seed pts) and then proceed to step II after a walk through the data.

And then, we will continue this particular method of assignment reassignments, until we find that no reassignment is possible. So, we will look at repeat step 2, until no more reassignment is possible. So, it is basically in these three simple steps, we will look at a simple data and then, try to see how this k means algorithm method actually goes about? Now, I just put it as note, what I said that rather than starting the process of this k means algorithm with a partition; a random partition of the data; random partition of all items into k initial groups. That is what, we have written in the algorithm as in step 1 of the algorithm.

We can also assign seed points. We can specify k initial centroids straight away, which are going to act as this seed points and then, proceed to step 2 of the previous algorithm Step Step 2 after a walk through the data through the data right. Why do you do that? Because once we have specified k such initial seed points, these may be k randomly chosen multidimensional items only. Then, we need to actually look at we have to look at all the data and then, we will have to look at these initial centroids. How do these centroids behave as far as the data is concerned?

Because when if a particular cases nearest to particular chosen or rather randomly initialized seed point, we will have that point to be associated with that centroid. We will have a cluster around that particular seed point, wherein the case which is closest to that particular centroid or the randomly chosen seed points is belonging to. And then, we walk through the data, go to step 2. And then, look at possible reassignments in the data that can be possible and then, the algorithm goes through.

(Refer Slide Time: 32:41)



Example :

items	variables (x_1, x_2)
A	(5, 3)
B	(−1, 1)
C	(1, −2)
D	(−3, −2)

→ K means cluster.
$\downarrow$
(= 2)

K = 2

Step I : Arbitrary partition → $\overset{I}{(A, B)}$, $\overset{II}{(C, D)}$
cluster centroids

cluster	cluster centroids $(\bar{x}_1, \bar{x}_2)$
(A, B)	(2, 2) ✓ → C_{n_1}
(C, D)	(−1, −2) ✓ → C_{n_2}

Now, let us look at a numerical example to illustrate, how this k means clustering algorithm actually behaves. So, this is an example we have the following that we have items to make lives simple, we just take four cases A, B, C, D. So, these are four items of four cases in the data. Let us assume that these are each of these cases are having two dimensions, so these are all two dimensional data. So, that the first case is characterized by this vector 5, 3. The second is characterized by this vector minus 1 plus 1, and then the third case is characterized by 1 minus 2. The third case fourth case or case D is characterized by minus 3 minus 2. So, this is the data what we have to start with, and we will look at starting with this data how to get to a k means cluster.

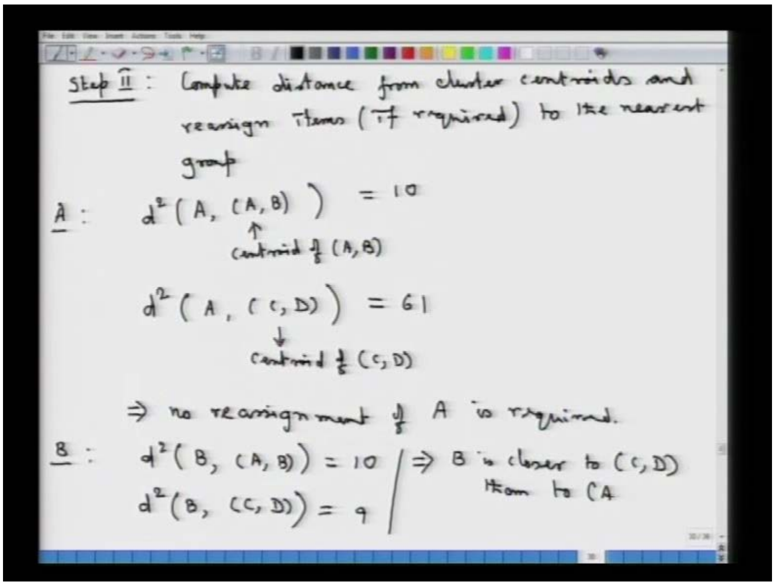
Now, let that k be equal to 2 for this given illustration. So, we are trying to divide these four cases into two clusters using a k means clustering approach, which is going to lead us to clusters which are non-hierarchical in nature. Since we have this k equal to 2 at the first step of implementation of this k means clustering algorithm, let us look at an arbitrary partition of the data. So, we need to have two elements in the partition arbitrary partition say, we take one partition to be A, B and the other partition to be C, D. So, we randomly put A and B one cluster, C and D the second cluster. So, this corresponds to the first cluster; this corresponds to this second cluster.

