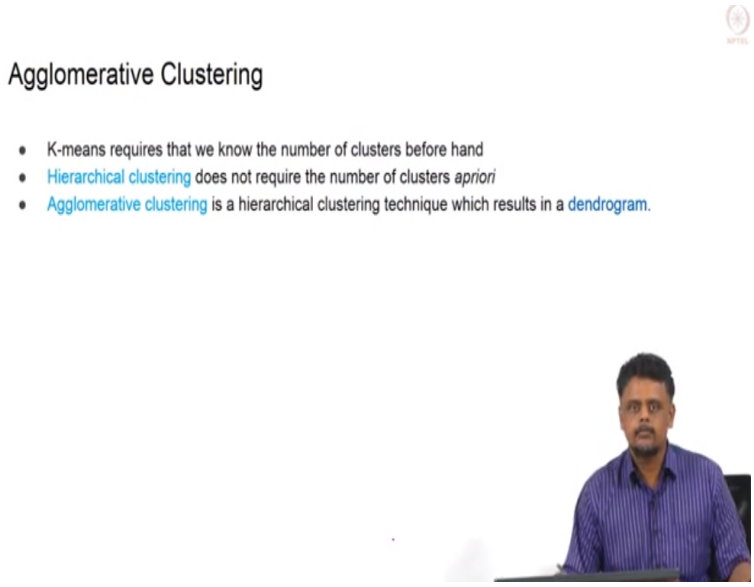
Machine Learning for Engineering and Science Application
Professor Ganapathy Krishnamurthi
Department of Engineering Design
Indian Institute of Technology Madras
Agglomerative Clustering

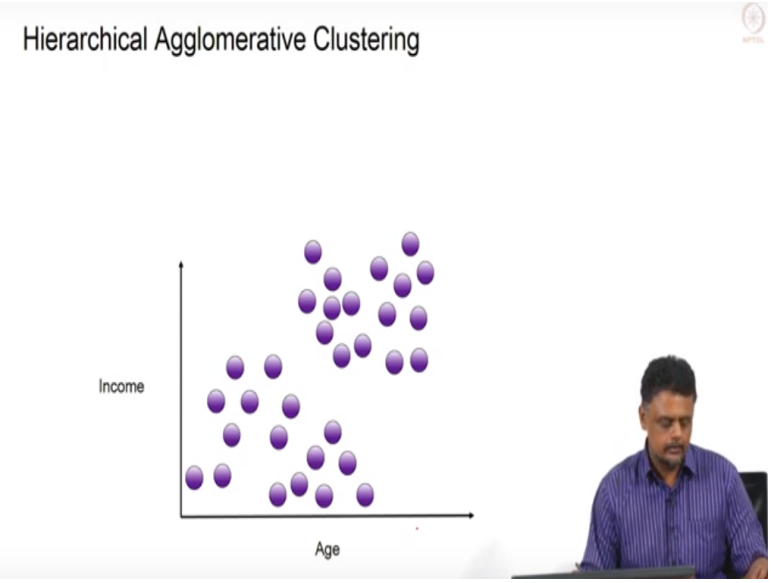
Hello and welcome back in this video we will continue with unsupervised clustering techniques we will look at agglomerative clustering only illustrations or figures in this presentation are courtesy of Intel software based on their educational software and also the material is also inspired by elements of statistical learning tech support by teacher and his colleagues.

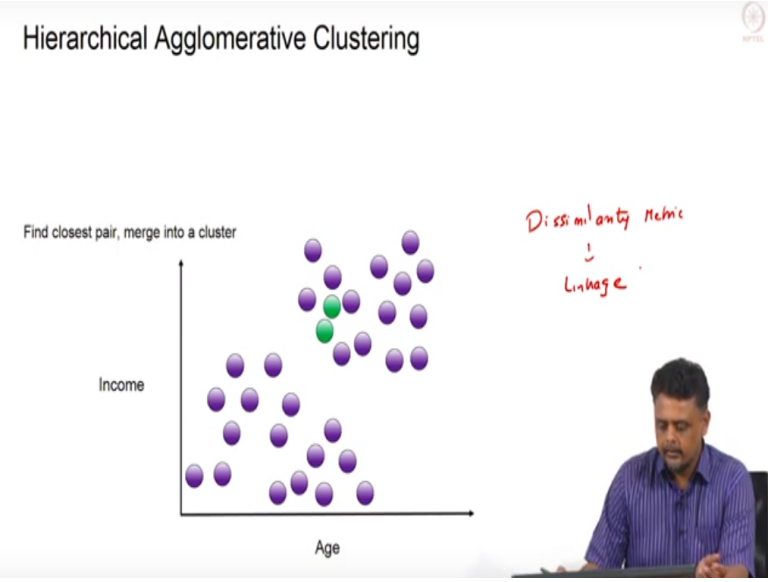
(Refer Slide Time: 00:38)



So the previous video we have looked at the K-means algorithm and we saw that in order to perform and supervised clustering we need to know the number of would be you need to know the number of clusters for hand, so that was input and of course there is heuristic to determine the number of clusters based on the inertia or the cost function the agglomerative clustering that we are going to look at it is an hierarchical clustering technique there is not really required the number of clusters APRIORI, but it result in what is called in DENDROGRAM, it is like a binary tress structure where the user is free to then choose the appropriate cluster from the tree structure, we will see what that is the in the video as we will see what that is in the next few slides.

