Lecture – 37
K – means Clustering

So, we are into the last theory lecture of this course I am going to talk about K-means Clustering today following this lecture there will be a demonstration of this technique on a case study by the TA's of this course.

(Refer Slide Time: 00:53)

So, what is K-means clustering? So, K-means clustering is a technique that you can use to cluster or partition a certain number of observations, let us say N observations in to K clusters. The number of clusters which is K is something that you either choose or you can run the algorithm for several case and find what is an optimum number of clusters that this data should be partitioned into.

Just a little bit of semantics I am teaching in clustering here under the heading of classification. In general typical classification algorithms that we see are usually super waste algorithms; in the sense that the data is partitioned into different classes and these labels are generally given. So, these are labeled data points and the classification algorithms job is to actually find decision boundaries between these different classes. So, those are supervised algorithms.

So, an example of that would be the k-nearest neighbor that we saw before where we have label data and when you get a test data you kind of bin it into the most likely class that it belongs to. However, many of the clustering algorithms are unsupervised algorithms in the sense that you have N observations as we mentioned here, but they are not really labeled into different classes. So, you might think of clustering as slightly different from classification, where you are actually just finding out if there are data points which share some common characteristics or attributes.

However, as far as this course is concerned we would still think of this as some form of classification or categorization of data into groups just that we do not know how many groups are there a priori. However, for a clustering technique to be useful once you partition this data into different groups as a second step one would like to look at whether there are certain characteristics that kind of pop out from each of this group to understand and label these individual groups in some way or the other.

So, in that sense we would still think of this as some form of categorization which I could also call as classification into different groups without any supervision. So, having said all of that K-means is one of the simplest unsupervised algorithms, where if you give the number of clusters or number of categories that exist in the data then this algorithm will partition these N observations into these categories. Now, this algorithm works very well for all distance matrix you again need a distance metric as we will see where we can clearly define a mean for the samples.

So, when we were talking about optimization for data science I told you that you know all kinds of algorithms that you come up with in machine learning there will be some optimization basis for these algorithms. So, let us start by describing what a K-means clustering optimizes or what is objective that is driving the K-means clustering algorithm. So, as we described in the previous slide there are N observations x_1 to x_N and we are asking the algorithm to partition this into K clusters. So, what does it mean when we say we partition it into K clusters? So, we will generate k sets s_1 to s_k and we will say this data belongs to this set and this data belongs to the other set and so on.

So, to give a very simple example let us say you have this observations x_1, x_2 all the way up to x_N and just for the sake of argument let us take that we are going to partition this data into 2 clusters ok. So, there is going to be one set for one cluster, cluster 1 and there is going to be another set for the other cluster, cluster 2. Now, the job of K-means clustering algorithm would be to put these data points into these two bins ok. So, let us say this could go here this could go here x_3 could go here x_N could go here maybe an x_4 could go here and so on.

So, all you are doing is you are taking all of these data points and putting them into two bins and while you do it what you are looking for really in a clustering algorithm is to make sure that all the data points that are in this bin have certain characteristics which are like in the sense that if I take two data points here and two data points here, these two

data points will be more like and these two data points will be more like each other, but if I take a data point here and here they will be in some sense unlike each other. So, if you cluster the data this way then it is something that we can use where we could say look all of these data points share certain common characteristics and then we are going to make some judgments based on what those characteristics are.

So, what we really would like to do is the following: we would like to keep these as compact as possible so that like data points are grouped together which would translate to minimizing within cluster sum of square distances. So, I want this to be a compact group. So, a mathematical way of doing this would be the following. So, if I have k sets into which I am putting all of this in, you take set by set and then make sure that if you calculate a mean for all of this data the difference between the data and the mean squared in a nonsense is as minimum as possible for each cluster and if you have k clusters you kind of sum all of them together and then say I want a minimum for this whole function. So, this is a basic idea of K-means clustering.

