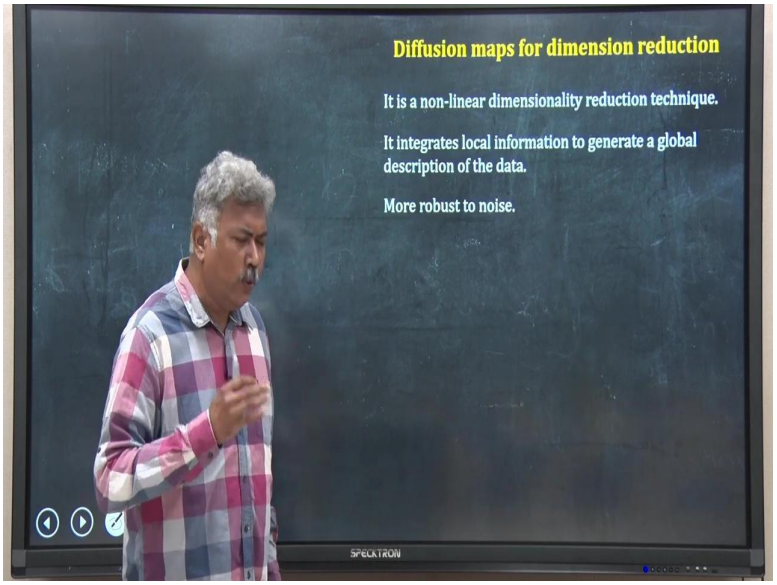
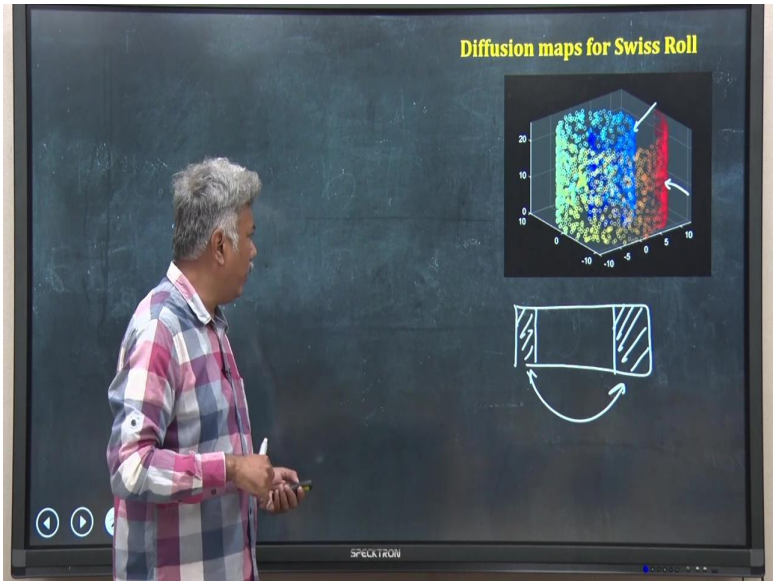
Data Analysis for Biologists
Professor Biplab Bose
Department of Biosciences & Bioengineering
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology, Guwahati
Diffusion Maps

(Refer Slide Time: 0:30)



Hello everyone, welcome back. In this lecture, we learn diffusion maps for dimension reduction diffusion maps is a nonlinear method to reduce the dimension of your data and it is said that it actually integrates the local information in your data to give a global representation of the whole data set. And it has been shown that it is much more robust with respect to when you have a noise in your data set.

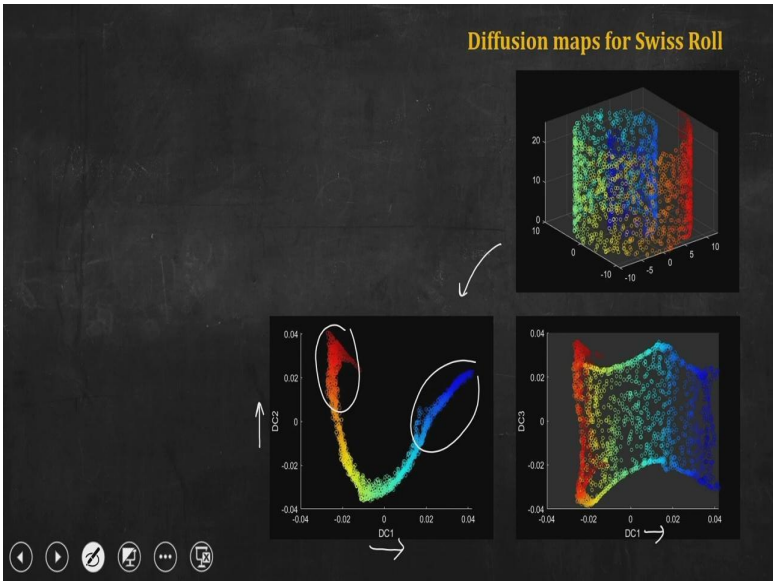
(Refer Slide Time: 1:13)



Before I go into detailed discussion of diffusion map and how it works, let me give you one example of diffusion maps. So, this is a typical Swiss Roll data very famous, what we have you can easily see, we have a Swiss roll like structure the data you can imagine as if the original data set is a strip and then we have rolled it in a Swiss Roll shape, concentric circle type thing. So, in this data, if you can see I have color codes.

So, these blue dot points they are locally close, because if I draw the original structure suppose, so, these blue data points may be somewhere here, they are all locally close to each other. Whereas, these rain data points are in another end and they are also very close to each other, but when I rolled it together, I rolled in a Swiss Roll structure, these two things has come close to each other. Now, I want to get back to the original local information local structure of the data by using diffusion map.

(Refer Slide Time: 2:18)

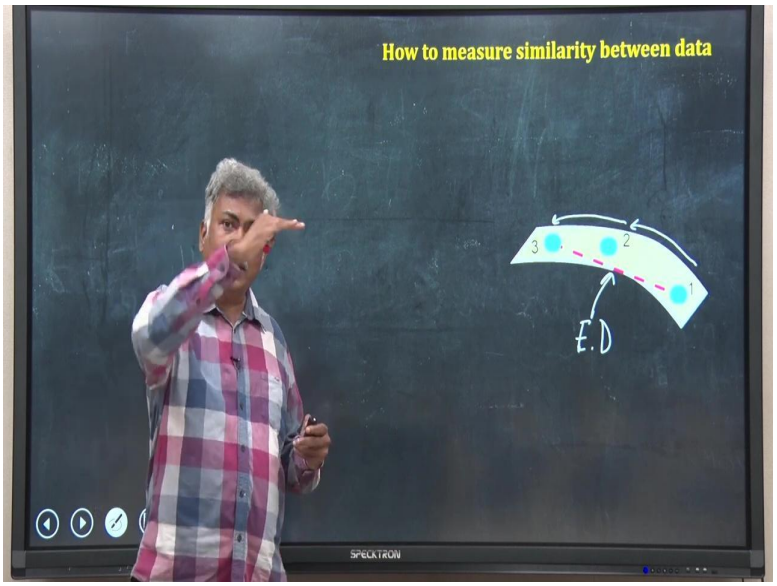


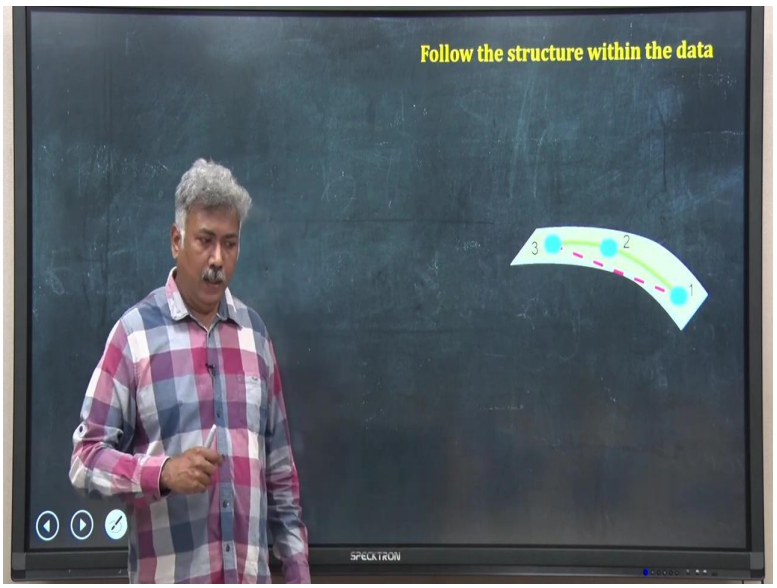
So, if I apply diffusion maps method on these, I get this type of result. So, if you see a in this case, I am representing the data the same Swiss roll data into the new dimension one is DC 1 and other one is DC 2, and you can easily see here the algorithm has segregated this blue data from the red data even in this representation, they are far away from each other.