Now, we will have to first look at, if these are two randomly formed clusters then, what are the centroids of these? And then look at possible reassignment 2; some other cluster different from the initial random cluster. So, we will have to look at these cluster centroids in this particular data. So, we will have this as two possible clusters and these are cluster centroids. This is the cluster A, B; this is the cluster C, D. Now, if you look at the cluster centroid it is nothing but, x 1 bar actually you can say and this is x 2 bar. So, where x 1 bar is the mean of the first component of the two elements, which are belonging to the first cluster that is A and B.

So, this cluster centroid, which is the cluster A B would be having the coordinates as the mean of this 2, as the first coordinate and the mean of this 2, as the second coordinate. And hence, both of them are to in this particular case. Similarly, for the cluster which is having elements C D, we look at what are these quantities? It turns that, this is minus 1 and this is minus 2. So, this is these are the centroids. Now, we have these two clusters. This is the centroid; suppose I say that this is centroid number 1; this is centroid number 2. So, this is centroid number 1 and this is another random cluster, which is having this centroid number 2 as that.

Now, there are two cases; A and B here and C and D here. So, we will look at whether A case is closer to this centroid center or the other cluster centroid center. These two clusters may not be so different. So, we might be having the clusters C n 2 as this point. So, there are cases C and D setting here. So, we will find out the distance between A and this cluster center A and this cluster center. And try to see whether A is closer to this cluster center than to the other cluster center and then, look at possible reassignment.

(Refer Slide Time: 37:18)



Step II : Compute distance from cluster centroids and reassign items (if required) to the nearest group

A : $d^2(A, (A, B)) = 10$

$\uparrow$ centroid of (A, B)

$d^2(A, (C, D)) = 61$

$\downarrow$ centroid of (C, D)

$\Rightarrow$ no reassignment of A is required.

B : $d^2(B, (A, B)) = 10$

$d^2(B, (C, D)) = 9$

$\Rightarrow$ B is closer to (C, D) than to (A

So, that goes into the second step of this algorithm. So, in this step 2, we will look at that assignment. So, the second step we compute the distances as I said that, compute distances from cluster centers from cluster centroids and reassigned items if possible or if required to the nearest group now. That possibly is going to happen, because we had thus the initial set of clusters to the nearest group now. For the given data here, we are looking at this to be one centroid and this to be the other centroid. So, what we are going to do is to look at the distance of A from its clusters center and distance of A from the other cluster center.

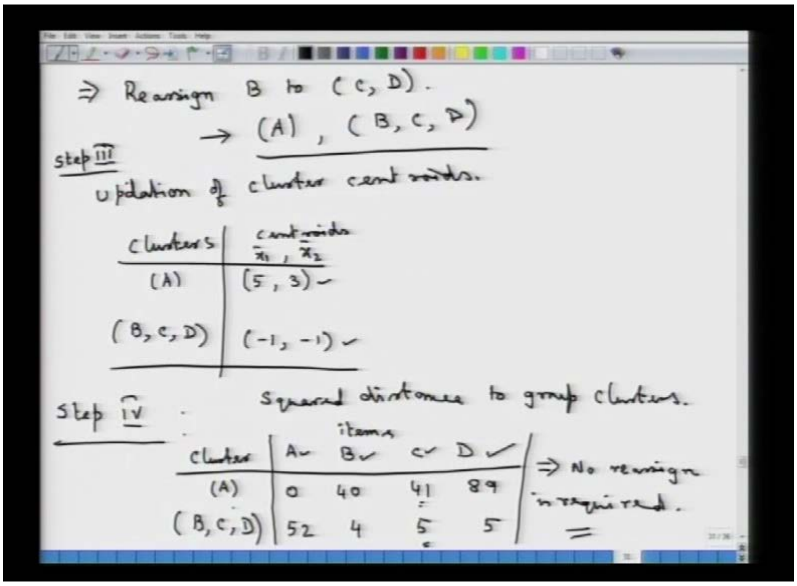
Now, if we find that, the distance of A from its own cluster center, initial randomly chosen cluster center is higher than the distance of A from the other cluster center. We will reassign A to this cluster. Once we do that, we will stop at that particular point and then, recalculate the centroids of the clusters. Because this cluster is now losing A and the other cluster is going to gain A. If that is possible, if that is not the case if you find that A is closer to its own cluster, own initial cluster than to the other cluster which is containing the points C and D. Then, we will not disturb A, keep it in the same cluster as what it was present in the initial randomly allocated clustering.