So, there are there is a double summation the first summation is for all the data that has been identified to belong to a set, I will calculate the mean of that set and I will find out a solution for this these means in such a way that this within cluster distance x which is in the set minus I mean is minimized not only for one cluster, but for all clusters. So, this is how you define this objective. So, this is the objective function that is being optimized and as we have been mentioned mentioning before this μ_i is mean of all the points in set s_i.

(Refer Slide Time: 09:09)
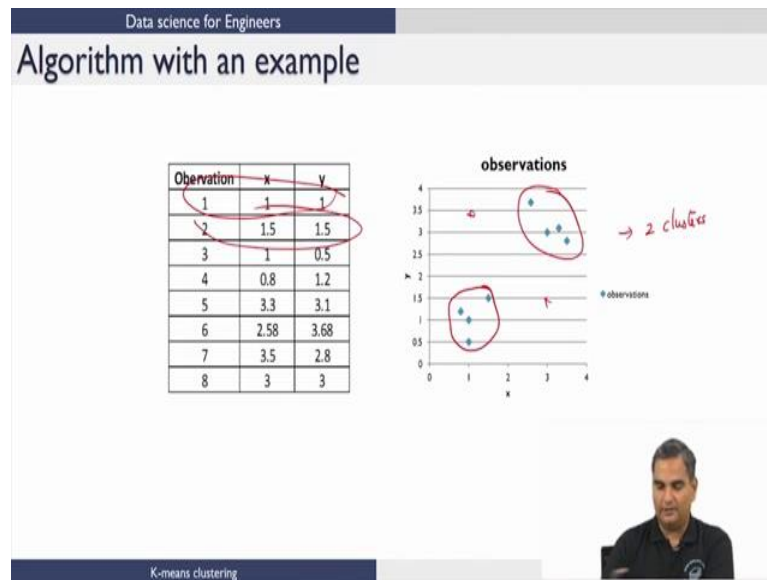
Now, let us look at how an algorithm such as K-means works it is a very simple idea. So, let us again illustrate this with a very simple example. Let us say there are two dimensions that we are interested in x and y and there are eight observations. So, this is one data point, this is another data point. So, there are eight data points and if you simply plot this you would you would see this and when we talk about cluster. So, notionally you would think that there is a very clear separation of this data into 2 clusters right.

Now, again as I mentioned before in one of the earlier lectures on data science very easy to see this in 2-dimensions, but the minute we increase the number of dimensions and the minute we increase the complexity of the organization of the data which you will see at the end of this lecture itself you will you will find that finding the number of clusters is really not that obvious. In any case let us say we want to identify or partition this data into different clusters and let us assume for the sake of illustration with this example that we are interested in partitioning this data into 2 clusters.

So, we are interested in partitioning this data into 2 clusters. So, right at the beginning we have no information right. So, we do not know that this all belong to one cluster all of these data points belong to another cluster, we are just saying that are going to be 2 clusters that is it. So, how do we find the 2 clusters because we have no information labels for this and this is an unsupervised algorithm. What we do is we know ultimately there are going to be 2 clusters. So, what we are going to do is, we are going to start off 2

cluster centers in some random location. So, that is the first thing that we are going to do. So, you could start off 2 clusters somewhere here and here or you could actually pick some points in the data itself to start these 2 clusters.