(Refer Slide Time: 01:35)

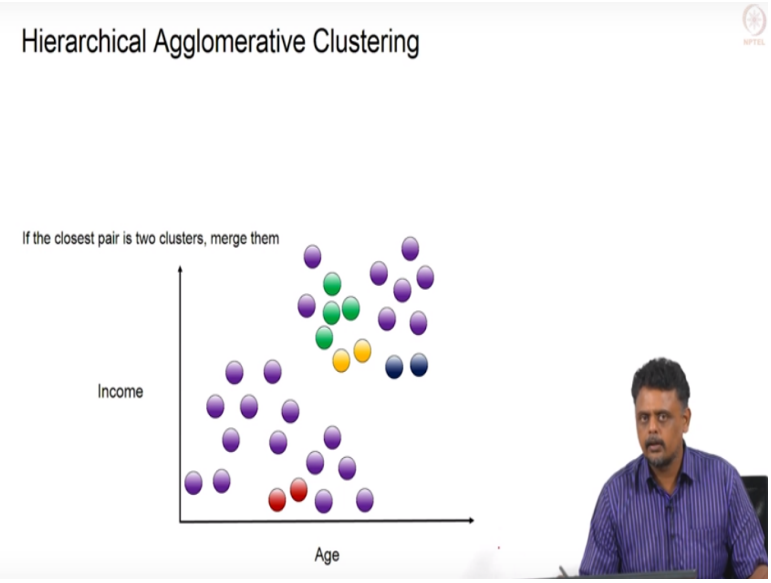


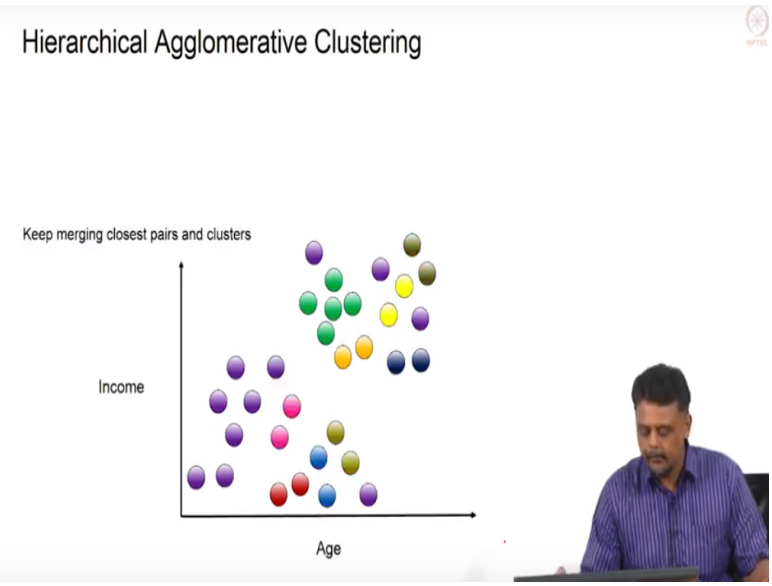


So an only of the algorithm let us consider the same data set as we had for the came in this algorithm where we had the income and age statistics of the user of particular website so it is plotted in 2D, so it is easy to visualize, so we have all this data points and hierarchical agglomerative clustering starts of by finding the closest square, so we treat each of this data points has a cluster by it and we find the closest square and merge amount to a cluster, so what do you mean by closest square it is a based on a dissimilarity matric, so basically in gradient distance we will see what the this various dissimilarity matrix are in the latest slide, it I also refer sometime as the linkage, so there are different type of linkage that we can use to find the least

this similar pair and group them, so based on dissimilarity matrix we will find the closest square in term of the dissimilarity matrix that is the least dissimilar and merge them into a cluster.

(Refer Slide Time: 02:48)