Then, we will look at the second case and look at whether any reassignment is required for that point or not. Then we will continue, at any point if you find that reassignment is done, we will stop at that particular point. And then, what we are going to do is to recalculate the centroids of the clusters. So, let us look at what we are going to get here. Now, if you are looking at the distance square of A from the cluster center; now by saying this I am saying that, this is centroid centroid of A B. We will have to look at this. We will have to this is for case number A. So, we will look at the distance of A from its own cluster; distance of A from the cluster center or the centroid of the cluster, which is C D cluster.

Now, it is easy to see what these quantities are? It turns out that, this is equal to 10. This is the (( )) distance square and this distances 61. So, A is closer to its any initial cluster than to the other cluster. So, this would imply no reassignment of A is required. Now, once that is turn, we move on to case number B and we will calculate the similar thing. We will have to look at the distance square of B from its own centroid. We will also have to look at this distance square of B from the other cluster centroid, C and D. As it turns out that, this is equal to 10 and this is equal to 9.

So, B is the distance of B from its own cluster center is higher than the distance of B from the other cluster which is C and D. So, this would imply that, B the point which was initially sitting in the cluster A, B is closer to the other cluster and its own cluster. and see And hence, reassignment of B is required. So, we will need to do this, because we see that B is closer to the cluster C, D than to the cluster A, B.

(Refer Slide Time: 41:53)



⇒ Reassign B to (C, D).

→ (A) , (B, C, D)

step III

updation of cluster centroids.

clusters	centroids $\bar{x}_1$, $\bar{x}_2$
(A)	(5 , 3) ✓
(B, C, D)	(−1, −1) ✓

Step IV : Squared distances to group clusters.

	items			
Cluster	A ✓	B ✓	C ✓	D ✓
(A)	0	40	41	89
(B, C, D)	52	4	5	5

⇒ No reassign is required.

This would further imply that, reassign B to C D. Now, the setup gets changed as B is reassigned to C D. So, the new clusters are, there is one cluster with single turn case A and there is another cluster with these three cases B, C and D. Since we have this as the two new clusters, we will require updation of what centroids updation of cluster centroids, because the previous centroids were that for the randomly allocated terms there. So, this is basically going to be the Step 3, because we had at Step 2 that… Reassigned I am sorry this is going to be Step 3 this is going to be Step 3 of this implementation; updation of the clusters centroids.

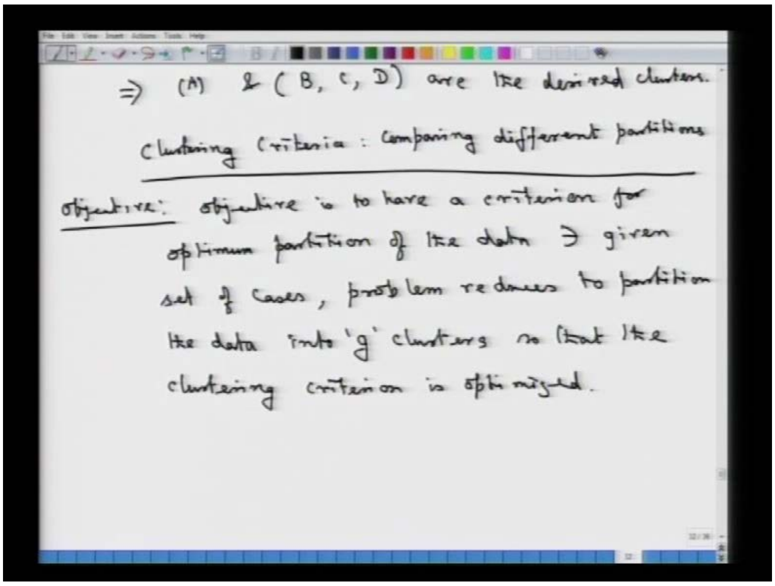
Now, what are the clusters? Now, in the data as we had said that, these are the clusters. This is cluster number A and the second cluster is having these three cases and these are centroids. It similarly, that x 1 bar corresponding to the cases under this x 1 bars corresponding to the cases under the second cluster. And this is going to be that point itself, because it is a single turn cluster. This point is the cluster centroid, which is the ordinates of A and for this B C D cluster one can calculate that, this is now the new cluster means centroid points of this cluster. Now, once we have that, we will have to look at possible reassignments.