(Refer Slide Time: 11:42)
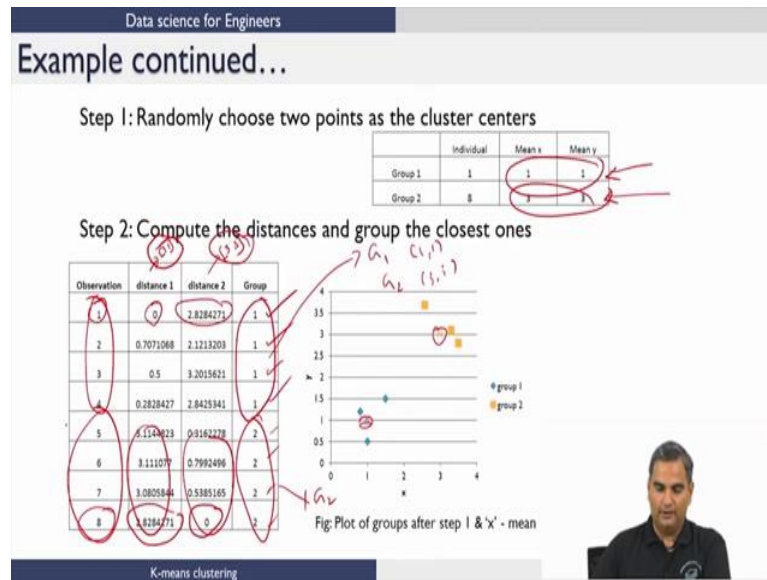
So, for example let us say we randomly choose two points as cluster centers. So, let us say one point that we have chosen is 1, 1. So, which would be this point here and let us assume this is what is group 1 and let us assume that for group 2 we choose 0.33 which is here. The way we have chosen this if we go back to the previous slide and look at this the two cluster centers have been picked from the data itself. So, the one group was observation 1 and the other group centre was observation 8.

Now, you could do this or like I mentioned before you could pick a point which is not there in the data also and we will see the impact of choosing this cluster centers later in this lecture. Now, that we have two cluster centers then what this algorithm does is the following it finds for every data point in our database this algorithm first finds out the distance of that data point from each one of this cluster centers. So, in this table we have distance 1 which is the distance from the point 1, 1 and we have distance 2 which is the distance from the point 3, 3.

Now, if you notice since the first point from the data itself, ie, the distance of 1, 1 from 1, 1 is 0. So, you see that distance 1 is 0 and distance 2 is the distance of the point 1, 1 from 3, 3. Similarly, since we chose 3, 3 as representative of group 2 if you look at point 8

which was a 3, 3 point the distance of 3, 3 from 3, 3 is 0 and this is the distance of 3, 3 from 1, 1 which will be the same as this. For every other point since they are not either 1, 1 or 3, 3 there will be distances you can calculate.

So, for example, this is a distance of the second point from 1, 1 and this is a distance of the second point from 3, 3 and if you go back to the previous slide the second point is the second point is actually 1.5, 1.5 and so on. So, you will go through here each of these points are different from 1, 1; 3, 3 you will generate these distances. So, for every point you will generate two distance one from 1, 1 other one from 3, 3.

Now, since we want all the points that are like 1, 1 to be in one group and all the points which are like 3, 3 to be in the other group, we use a distance as a metric for likeness. So, if a point is closer to 1, 1 then it is more like 1, 1 than 3, 3 and similarly if a point is close to 3, 3 it is more like 3, 3 than 1, 1. So, we are using a very very simple logic here.

So, you compare these two distances and whichever is a smaller distance you assign that point to that group. So, here distance 1 is less than distance 2. So, this is assigned to group 1 this observation which is basically the point 1, 1 the second observation again if you look at it distance 1 is less than distance 2. So, the second observation is also assigned to group 1 and you will notice through this process a third and fourth will be assigned to group 1 and 5, 6, 7, 8 will be assigned to group 2 because these distances are less than these distances.

So, by starting at some random initial point we have been able to group these data points into two different groups one group which is characterized by all these data points the other group which is characterized by these data points. Now, just as a quick note whether this is in 2-dimensions or N dimensions it is really does not matter because the distance formula simply translates and you could have done that for two classes in N dimensions as easily. So, while visualizing data in N dimensions are hard this calculation whether it is 2 or N dimension it is actually just the same.