So, as if I have opened up the swiss role, and I am representing the data as it is in the structure similarly, if I represented in DC 1 new coordinate and the third new coordinate, then also you can see the blue data and read data are separated and all the other data points are nicely arranged in between.

So, in this way the diffusion map has used the local data to give a global representation of the whole data set. Now, how do you perform diffusion maps dimension reduction, how the algorithm works, and the mathematics behind diffusion maps is beyond the scope of this course. So, I will not go in details of derivation of all these things, I will skip lots of mathematics and I will try to understand the principle of diffusion maps intuitively

(Refer Slide Time: 3:36)





See, whenever you are trying to create some lower dimensional representation of a higher dimension data, you always need some sort of measure of distance or similarity. And as I said that in diffusion maps, we want to integrate that local information that means, I want to know the local distance between different data points.

So, suppose I have a three dimensional data just three data points and they are 123 that we have shown. Now, they are you can imagine as if they are on a paper like this flat and in the three dimensional space that paper has got turned like this. Now, I want to calculate the similarity or distance between these three data points pair wise.

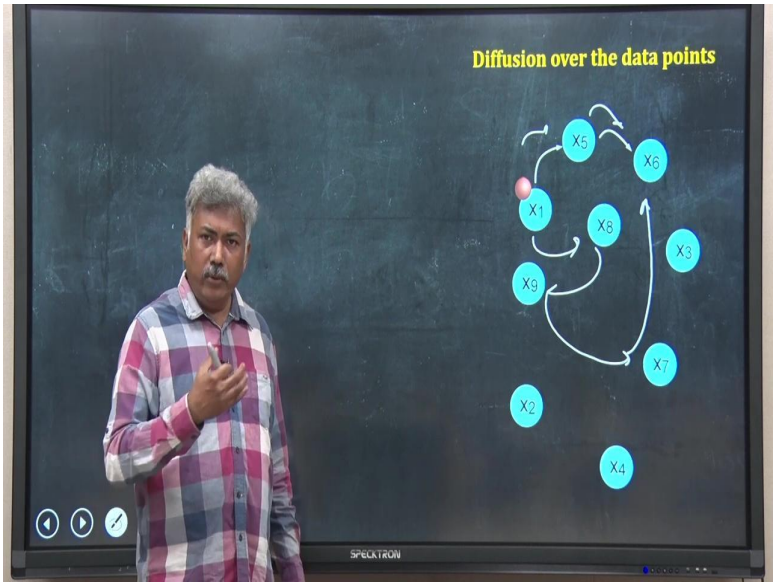
So, suppose I want to know the distance between one and three. One way the easiest way to do these distance measurement or similarity measurement is to use Euclidean distance. So, if I measure the Euclidean distance, this red dotted line, so, this is Euclidean distance, you can see this Euclidean distance is nothing but direct flight as if you are measuring the areal distance between one and three.

So, that is the Euclidean distance. But in reality if you look into this data, you know that 3 is actually Not so close to one, if you have to reach 3, you can imagine that you are working over 1, you start from 1 and then you reach 2, then from 2 you follow and then you reach 3.

So, if you want to represent the local information and you want to retain that, then this Euclidean distance will cheat you, because it will say 3 is actually close to 1. But that is not true. If you follow the structure, the geometry of the data, you have to work from one to 2 and three, and that is quite a distance. That means this green line that I am showing here is the right distance measure.

So, how should I define that distance? How should I define that distance that will be a distance measure that will capture this as if I am moving along the structure the geometry of the data from 1 to 2 to 3 in diffusion maps, we use the idea of diffusion and diffusion distance for this particular purpose.

(Refer Slide Time: 6:05)

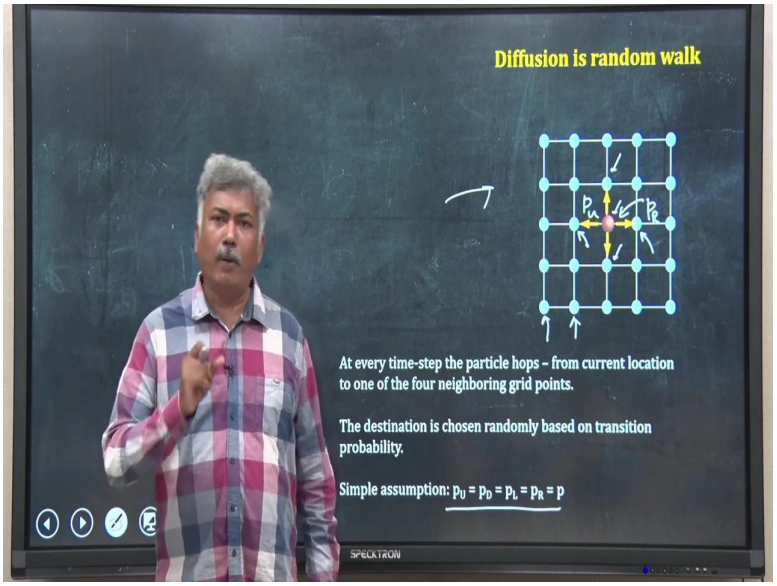


So, let me explain you what is diffusion distance? Now as I am talking of diffusion distance, so there must be some sort of diffusion. So, let us first understand what diffusion I am talking of here. Suppose I have nine data points, I have shown them starting with X1 up to X9 and they are scattered in a space they may be multi-dimensional data, maybe 10 dimension 100 dimension, we do not bother here.

I am representing them as a two dimensional now, what do you do you take a particle and hypothetical particle, imaginary particle and put it X1 and now let it diffuse in this data set. That means let it hop from X1 to X5 and X6 that we have shown here. This is diffusion, Similarly, it can also diffuse from X1 to X8, X8 to X9, then to X7, then to X6, so what is happening here, and imagining that there is an imaginary particle on that point and one data point and it is hopping from one data point to another.

That way, I just said few slides back that I want to walk from that point 1 to 2 to 3 following the structure of the of the geometry of the data set. So here this imaginary particle is diffusing from one data point hopping from one data point to another data point. Now, in this way, it can traverse the whole space hopping from one data point to another data point. Now, how can I numerically capture this diffusion process, there are many ways to mathematically capture this diffusion. The one way that we will use here is a probabilistic way.

(Refer Slide Time: 7:48)



$$p_U = p_D = p_L = p_R = p$$

Let me explain that with a very simple model of diffusion. This has nothing to do this particular slide has nothing to do with the diffusion in data space, but I am using it as an example. Suppose I want to study or simulate the diffusion of a particle on two dimensions, what will be the simplest model what I can do, I can consider these two dimensional space into a grid square grid with these grid points, the particle is here at the center get great point, in the next time point, it will either jump to the upper grid point or the lower grid point or the left one or the right one.