Possible reassignments meaning there by, will have to see whether the distance of A from itself is always 0. So, A cannot be reassigned here. We will look at, what is the distance of B from this is center? And what is the distance of B from this center? We will look at the distance of C from this point here and this point here and look at possible reassignment. Now, we will have this particular table. The squared distance to group clusters. We can have the following table that; we are looking at items which are A B C and D. We will have to look at the distance of each of these items from the two cluster centers one this and one this.

So, we have got a cluster as A and another cluster as B, C, D. And we look at what is the distance of A from A that would be 0. The distance of A from this particular point here can be obtained, which is 52. And similarly, this table can be completed 40, 41, 89 and this is 4, 5 and 5. So, this for example, denotes the distance of the items C from A clusters centroid and this denotes distance of C from this B, C, D cluster centroid. Now, we see that, A anyway cannot be reassigned. We first look at B, the distance of B and the cluster center A is 40; this is 4 and hence, B is correctly put into this set.

So, no reassignment for B is required neither for A. Then, we look at the item number C. C is closer to the cluster centroid, which is having the points B, C, D than what it is to A, because this is smaller than this. And hence, no reassignment is required for the item C and also for D; we find that it is closer to this cluster centroid than to this cluster centroid. Hence, no reassignment also for D is required. So, this is this step is going to tell us that, no reassignment further is required. If that is the case, then we terminate this particular procedure. Then finally, say that these two are the two clusters in the data, which are thrown up by the k means clustering.

(Refer Slide Time: 46:34)



⇒ (A) & (B, C, D) are the desired clusters.

Clustering Criteria : Comparing different partitions

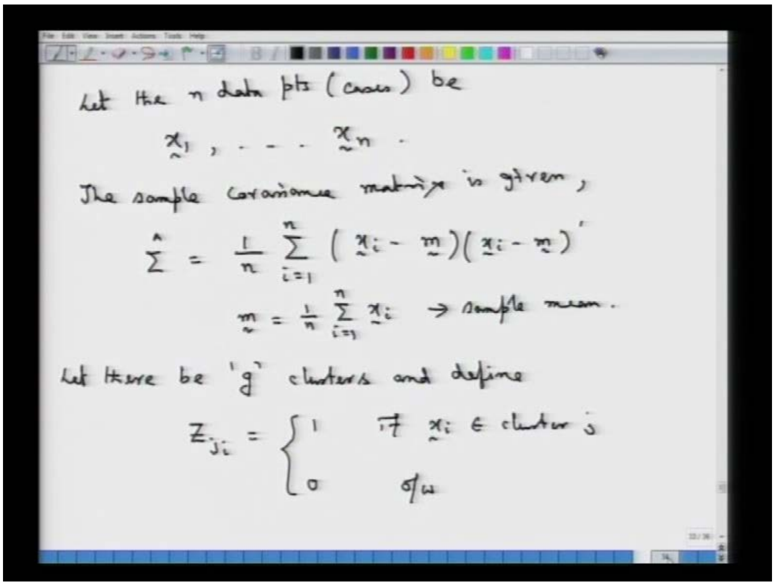
objective: objective is to have a criterion for optimum partition of the data ∋ given set of cases, problem reduces to partition the data into 'g' clusters so that the clustering criterion is optimized.

So, this would imply that, this 1 and it is I am sorry it is not 1. It is A and B, C, D. A and this B, C, D are the desired clusters. So, we have just two clusters in this particular data. Now, the next thing that is, what we are going to look at is some sort of optimality criterion in deciding or some sort of approach of comparing different partition or the different clusters in the data. So, that is the last thing which we are going to see. After that, we will look at some real life data and look at how the clustering for such real life data actually behaves? A will be actually looking at many such criterions criterion.