Now, what you do is you know that these group positions are the centers of these groups were randomly chosen now, but we have now more information to update the centers because I know all of these data points belong to group 1 and all of these data points belong to group 2. So, a better representation for this group; so, initially the representation for this group was 1, 1, a better representation for this group would be the

mean of all of these four samples and the initial representation for group 2 was 3, 3, but a better representation for group 2 would be the mean of all of these points. So, that is step 3.

(Refer Slide Time: 17:23)

So, we compute the new mean and because group 1 has points 1, 2, 3, 4, we do a mean of those points and the x is 1.075 and y is 1.05. Similarly, for group 2 we do the mean of the labels are the data points 5, 6, 7, 8 and you will see the mean here. So, notice how from 1, 1 and 3, 3 the mean has been updated in this case because we chose a very simple example the updation is only slight. So, these points move a little bit.

Now, you redo the same computation because I still have the same eight observations but now, group 1 is represented not by 1, 1 but by 1.075 and 1.05 and group 2 is represented by 3.095 and 3.145 and not 3, 3. So, for each of these points you can again calculate a distance 1 and distance 2. And, notice previously this distance was 0 because the representative point was 1, 1. Now that the representative point has become 1, 1.075 and 1.05 this distance is no more 0, but it is still a small number.

So, for each of these data points with these new, new means you calculate these distances and again use the same logic to see whether distance 1 is smaller or distance 2 is smaller and depending on that you assign these groups. Now, if we notice that by doing this there was no assignment reassignment that happened. So, whatever were the samples that were

originally in group 1 remain in group 1 and whatever are the samples which are in group 2 remain in group 2.

So, if you again compute a mean because the points have not changed the mean is not going to change. So, even after this the mean will be the same. So, if you keep repeating this process there is no never going to be any reassignment. So, this clustering procedure stops. Now, if it were the case that because of this new mean let us say for example, if this had gone into group 2 and let us say this and this had gone into group 1, then correspondingly you have to collect all the points in group 1 and calculate a new mean collect all the points in group 2 and calculate a new mean and then do this process again.

(Refer Slide Time: 19:50)

And, you keep doing this process till the mean change is negligible or no reassignment. So, one of these two things happen, then you can stop the process. So, this is a very very simple technique. Now, you notice this technique is very very easily implementable it does not matter the dimensionality of the data you could have 20 variables, 30 variables the procedure remains the same. The only thing is we have to specify how many clusters that are there in the data and also remember that this is a unsupervised learning and so, we originally do not know any labels.

But, at the end of this process at least what we have done is we have categorized this data into classes or groups. Now, if you find something specific about this group by further analysis then you could basically sometimes maybe even be able to label these groups

and the broad categories. If that is not possible at least you know from a distance viewpoint that these two might be different behavior from the system viewpoint. So, from an engineering viewpoint why would something like this be important?

If I let us say give you data for several variables temperatures, pressures, concentrations and so on flow for several variables, then you could run a clustering algorithm purely on this data and then say I find actually that there are two clusters of this data. Then, a natural question and engineer might ask is if the process is stable I would expect all of the data points to be like, but since it seems like there are two distinct groups of data either both or normal, but there is some reason why there is this two distinct group or maybe one is normal and one is not really normal, then you can actually go on probe of these two groups which is normal which is not normal and so on.

So, in that sense an algorithm like this allows you to work with just raw data without any annotation; what I mean by annotation here is without any more information in terms of labeling of the data and then start making some judgments about how the data might be organized. And, you can look at this multi-dimensional data and then look at what is the optimum number of clusters maybe there are five groups in this multi-dimensional data which would be impossible to find out by just looking at an excel sheet.

But, once you do an algorithm like this then it maybe organizes this into 5 or 6 or 7 groups then that allows you to go and probe more into what these groups might mean in terms of process operations and so on. So, it is an important algorithm in that sense. Now, we kept talking about finding the number of clusters, till now I said you let the algorithm know how many clusters you want to look for, but there is something called an elbow method where you look at the results of the clustering for different number of clusters and use a graph to find out what is an optimum number of clusters. So, this is called an elbow method.