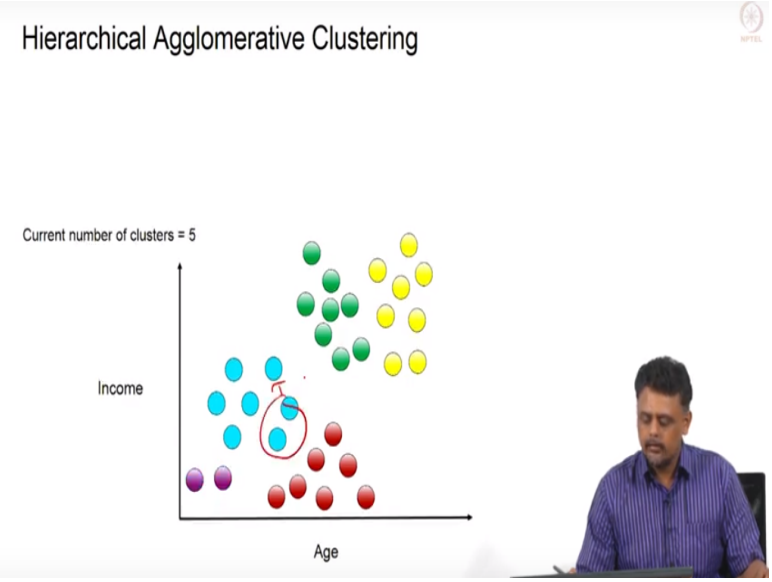


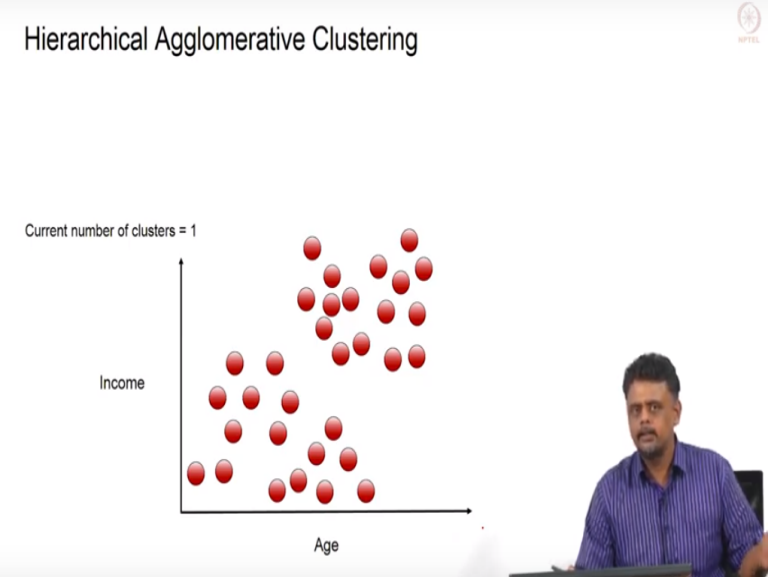


So the next iteration give fine the next closest square and merge them so in the first iteration we have this two and in the second iteration we found this, so we continue this way so we will find two at a time, so we will be essentially we looking at two data point or two cluster, so that time find the dissimilarity matrix and merge those with the least dissimilarity matrix so we can continue doing that so we will have this case so far about four clusters, so this a created four clusters and if the closest pair is two clusters and we can the clusters themselves.

So we have this two here and in the next iteration we merge cluster if they happen to be the closest square, so we will see how to determine the distance between two clusters even this clusters are not individual data points we are going to consider if they contain multiple data points we will look at how to, that is where the linkage comes in see how to decide which one are the closest pair similar case, so for now let us assume that is figure out a way to find out which two clusters are dissimilar based on some a clean an matrix and we are just merging them based on the least similarity so we can continue this way we can keep merging through a spares and of data point as well as the clusters.

(Refer Slide Time: 04:13)

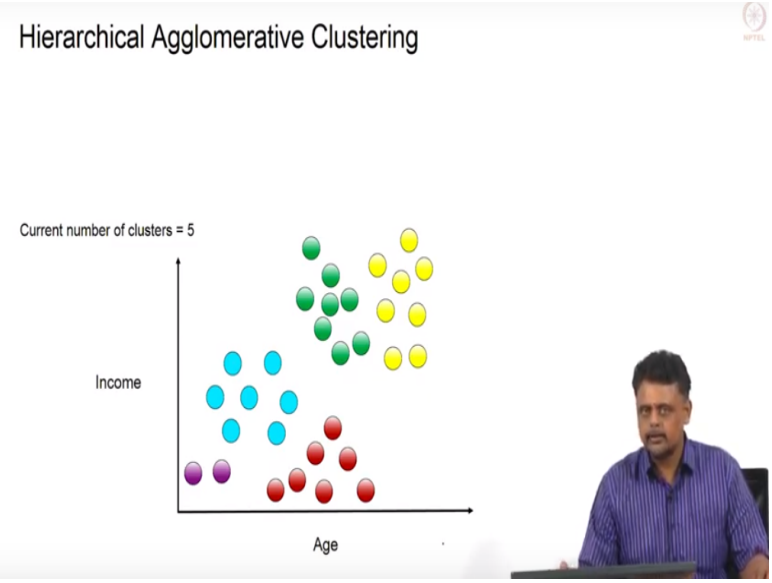




And till just you can see we keep merging we will have the number of clusters will begin to reduce, so let us start here we have what five cluster in this particular sorry six clusters in this scenario and each color denoting a cluster and if you can merge them we get so we have merge this two point of this cluster and we have five, so then we have here to there in a four clusters three cluster two and one, so finally the algorithm stop when the entire data set is assign to eight clusters, so that is the idea behind this, so we can start out with individual data points pair them up based on a dissimilarity matrix to least to get dissimilar once get to merge cluster.