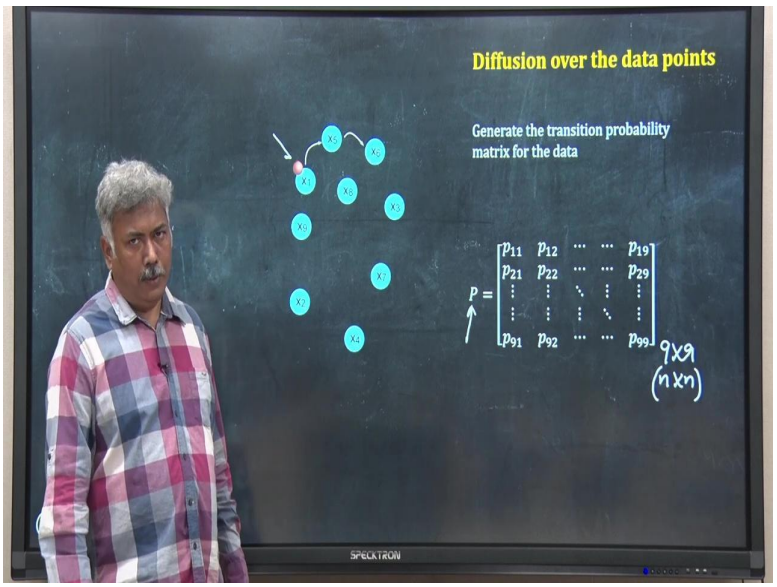
So it cannot stay between two grid points. So at every time point, as I increase the clock, as the clock ticks, it jumps from one-point grid point to another data point. That is diffusion. Now, how does it choose which grid point it has to go, it has four neighbors up down left and right. That is where the random walk comes, we associate certain probability of jump for example, I can see this is P-up probability of going to upgrade whereas, this jump from to the right can be P-R.

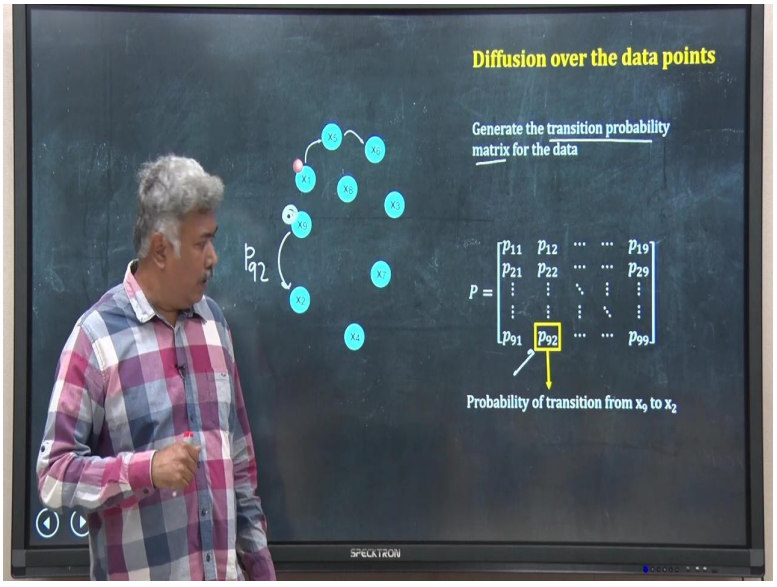
So, you assign certain probability of jump from one grid point to another grid point, which is the neighbor so in this way, I can actually capture and simulate and study the diffusion of this article from the central position across the whole space.

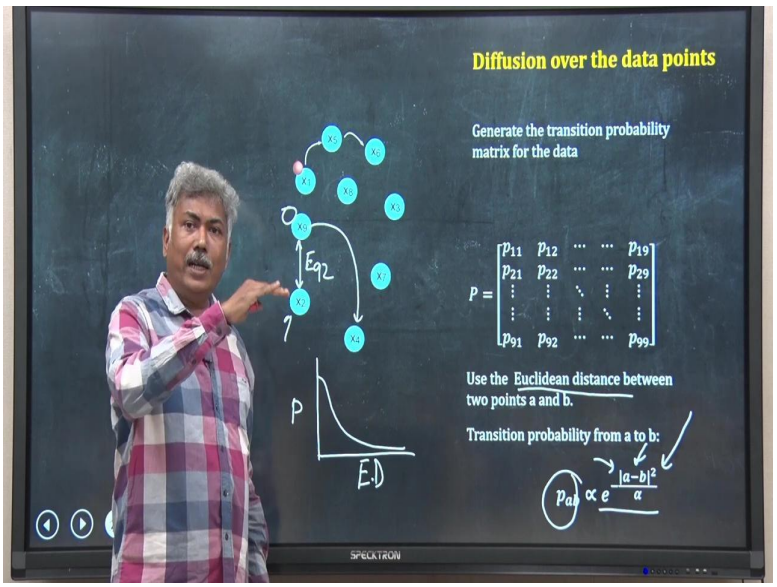
And when you simplify many times, what you can do is I can consider that all the probability all these four probability are same that will make your life easier, but there is no reason that I

have to do that I can actually consider different values for these probabilities also. So learning from these learning from this, what I will do will also in the diffusion in the data space will also assign some sort of probabilities.

(Refer Slide Time: 9:54)







p11	p12	...	...	p19
p22	p22	...	...	p29
..	:	...	:	:
..	:	:	...	:
p91	p92	...	...	p99

$$p_{ab} \propto e^{-\frac{|a-b|^2}{\alpha}}$$

So I am going back to my data space. I have a particle here in X1 and it will hop from one data point to another data point and this hopping has a probability. So, how are we present that probability, I can represent this probability using a matrix called P and you can easily see this is a 9 by 9 matrix. So, if I have N Sample N data point it will be N by N matrix, each element of this matrix represent a hopping probability. Let me explain one suppose this P_9_2 what is P_9_2, P_9_2 is the probability of hopping from 9 data point to the 2 data point.

So if the particle is here, right now, what is the probability it will jump to X2 the second data point that is given by P_9_2 in this way, rest of the entries in this matrix you can understand. So this is a transition probability matrix. And how do I get that matrix? Obviously, we do not assume any value for that we calculate this probability matrix P from data. To calculate that probability matrix what we do we take our good old friend Euclidean distance.

So what I do I measured the Euclidean distance between all the data point all the pairs. So I measured the Euclidean distance between X9 and X2. So suppose that is E_9_2. So then what I do I use that Euclidean distance and then I use this function, which is the exponential

function and you can easily identify this is half of the Gaussian distribution, because you have E to the power minus something divided by alpha.

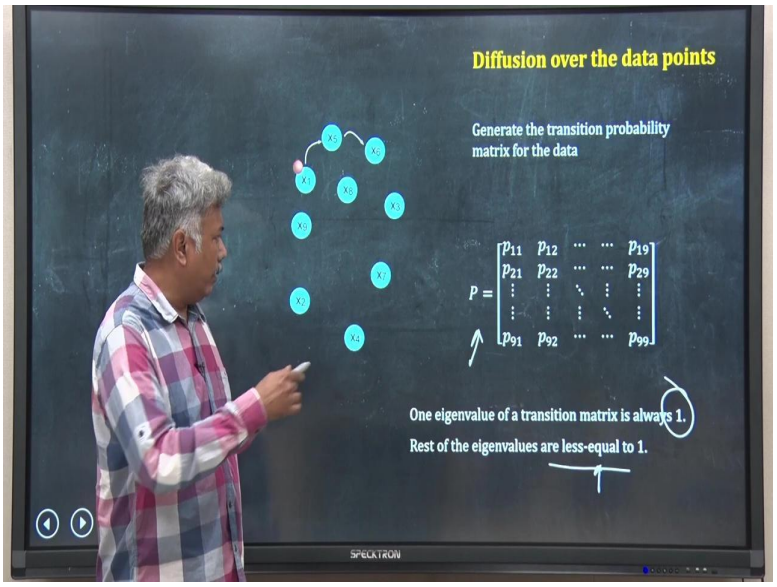
And this one is the square of the Euclidean distance between two data points A and B. So the probability of hopping from A to B is given by this function, which is called a Gaussian kernel, in essence, is an exponential decay type thing, you can easily see if my Euclidean distance, which is here in the denominator, Euclidean distance, if I increase, then this probability will look like this.