So, we look at cluster criterion that is, basically for comparing different partition or different clustering levels, comparing different partitions of the data. Now, what is the basic objective of looking at this type of criterion? The objective is to have the objective is to have a criterion for optimum partition of the data for optimum partition of the data such that given set of cases which are going to be clustered. The problem reduces to partition the data into g clusters. g is a number which has to be derived clusters. So, that the clustering criterion is optimized.

So, that the clustering criterion is optimized, because as we have seen say, in the hierarchical clustering approach or in the non-hierarchical clustering approach, we can have different cluster, different partitions of the data. If we are looking at a hierarchical clustering algorithm, if we look at two different fusion levels or the threshold distances then, we have completely different clusters in that particular level. And hence, there has to be some way of comparing such clusters, whether we should look at particular say two clusters in the data, whether to look at three clusters in the data. What is going to give us some sort of optimality with respect to some criterion? That we are going to propose shortly.

(Refer Slide Time: 50:07)



Let the n data pts (cases) be

$\underset{\sim}{x}_1 , \ldots \underset{\sim}{x}_n .$

The sample covariance matrix is given,

$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (\underset{\sim}{x}_i - \underset{\sim}{m})(\underset{\sim}{x}_i - \underset{\sim}{m})'$$

$$\underset{\sim}{m} = \frac{1}{n} \sum_{i=1}^{n} \underset{\sim}{x}_i \rightarrow \text{sample mean.}$$

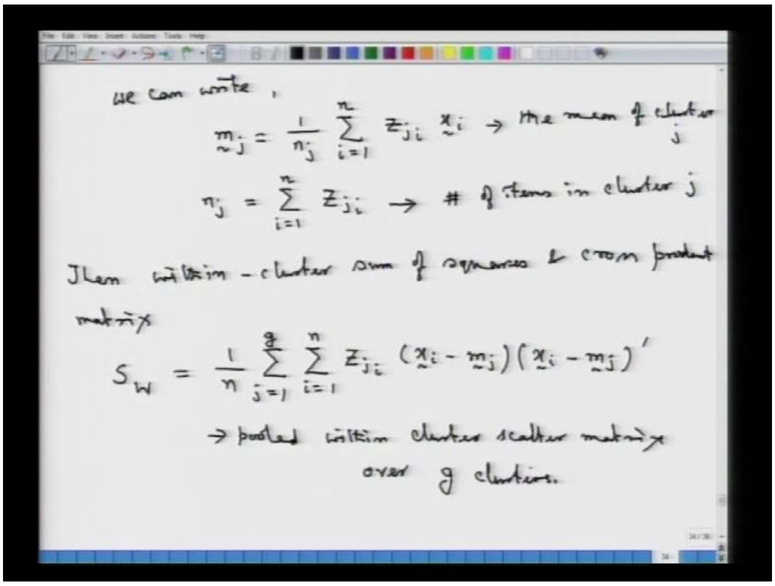
Let there be 'g' clusters and define

$$Z_{ji} = \begin{cases} 1 & \text{if } \underset{\sim}{x}_i \in \text{cluster } j \\ 0 & \text{o/w} \end{cases}$$

So, that is what, is a basic objective of this particular analysis. Now, let there be n data points. Let the n data points, now these are cases be given by say x 1 x 2 and x n. Now, given this particular data, the sample variance covariance matrix the sample variance covariance matrix is given by this we have seen time and again say sigma hat which is say with a divisor n. So, that it is corresponding to the maximum likely hood estimator. So, that is i equal to 1 to up to n x i minus say m, where m is the sample mean x i minus m transpose, where this m vector is 1 upon n summation i equal to 1 to n x i. So, this basically is a sample mean.

Now, let us define the number of clusters. Let there be g clusters, we are trying to compare or rather have a platform for comparing such possible clusters. Let there be g clusters and define this following indicator function. z j i to be equal to 1 and 0 1 if the case x i belongs to cluster j and is equal to 0 if it is otherwise. So, if z j i is given by this, we can write the following quantities. We can write that, our m vector as following.