And, you will see a demonstration of this in an example. The case study that follows this lecture you will see how this plot looks and how you can make judgments about the optimal number of clusters that you should use. So, basically this approach uses what is called a percentage of variance explained as a function of number of clusters. But those are very typical looking plots and you can look at those and then be able to figure out what is the optimal number of clusters.

So, I explained how in an unsupervised fashion you can actually start making sense out of data and this is a very important idea for engineers because this kind of allows a first level categorization of data just purely based on data and then once that is done then one could bring in their domain knowledge to understand these classes and then see whether there are some judgments that one could make.

Couple of disadvantages of K-means that I would like to mention: the algorithm can be quite sensitive to the initial guess that you use. It is very easy to see why this might be. So, let me show you a very simple example. So, let us say I have data like this. So, if my if I start my cluster points initial cluster points here and here you can very clearly see by the algorithm all these data points will be closer to this.

So, all of this will be assigned to this group and all of these data points are closer to this. So, they will be assigned to this group, then you will calculate a mean and once a mean is calculated there will never be any reassignment possible afterwards then you have clearly these two clusters very well separated. However, if just to make this point supposing I start these two cluster centers for example, one here and one somewhere here then what is going to happen is that when you calculate the distance between this cluster and all of these data points and this cluster center.

And, all of these data points it is going to happen that all these data points are going to be closer to this and all of these data points are going to be closer to this than this point. So,

after the first round of the clustering calculations you will see that the center might not even move right because the mean of this and this might be somewhere in the center. So, this will never move, but the algorithm will say all of these data points belong to this center and this center will never have any data points for it. So, this is a very trivial case, but it just still makes the point that if you use the same algorithm depending on how you start your cluster centers you can get different results.

So, you would like to avoid situations like this and these are actually easy situations avoid, but when you have multi-dimensional data and lots of data and which you cannot visualize like I showed you here it turns out it is not really that obvious to see how you should initialize. So, there are ways of solving this problem, but that is something to keep in mind. So, every time you run an algorithm if the initial guesses are different you are likely to get at least minor differences in your results.

And, the other important thing to notice look at how I have been plotting this data, I typically I have been plotting to this data, so that you know the clusters are in general spherical. But, let us say if I have data like this where you know all of this belongs to one class and all of this belongs to another class, now K-means clustering can have difficulty with this kind of data simply because if you look at data within this class this point and this point are quite far though they are within the same class whereas, this point and this point might be closer than this point in this point.

So, if in an unsupervised fashion if you ask K-means clustering to work on this depending on where you start and so on, you might get different results. So, for example, if you start with three clusters you might in some instances find three clusters like this. So, though underneath the data is actually organized differently and if these happen to be two different classes in reality, in an unsupervised fashion when you run the K-means clustering algorithm you might say all of this belongs to one class; all of this belongs to another class and all of this belongs to another class and so on.

So, these kinds of issues could be there. Of course, as I said before there are other clustering algorithms which will quite easily handle data that is organized like this, but if you were thinking about K-means then these are some of the things to think about. So, with this we come to the conclusion of the theory part of this course on data science for engineers, there will be one more lecture which will demonstrate the use of K-means on

a case study. With that all the material that we intended to cover would have been covered um. I hope this was an useful course for you we tried to pick these data science techniques, so that you get a good flavor of the things that you think about and also actually techniques that you can use for some problems right away.

But, more importantly our hope has been that this has increased your interest in this field of data science and as you can see that there are several fascinating aspects to think about when one thinks about machine learning algorithms and so on. And, this course would have hopefully given you a good feel for the kind of thinking and aptitude that you might need to follow up on data science and also the mathematical foundations that are going to be quite important as you try to learn more advanced and complicated machine learning techniques either on your own or through courses such as this that are available.

Thank you very much and I hope a large number of you plan to take the exams and get a certification out of this course.

Thanks again.