So as the distance increases, the probability of hopping decreases exponentially. And that is what you will expect. So P 9, if the particle is at P 9, it will hop to X2 with the probability, but the probability of hopping to X9 to X4 should be less than the probability of hopping from X9 to X2, because X2 is closer.

So, using this Gaussian kernel, we calculate the probability of hopping from one data point to another data point based upon that Euclidean distance in the original data space, we call it feature space. Now, once I have got this probability matrix, then we can actually move towards calculating and defining and calculating the diffusion distance which will be used to embed the original data to lower dimension.

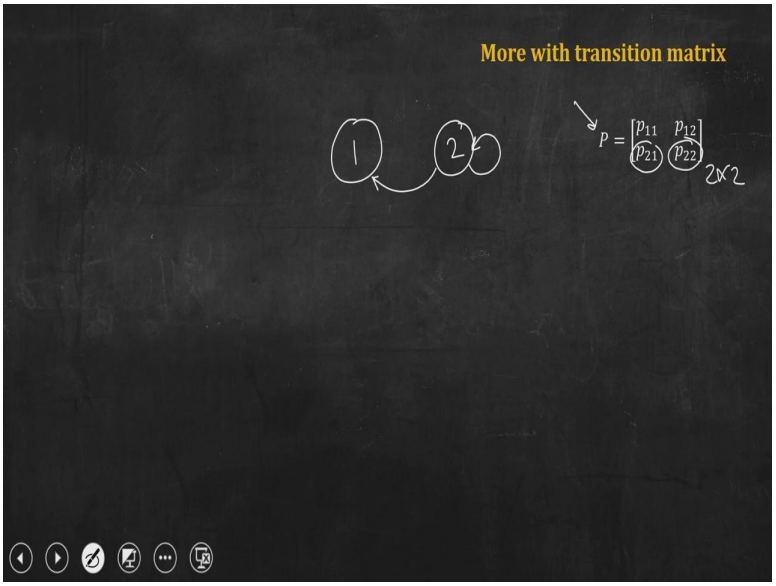
Now, before I move into that hardest definition of this diffusion distance, let us look into some properties of this problem rematch metrics, the transition probability matrix or hopping probability matrix, these are probability matrix.

(Refer Slide Time: 13:17)



So all these entries are from 0 to 1 and all positive, and we can easily check out that one of the Eigen value of these metrics will be always 1 and rest of the eigenvalue of this matrix will be less equal to 1. This property is very useful, and we will use it here today in subsequently.

(Refer Slide Time: 13:52)

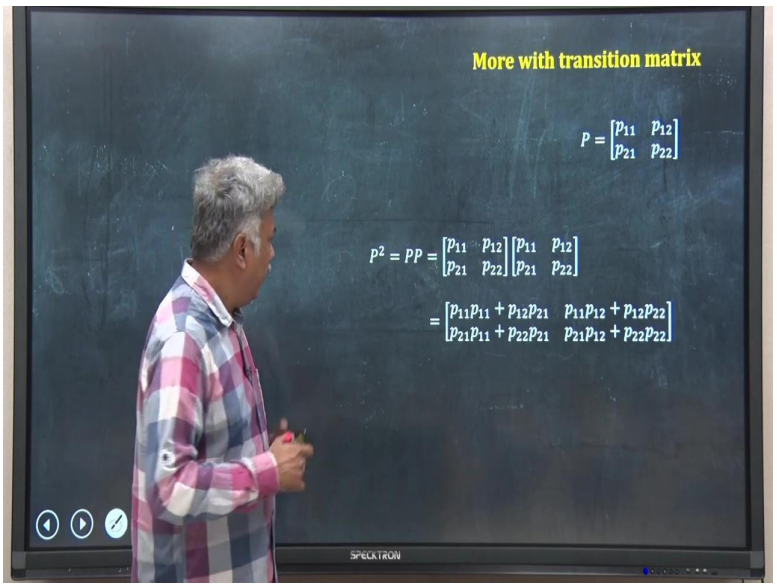


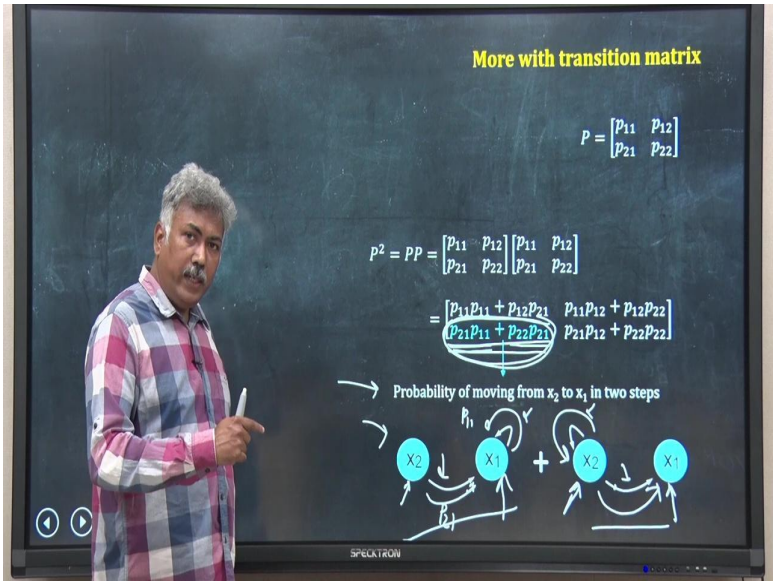
P =

$$\begin{vmatrix} p11 & p12 \\ p22 & p22 \end{vmatrix}$$

Let me play with this matrix A bit more. Suppose I have just two data points 1 and 2. So, then the hopping probability matrix a transition probability matrix will look like this, it will be 2 by 2. So, P_2_1 is the probability of going from 2 to 1, whereas P_2_2 is the probability of staying at 2 it is not going anywhere it is staying back into now, what I do I get the P square that means I multiply P by P and I get another matrix it is simple matrix multiplication you can try yourself.

(Refer Slide Time: 14:28)





$$P^2 = PP =$$

$$\begin{vmatrix} p11 & p12 \\ p22 & p22 \end{vmatrix}$$

X

$$\begin{vmatrix} p11 & p12 \\ p22 & p22 \end{vmatrix}$$

=

$$\begin{vmatrix} p11*p11 + p12*p21 & p11*p12 + p12*p22 \\ p21*p11 + p22*p21 & p21*p12 + p22*p22 \end{vmatrix}$$

Now, what is the meaning of each of the entries each of the elements of this a new matrix P square, let me check. Let me check with this one. If you carefully look into it, the first step is

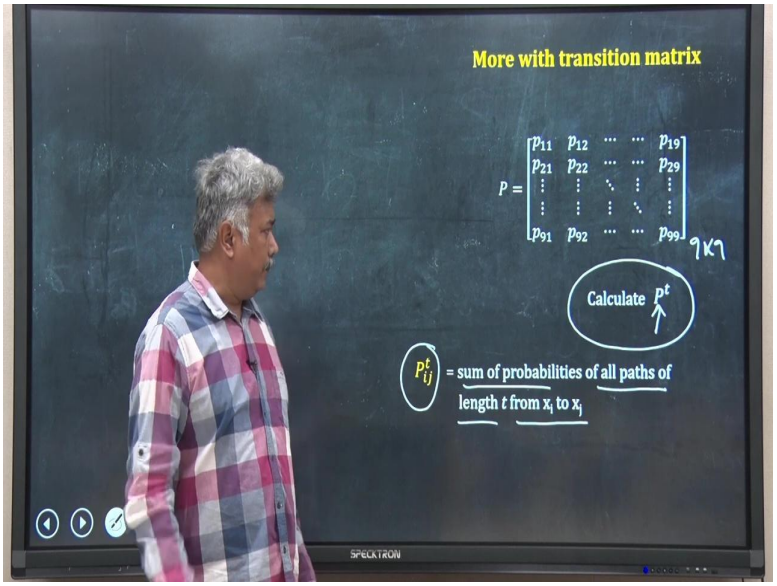
P_2_1 into P_1_1, what does that mean? So I am right now and data to X2, I am moving to X1. So, this is the probability P_2_1 and then I hop from X1 to X1 that means the next time step I do not go anywhere I stay there.