(Refer Slide Time: 52:01)



we can write,

$$\underset{\sim}{m}_j = \frac{1}{n_j} \sum_{i=1}^{n} z_{ji}\, \underset{\sim}{x}_i \rightarrow \text{the mean of cluster } j$$

$$n_j = \sum_{i=1}^{n} z_{ji} \rightarrow \# \text{ of items in cluster } j$$

Then within-cluster sum of squares & cross product matrix

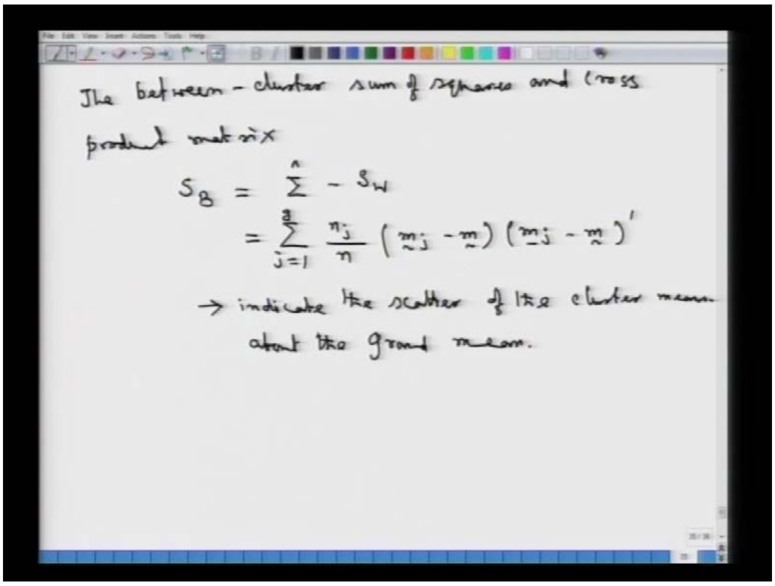
$$S_W = \frac{1}{n} \sum_{j=1}^{g} \sum_{i=1}^{n} z_{ji} (\underset{\sim}{x}_i - \underset{\sim}{m}_j)(\underset{\sim}{x}_i - \underset{\sim}{m}_j)'$$

→ pooled within cluster scatter matrix over g clusters.

So, this m vector say I have that say m j vector; this is going to the cluster mean for that j th cluster, which is going to be 1 upon n j. n j is the number of points in that particular cluster and this is summation i equal to 1 to up to n, the entire data z j i that in to x i; wherein we have n j to be equal to summation z j i, this indicate variables for i equal to 1 to up to n. So, this is the mean of cluster j and what is this? This is the number of items in cluster j. This is simple to see that, because if you look at z j i; z j i is equal to 1 if x i belongs to the jth cluster. And if we have n j items among this small n to belong to cluster number j, exactly n j of them among this z j i is for a particular j would be equal to 1. And hence, the sum would be equal to n j only cluster or not.

So, these two are the two simple quantities. Then, we can define the two following quantities. The within-cluster, sum of squares and cross product matrix some of squares and cross product matrix is going to be given by say s w, which is equal to 1 upon n. We will still use that indicator function, this j is equal to 1 to up to g. So, these are the number of clusters and i is equal to 1 to up to n. Then, we have this as z j i the indicator. This is x i minus m j; this is the cluster center for the j th cluster x i minus m j transpose. So, this is also called the pooled within cluster pooled within cluster scatter matrix scatter matrix over the g clusters. Now, if this is the within-cluster, sum of squares. One can define also the between-cluster.

(Refer Slide Time: 55:04)



The between - cluster sum of squares and cross product matrix

$$S_B = \hat{\Sigma} - S_W$$

$$= \sum_{j=1}^{g} \frac{n_j}{n} (\underset{\sim}{m}_j - \underset{\sim}{m}) (\underset{\sim}{m}_j - \underset{\sim}{m})'$$

→ indicate the scatter of the cluster mean about the grand mean.

So, the between-cluster, sum of squares and cross product matrix some of squares and the cross product matrix is say S B, which is equal to sigma hat minus this S w which is going to be given by simple subtraction. This is going to be given by this n j divided by n and then, we will have this as m j minus m into m j minus this m, where m is the grand mean. So, what is this going to indicate? This is going to indicate the scatter of the cluster means cluster means about the grand mean. This is of the cluster means, because we are looking at the deviation of this m j from m, the grand mean grand mean corresponding to all the clusters and we are looking at how that is being deviated?

So, this is what is termed as the between-clusters sum of squares and cross product term and the previous is what is called the within-cluster sum of squares and cross product matrix. Now, the optimality criterion for clustering are basically based on these two measures S W, S B and sigma hat, which we see will see in the next lecture. Thank you