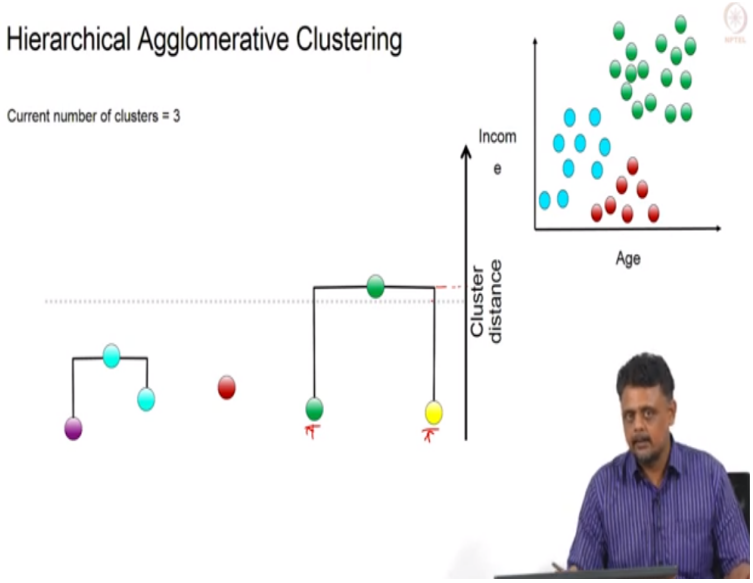
So as we keep the progressing through the data set at some points we will have to start merging clusters also based on the dissimilarity matrix and soon at the last of the algorithm will result in one cluster so that will be this is a bottom of approach this is talked on about also wherein we can divide the data set based on the dissimilarity matrix and keep dividing till data point where each individual data point is a clusters so this are two ways of doing it we are looking at the bottom of approach now.

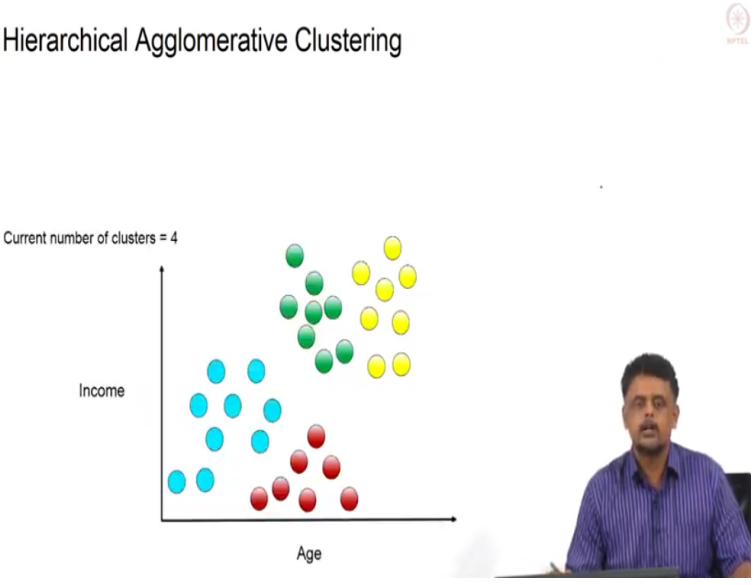
(Refer Slide Time: 05:37)



So let us consider the points where we have a both five clusters so this is situations and we will see how we can actually decide on the number of clusters, so it is slightly subjective but visually it is very appearing

(Refer Slide Time: 05:50)





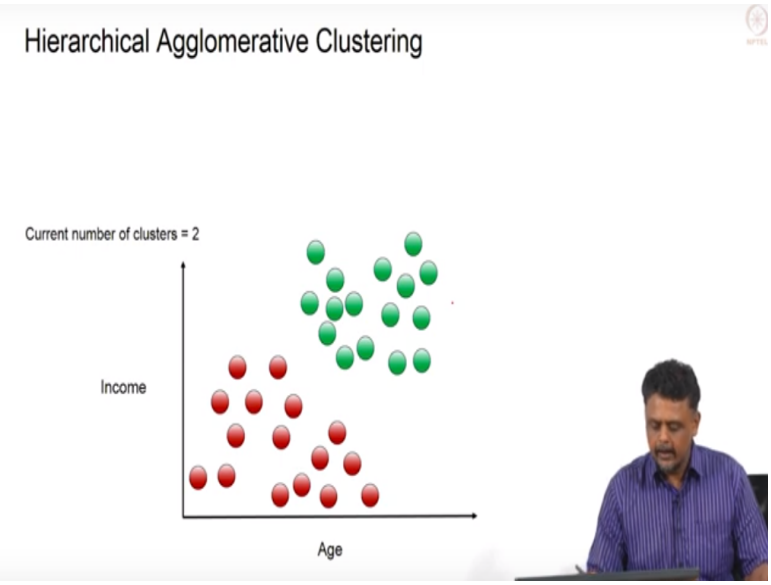
So in this case so these is the distance is two dimensional problem we can visualize it so here we have each of these clusters plotted here, so we will call this cluster index if you want to call in that actually does not this access does not have label pretty much so we have plotted of this cluster at a height, so this is the cluster so this is the distance that ultimately we are talking about, so when we merge to cluster it use rise to a new cluster and the height is basically the dissimilarity measure between the two cluster that we merged, so now we have five clusters which are resulted from the merging of several clusters and the heights along this axis is