So, that is P_1_1 that means these P_2_1 into P_1_1, this is the probability of doing these hopping, I start from X2 go to X1 and then stay in X1. Now, let us look into the second thing P_2_2 and into P_2_1 that means, in the first step, I am staying at that point 2. And in the next step, I am moving to X1. So, in both cases, you can see that you started or I started at X2 and eventually landed in X1, I started at X2 and eventually landed in X1 and this one and this one these two path these two options are mutually exclusive.

So, their probabilities are summed here. So, what does this summation represent? It represents the probability of going from data point to X2 to data point X1 that is the first data point using two steps right here I have first step, second step, first step, second step. So, this is this element here in the P square matrix is giving me the total probability of reaching to the first data point from the second data point in two steps of diffusion or hopping.

So, in this way, rest of the other elements you can explain they are all representing path of size 2 where two steps has been taken and the total probability is calculated from reaching from one data point to another data point.

(Refer Slide Time: 16:57)

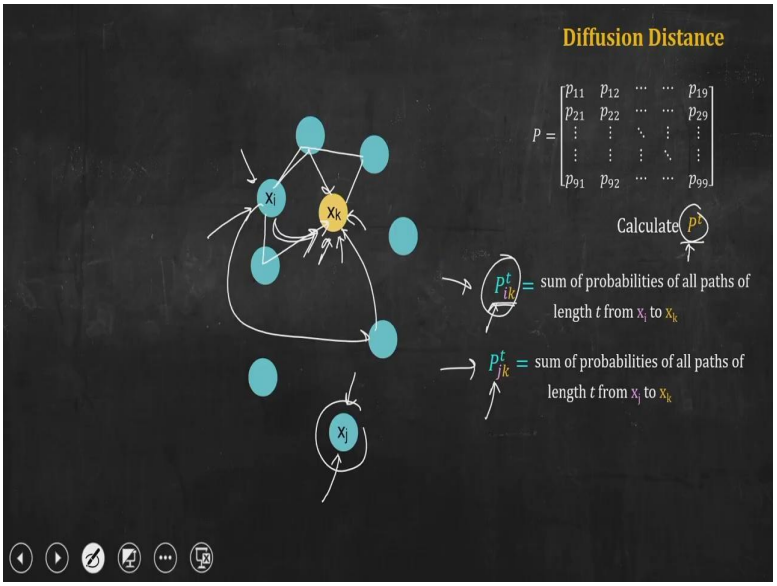


So, in this way, if I have supposed for my original line and data point I have 9 by 9 matrix, I can calculate P to the power T here T is the number of time steps are number of hops. So, then what will happen each of these elements of this PT each of these elements are P to the power T will represent something. What it will represent?

For example, the ij element of this P to the power T metrics will give the sum of probabilities of all paths of length T from Xi to Xj that means, it will give me the total probability of going to j8 data point from i8 data point using T steps or T hops. So, in this way, originally I have the data set I calculate the Euclidean distance from that Euclidean distance using the Gaussian kernel that I have shown I calculate the P the transition probability matrix or hopping probability matrix.

Now, I what I do I raise this P to the power something for example, 1 2 or 3 depending upon my choice that is my decision as a user. So, I raise it to P to the power T and that matrix gives me the probabilities are moving or hopping from one point to another data point using T number of steps.

(Refer Slide Time: 18:28)



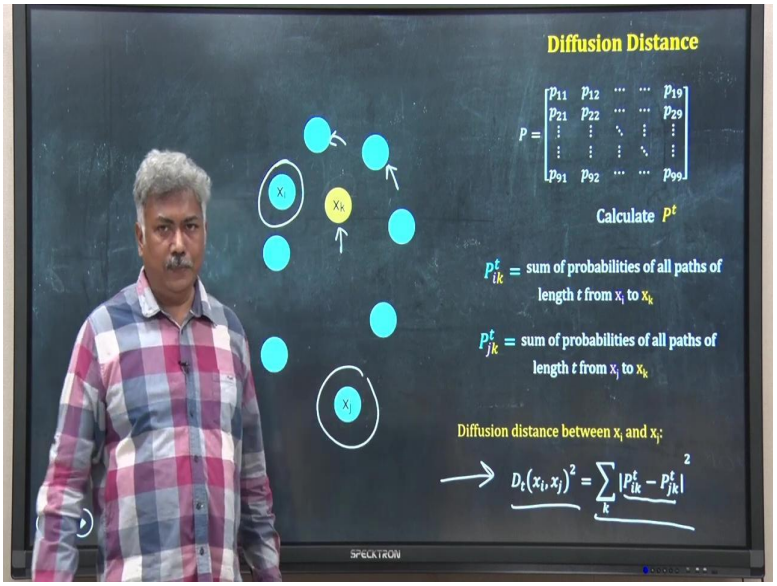
Let me move to now defining the diffusion distance using this P to the power T matrix. So, I have all the data points and I have calculated the P to the power T metrics. Now, I want to calculate the diffusion distance between the iX data point and the jX data point what I will do I will take another data points the third data points that kth Xk.

Now, I find out all possible paths of going to excite to Xi to Xk data to kth data using T steps and get calculate the probability and sum them. So, that probability can be obtained from these metrics. So, this P sub I K P to the power T sub I K is the probability total probability of going from Xi to Xk in T steps following all possible path, because you can see I can suppose I want to go into two steps to escape I can go like this, then this now it will become three actually.

So, I can take this one and this one this is two steps. I can go from here to here two steps, I can go here and then to two steps. So, I calculate the probability of all those possibilities. So, P to the power T sub ik is a probability of going from i-th to data point to the kth data point using T steps using all the paths possible. Similarly, this P sub jk is the probability of going from Xj data point to Xk data point in T steps, following all the possible path.

Now, imagine this if Xj and Xi these two data point, or equivalently, close to this K-th data point, then this one and these two probabilities should be close to each other. So, if I subtract them, I should get a value of 0 or close to 0. And that is what we do, when you define diffusion distance, what do I do?