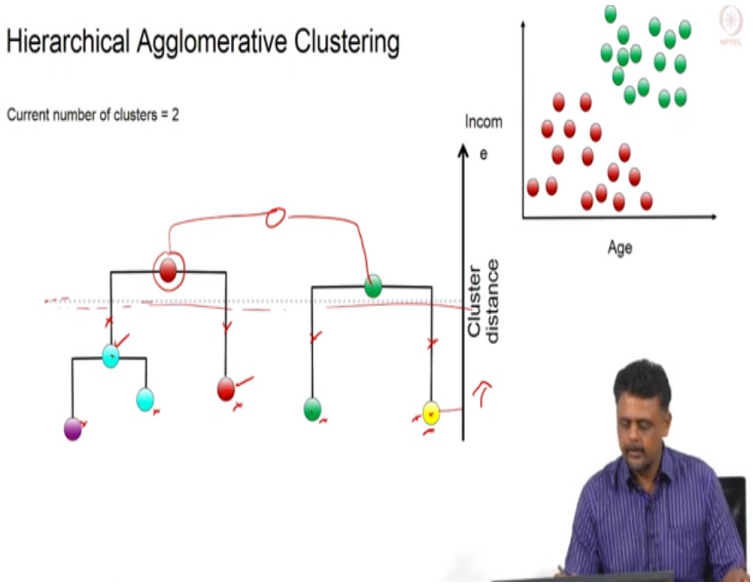
basically the dissimilarity matric between the cluster that were merged to obtain this clusters, this of them as each of them as a single clusters.

So now if we so we start of it five, so let us say we merge the closest so we will get four cluster, so if you go back here so we just merge this with that so we have four clusters now then if we want to visualize it this way here we have merge this two clusters to get a new one here and this height is basically the dissimilarity measure that we compute between this two cluster so similarly we can go ahead and merge two other cluster to a three cluster that is what this two where merged and again this height is the integration of the dissimilarity measure between this two clusters.

So here is slightly intuitive in a sense that we realize that as we keep merging clusters the dissimilarity matric keeps increasing, so it is automatically increasing function because we would merge points which are very close to each other and this start to merge clusters we will end up merging cluster slightly very dissimilar so that's way this height will start increase, so that use your very clear idea and how we can merge the clusters or how we can figure out how many cluster are there in a direct.

(Refer Slide Time: 08:16)





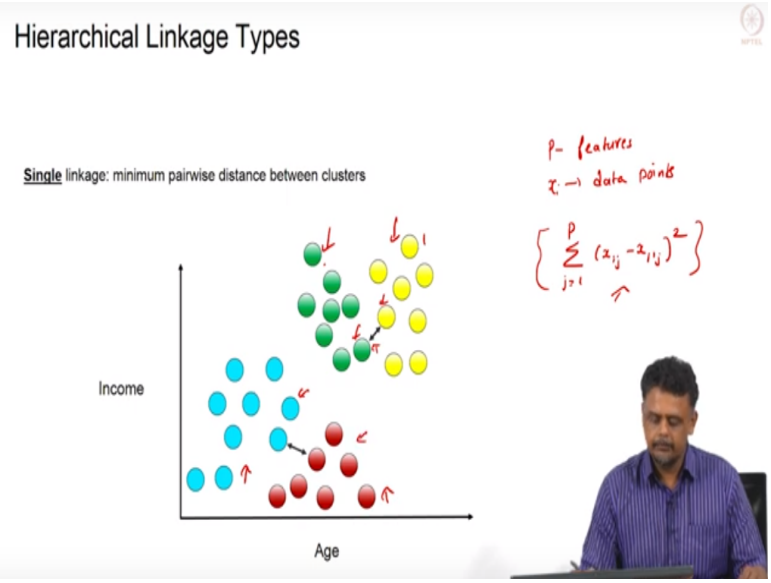
So then we can go to the point where you have both two clusters and of course here this Q raise to this one, so finally what we can do is merge this two to get something here at that, so then how do we do the split so that's why we look at this great line here so based on the height along this axis so this is the cluster distance we can draw line that is basically threshold at which we want to create the cluster, so when we cut and cross this way we will end up hiring one two three four unique clusters in a data set, because we see that merging these two cluster leads to very large similarity matrix same way merging these two cluster this one and here give rise to a very large dissimilarity matrix.

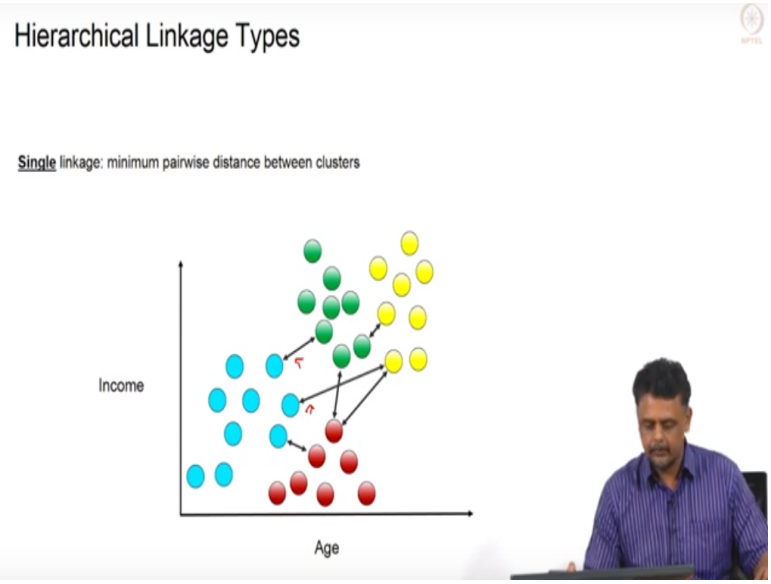
So we can just make a cut here it give rise to four unique cluster so the idea behind using this cluster distance is a threshold is that if you look at the so each of this point here this are the nodes of the tree that we have construct here they represent clusters and all the data point in that particular cluster or basically more similar to each other than to data points in another cluster represented by a same node at the same level basically, so we obtain this data points by combining several clusters data points and it is plotted of the height of the dissimilarity matric that we calculated for merging for data point.

So even though all of them all pretty much around the same height the point is that all the elements the cluster are more similar to each other than two elements in another cluster at the same height along this axis, so that is the point behind this DENDROGRAM, so this what is

called DENDROGRAM which basically a binary tree and we decide where to make the cut, so that we end up with unique cluster.

(Refer Slide Time: 10:24)





So how we talked about dissimilarity matric and now we have to see how are this dissimilarity matrix calculated, so basically just gradient distance between the points there are different type of them, so we will look at one by one so what do you mean by gradient distance, let us say you have a P features and XI denotes data points, so what we like to calculate is basically this between any two data points let us say I prime we want to calculate the gradient distance so that

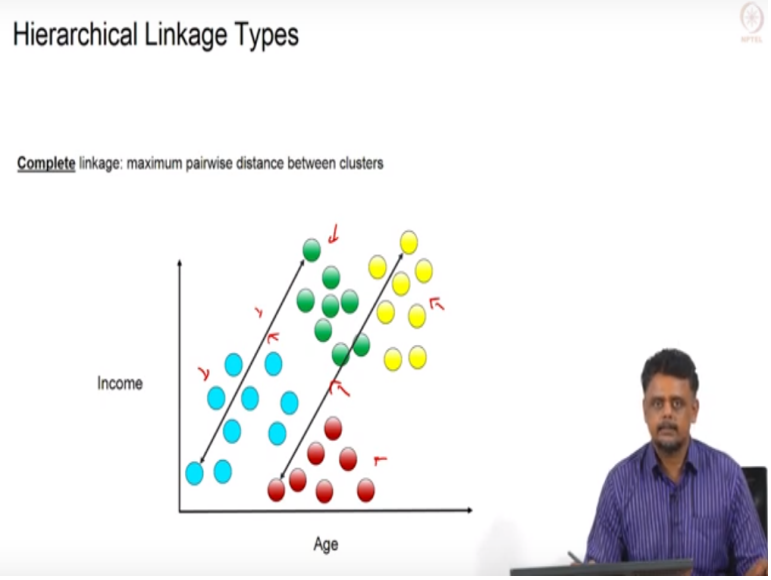
will be X IJ and this X I prime J square, so J would go from one to P, so that is this is the matric that we typically calculate.

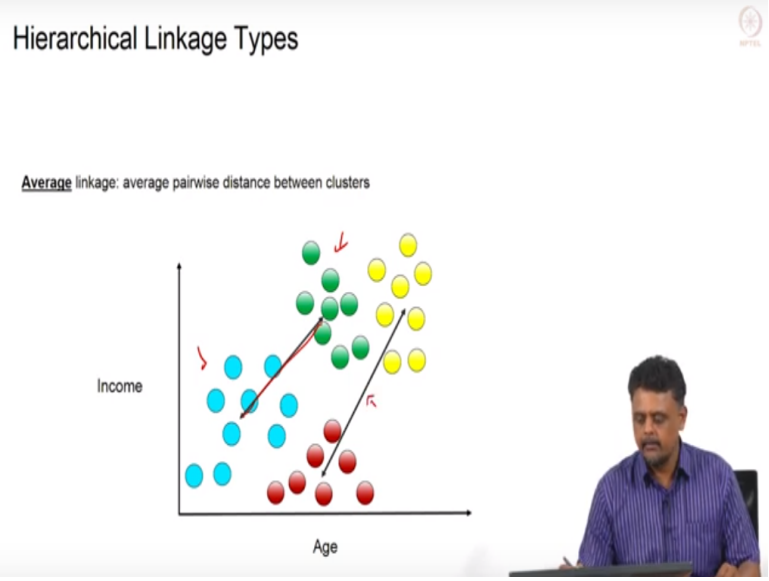
So we can calculate this matric between any two pairs of data point that we want to merge but than we are trying to merge clusters what do we do, so than that is want the linkage becomes slightly more important, so here we have different type of linkages the first one we look at this single linkage which is a the minimum pair wise distance between cluster, so what we shown here, so all the pairwise distance between this to and this two clusters can see that this is way to do that this is two calculate this distance here between all the points in one cluster and all the points in the other cluster.