(Refer Slide Time: 20:52)

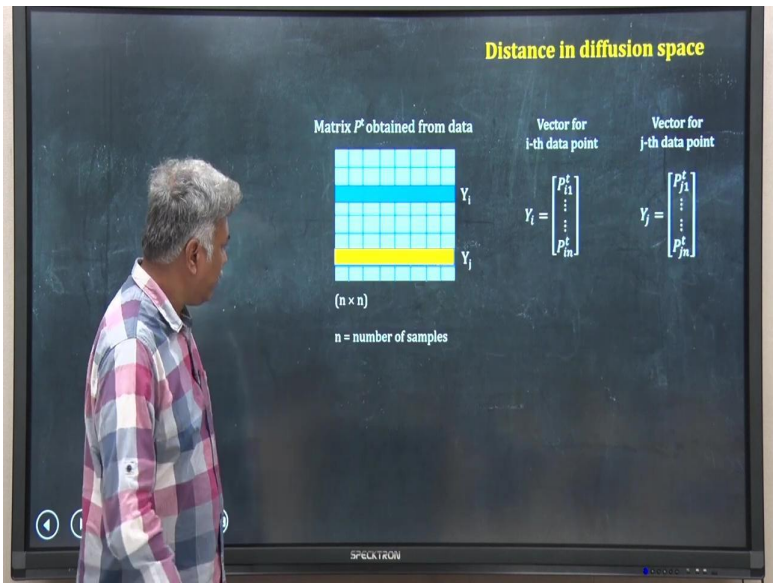


$$D_t(x_i, x_j)^2 = \sum_k \left| P^t_{ik} - P^t_{jk} \right|$$

I calculate this Ptik, Ptjk and subtract them and I square it and then I have initially taken this one k-th data point. Now, I take this one k-th data point, I calculate the same thing, I again take this one as the k-th data point in this way, I take all other data points as the intermediate data point and do the calculation and I sum them all.

So, now, you can imagine if the j-th data point this one and the i-th data point is locally close in the overall structure of the data, then what will happen this distance will become close to zero. So, this is the definition of diffusion distance. Now, I want to use this diffusion distance to embed my high dimensional data to lower damage. Now, interestingly, I actually do not need to calculate this diffusion distance.

(Refer Slide Time: 21:59)



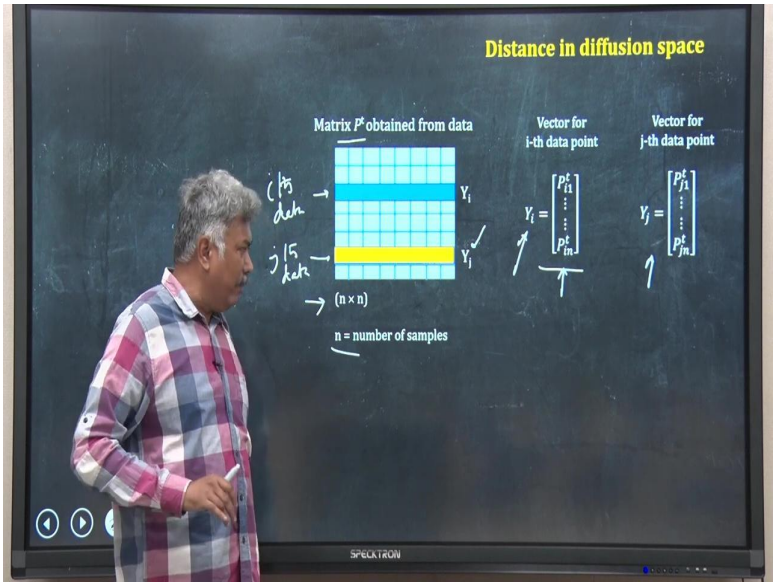
$$Y_i = \begin{vmatrix} P^t_{i1} \\ \vdots \\ P^t_{in} \end{vmatrix}$$

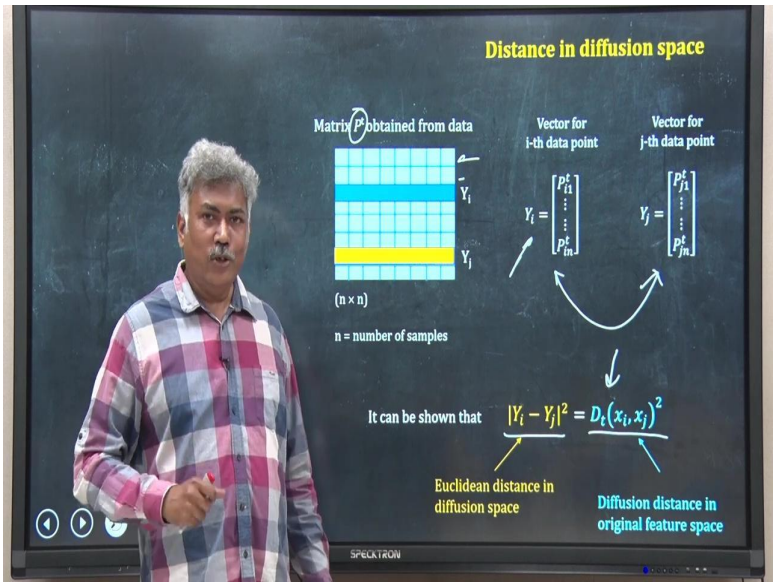
$$Y_j = \begin{vmatrix} P^t_{j1} \\ \vdots \\ P^t_{jn} \end{vmatrix}$$

I will show you. So, I have the data in higher dimension, I calculated the P to the power T matrix, which is again it says the probability matrix probably transition a transition probability matrix and I have N number of samples. So, it is a N by N matrix square matrix and this row this is for the i-th data or i-th sample whatever you say this is for the j-th data.

So, you take this i-th row of this matrix inverted transpose it and you get a vector Yi and this vector will represent my i-th data point in some way. Similarly, for the j-th one, I have Yj Vector, which has nothing but this row in this matrix and they have transpose it and made it vertical. So, row vector to column vector. So, I have now two column vector Yi and Yj corresponding to the i-th data point and the j-th data point. Now,

(Refer Slide Time: 23:09)





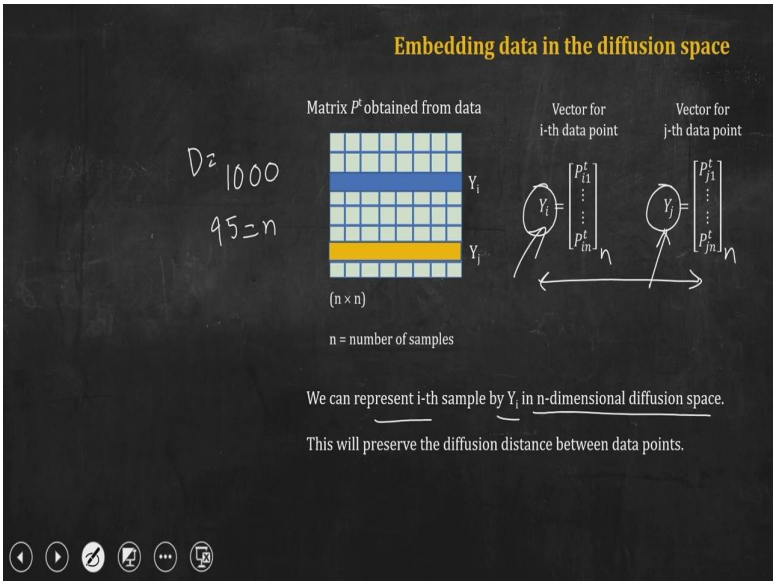
$$\left|Y_i - Y_j\right| = D_t\left(x_i, x_j\right)^2$$

I will skip all the mathematical derivation all this thing interestingly, it can be shown that if I calculate the Euclidean distance between these two vectors, that is equal to the diffusion distance. So, that is written here it can be shown that the Euclidean distance square is equal to the diffusion distance squared.

So, that means, I do not need to calculate diffusion distance from the data I will calculate P from P, I will calculate P to the power T, T is defined by me as a user I can keep it one also. So, and then from that P to the power T, I can calculate these vectors very easily because each

of these rows are those vectors and the Euclidean distance between these vector I am saying is equal to the diffusion distance.

(Refer Slide Time: 24:01)



So, what I can do now, see, I want to use diffusion distance in the original data space higher dimensional data space as a measure of similarity when I reduce the dimension and now, what I have got, I have shown that the Euclidean distance between these two Vector can actually represent my diffusion distance.