So we will get pairwise distances between the two cluster within is all the points in both the cluster across clusters and you will choose the as a dissimilarity matrix you will choose the minimum pairwise among them, so that will be the before in this case we can see that this two point are the closest, so then that will be the distance the minimum distance pairwise distance between the cluster same thing for this two clusters here this two here.

So we can do the same so the four clusters so than we should be able to calculate the minimum pairwise distance between each of them and that's what this black arrows represents, so between the each of the color distance they can calculate the minimum pairwise distance and use that as a dissimilarity matric.

(Refer Slide Time: 12:56)





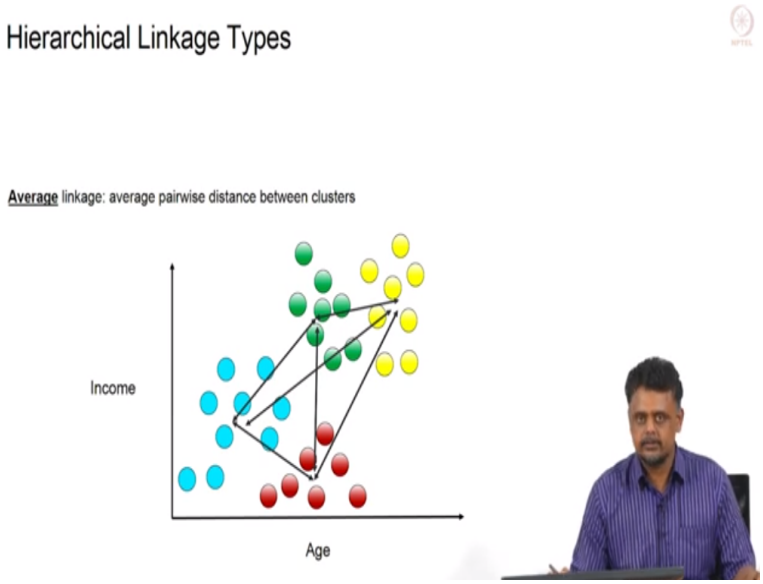
The complete linkage calculates the maximum pairwise distance within cluster as a name implied is basically calculate the maximum distance between the points in the cluster between the two cluster, so in this case as an illustration between this cluster here the green and the light blue this black arrow represent the maximum pairwise distance between the elements in the cluster or the data points in the cluster similarly between the yellow and the red this particular arrow represent the maximum pairwise distance between them.

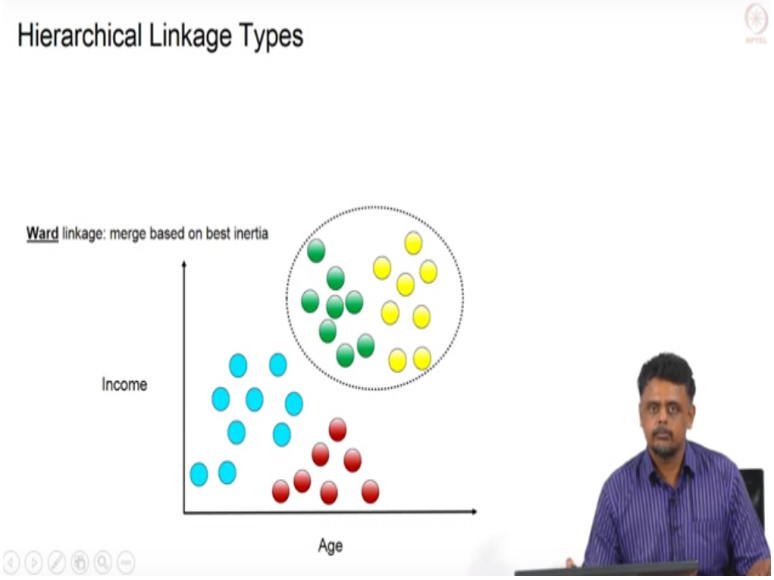
So as we saw earlier for single linkage we will do the same for all the cluster taking each pair at a time and calculating the maximum pairwise distance between them and we use that as a

dissimilarity matrix, so average linkage is a basically the average of the pairwise distances between the element cluster, so we take all the elements of cluster one and all the elements of cluster this case cluster two and this case blue and green and you calculate the distance between or pairs of them taking one from each cluster an you do the average.

So it is the average the if you take the individual distances and do an average of all those distance and that is the dissimilarity matrix, so this case in this can have expected this arrow here represent the average distance between the green and the blue cluster similarly this arrow represent the average distance between the red and the yellow cluster.

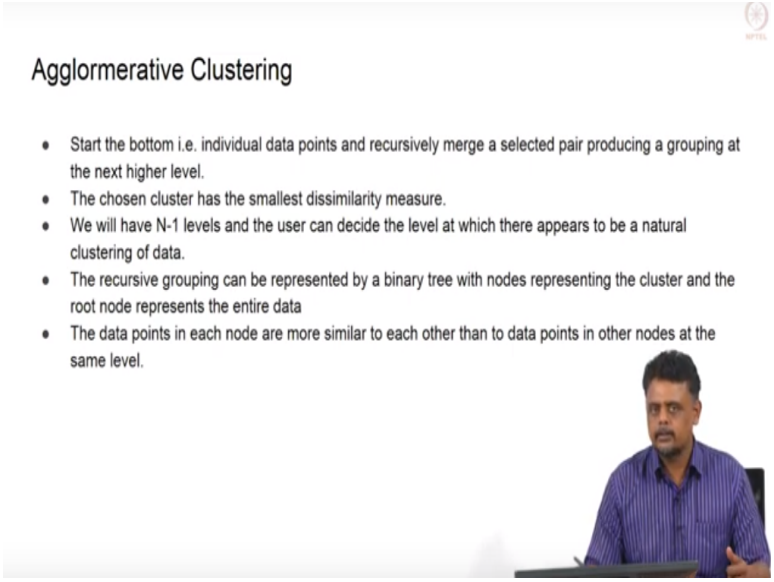
(Refer Slide Time: 14:42)





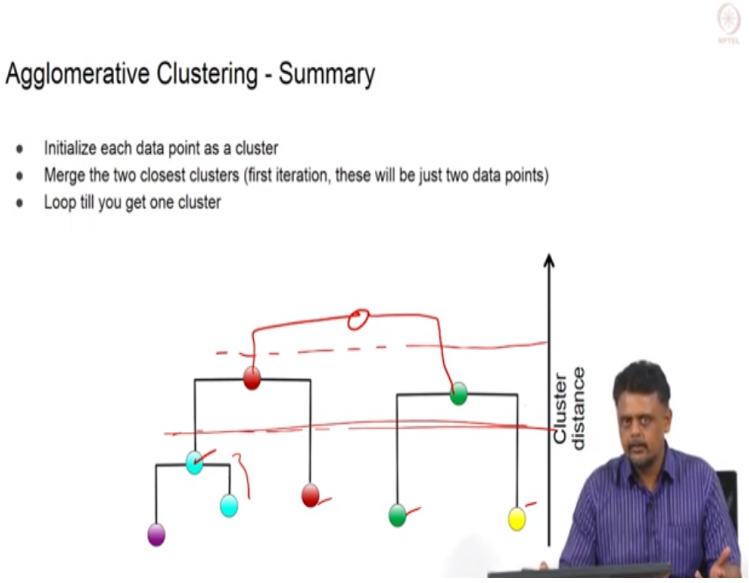
So we can show in this plot shows the all the average distances between all pairs of cluster so finally we will come to a ward linkage or a centroid based linkage we just merge based on based inertia this inertia is basically what we use for the came in algorithm, so it is a distance between the centroid of the cluster so based on that than we can link so this would be merge if you are looking at centroid base distance at this two are the closest centroid and you would merge if you are looking at centroid base distance at this two are the closest centroid and you would merge up, so that is the dissimilarity matrix the basically the distance between the centroid of each of the clusters.

(Refer Slide Time: 15:25)



So agglomerative clustering summarize to you start at the bottom that is individual data points you treat each individual data point as a cluster and recursively merge them pair at a time producing a grouping at the next highest level so we will have N minus one levels and how do you merge the pair you merge a pair based on the smallest dissimilarity measure between the pair we saw that the different dissimilarity measure we used the recursive grouping lead to a DENDROGRAM or a binary tree where the node representing the cluster and the root node representing the entire data set and as I mentioned earlier there is data point on each of the nodes more similar to each other than the data points other nodes at the same level and the level along the DENDROGRAM is depend on the dissimilarity measure between the two clusters that we are combine to give there node, so that is the that is how that give rise to the DENDROGRAM.

(Refer Slide Time: 16:25)



So to summarize the algorithm itself proceed likes this so initially each data point as a cluster and merge the two closest clusters so initial iteration mostly be that is the data point and we keep looping till you get a single cluster and then so in this case this two will merge we saw earlier just reproducing the graph here and it is up to the user to determine what is the ideal numbers of cluster so you can make the cut at whatever level you choose, so for instance you can make the cut at this level or you can make the cut at this level.

So you can look at this it makes more sense to make cut here because of the large change in the cluster distance just the dissimilarity matrix and that you will hope the data that you rise to one

two three and four, this is the one two three and four unique cluster from the data set, so with this we conclude a our short for into our supervised clustering techniques to summarize this two merger that we look at L-means and the agglomerative clustering techniques they are both useful when we have large amounts of unlabeled data and we are just trying to figure out and underlying structure in that data set, given that we do not have any pure information regarding the data set this techniques are very useful and we can also be use very large data set.

So typically they have been used for instance agglomerative clustering has been use in DNA micro rays, came in has been use for image processing so on and so forth, so even with very large data set it is possible to use this techniques given the absences of any other a pure information or label information and gives you a variety of the what they grouping is like in the data thank you.