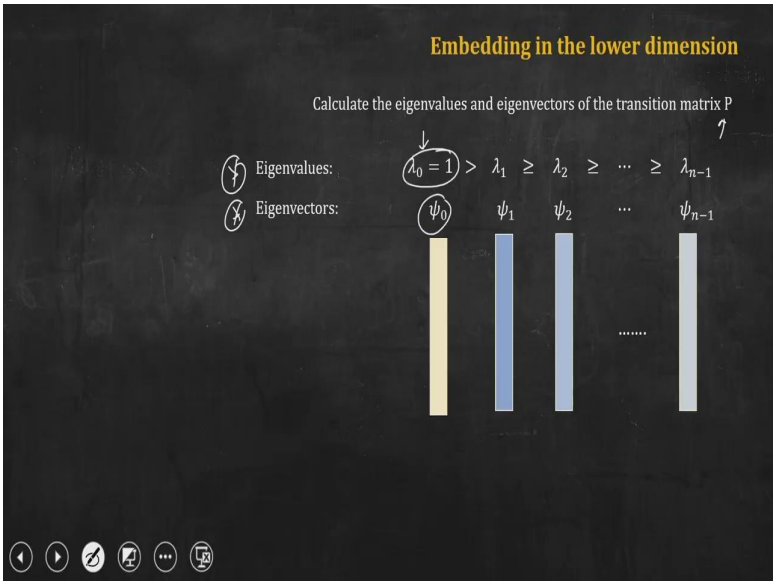
So, that means, I do not need the original data representation each of these data point I can represent by these vectors for example, i-th data point can be represented by Yi vector and Yj vector for j-th data point and what is the dimension of this dimension is N this dimension is N sample size.

So, now in this N dimensional space, each of these vectors is now representing my original data points, but still retaining the diffusion distance because the Euclidean distance between Yi and Yj is equal to their diffusion distance. So, what I can do, I can represent the i-th sample by Yi in the N dimensional diffusion space and it will still retain the diffusion distance from other data points.

So, have I reached some sort of dimension reduction I have received which I have done that for example, original data suppose you have 1000 genes you have measured in 45 samples. So, 45 is equal to N original dimension was 1000. Now, you are representing by vector of size N that means 45 So, from 1000 You have reduced the dimension to 45.

But you must be wondering that still 45 Dimension Data How can I visualize it is really not dimension reduction, I want it to two dimension or three dimension. Yes, I will proceed to that. And that is what to achieve further reduction I will take help of Eigen values and Eigen vectors.

(Refer Slide Time: 26:08)

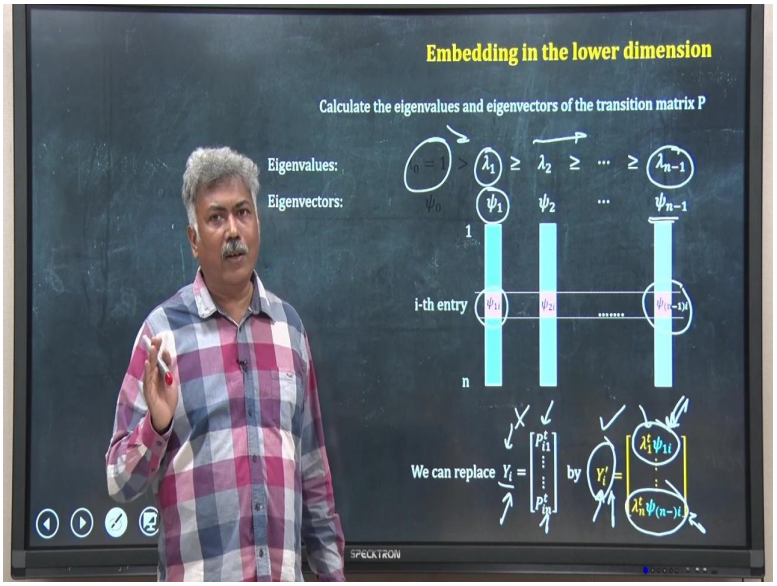


$$\lambda_o = 1 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n-1}$$

Now, if you remember I said these probability transition matrix has 1 Eigen value equal to 1 and rest of the Eigen values are less equal to 1. So, what I do I calculate the P from data and then I calculate the Eigen values and eigenvectors of that matrix and I arranged them in a descending order. So, this is the first Eigen value the highest value 1 and rest of them R the other Eigen values and the corresponding Eigen vector as shown by these bars.

Now, you can check out I am not going details, the Eigen vector for this first Eigen value highest Eigen value, all the entries here in this vector will be seen they will be constant. So, that does not help us. So, we remove that now, we are left with n minus one Eigen vectors and Eigen values. Now, again I will not go to details on mathematical derivation,

(Refer Slide Time: 27:15)



$$Y_i^t = \begin{vmatrix} \lambda_1^t \psi_{1i} \\ \vdots \\ \vdots \\ \lambda_n^t \psi_{(n-1)i} \end{vmatrix}$$

You have to take my word one interesting thing can be shown what we can show that we can actually replace this Yi vector, which I just discussed few slides back by a new vector Yi prime, but still Yi prime will retain that property of that diffusion distance that we have shown for Yi. Now, what do I have in Yi prime?

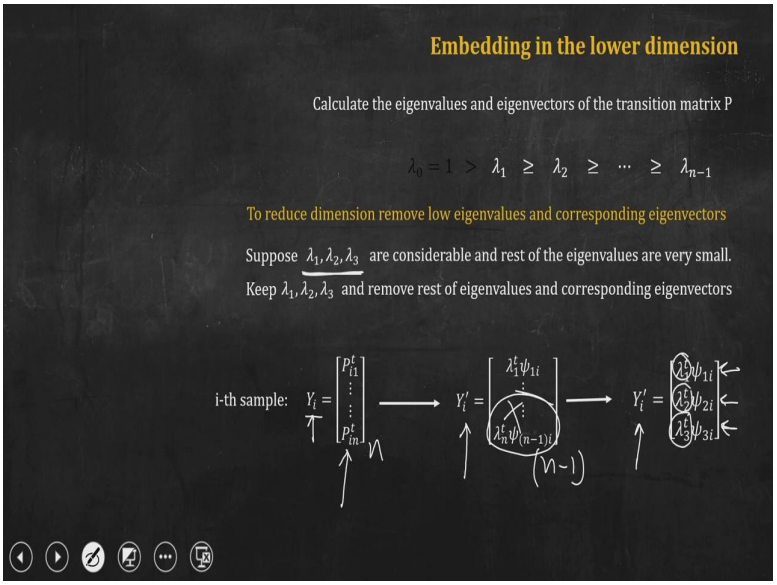
Why am I writing it separately look into the entries of Yi prime is why I have the probabilities coming from the probability transition probability matrix. Whereas, in Yi prime you have entries coming from the Eigen values and Eigen vector let me check the first entry for Yi prime this is multiplication of we have taken these lambda one the second highest Eigen value raised it to the power T and then multiplied with the value of i-th value in the psi 1 eigenvector.

Similarly, what is the last entry we have taken the i-th entry in the last Eigen vector and multiplied it with lambda corresponding lambda Eigen value up there to the power T. So, in this way I can replace this one by this one, but how that will help us I want to reduce the dimension I want to reduce the dimension from N to further low some amount of diamond

region reduction has happened because this is N and this is in minus 1 just 1 that will not here that will not make you happy you have to go further low.

Now, remember these Eigen values here highest one was 1 and I have checked it off, rest of them are smaller. So, that means there are fractions. So, it as I go down their values become smaller- smaller and smaller. So, that mean I can actually reject most of these lower value Eigen values, I do not have to consider those Eigen vector Eigen values at all.

(Refer Slide Time: 29:30)



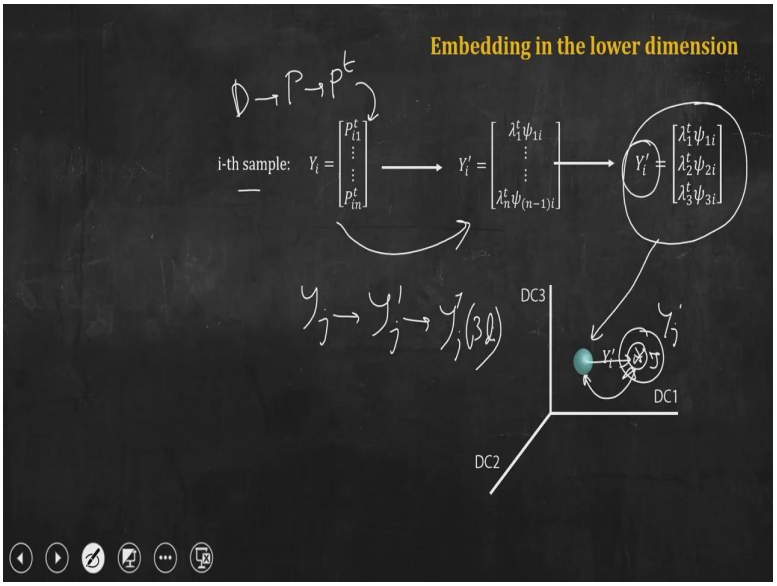
Let me give you an example. Suppose for a particular data set we calculated the P matrix and we saw that these Eigen values first three Eigen values 123 Not the first second, third and fourth, but I am writing it 123 Here, because the first one was lambda zero they have considerable values when they are close to one that was 0.9, 0.8 0.6 something like that and the rest of the other Eigen values may be very small, very small fractions.

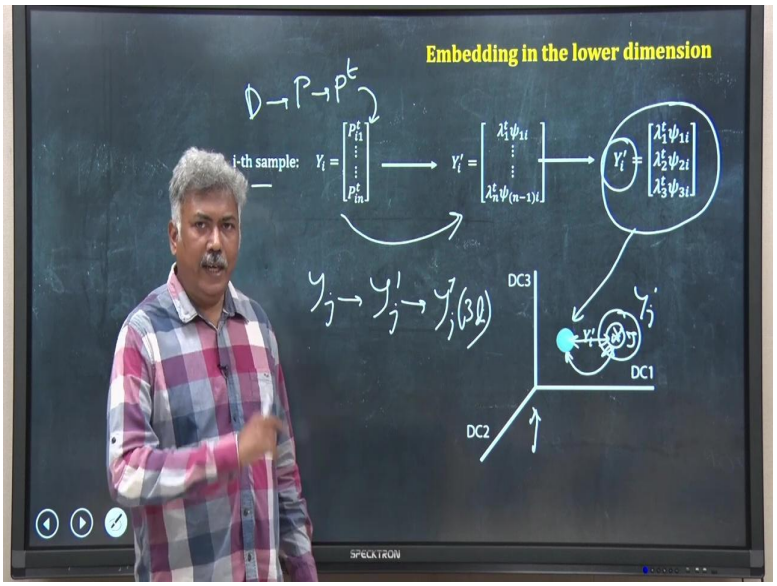
So, what I can do I can check off I can remove those Eigen values and corresponding Eigen vector, because they will not contribute much in my analysis. So, then what I get? I said originally I will embed the data using Yi vector then I said that is equivalent to using Yi prime vector. So, this is N dimensional, this is N minus 1 dimensional.

Now, I am saying get rid of this lower Eigen value thing get rid of them and only keep these three lambda 1 lambda 2 lambda 3 that means, I get a reduced Y prime, which has only three entries corresponding to 3 considerable lambdas Eigen values. So, what is the dimension of

this vector 3. So, that means, the i-th data point can be represented in state three dimensions using these vector.

(Refer Slide Time: 31:05)





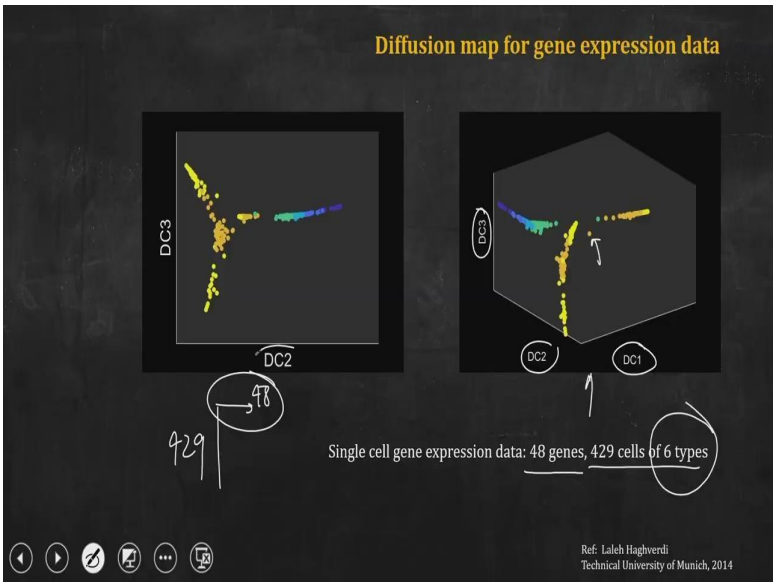
So, that is what I will do, I start with our original data with D dimension from that I calculate P I do not calculate diffusion distance because I do not need to explicitly calculate that, from that P I calculate P to the power T, T I can vary from one to something from that I calculate Yi for each of these data points i-th data points or i-th sample and that can be represented by y the in terms of Eigen values Eigen vector of P matrix.

And I can get rid of all the low Eigen values and the corresponding Eigen vector and eventually, I get I have here in this example, I have kept three Eigen values and the corresponding Eigen vector. So, I have a reduced presentation of i-th sample, i-th data point using that Yi prime vector. So, in three dimension it will represent this point if I do similarly

for Yj from that I calculate Yj prime from that again I calculate Yj prime in three dimension then may be that point is here. What can we mathematically shown and I have skipped.

But I have just intuitively explained that that a diffusion distance between these j-th data point this is Yj prime this diffusion distance between these two data point in the original dimension is retained here is represented here by the Euclidean distance between these two data points in this three dimension. That is how we reduce the dimension of a higher dimensional data to lower dimension using the algorithm of diffusion maps.

(Refer Slide Time: 33:06)



Before I end this lecture, let me give you an example from biologic diffusion maps are now becoming very useful to reduce the dimension of data, where you have you know, single cell experiment for gene expression or something like that flow cytometry, particularly, you will find nice work where people have tried to understand the differentiation of sale trajectory of cell as they change their phenotype using the single cell gene expression data, or flow cytometry data, and then reduce the dimension using diffusion map and visualize them in two dimension or three dimension.

And then to find out the trajectory of cell as they change their phenotype in this reduced space. The example that I am showing here, in this case, we have 429 cells of 6 different types of cells, and they have measured only 48 genes in this case 48 genes in those cell, because those 48 gene can characterize some differentiation property of the cells, and then what we I did I reduce the dimension of this dataset, so you can say it is a 48 versus 429.

So, it is a 48 dimensional data, I reduce it to two and three dimension. So in three dimension, these are corresponding to three main Eigen vectors after a chapter the first one, and each of these data points is a cell and they are color coded based on the 6 types that they have assigned. And in three dimension, I can see it in this way. in two dimension what I have done, I have considered DC 2, and DC 3, and you can see nicely, the different cell types of got segregated.

Now you can look into the biology of these cells and why they are different, and how they are, gene expression affects their phenotype. All these things we can study. So that is all for this lecture. In this lecture, we have learned diffusion map, which is a nonlinear technique of dimension reduction for high dimensional data. Thank you for learning with me today.