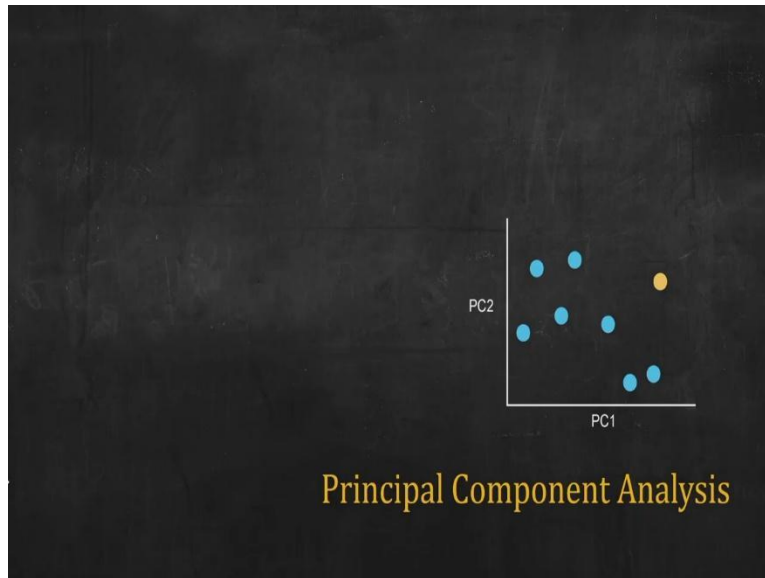


Data Analysis for Biologists
Professor Biplap Bose
Department of Biosciences and Bioengineering
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology, Guwahati
Lecture 45
Principal Component Analysis

(Refer Slide Time: 0:29)



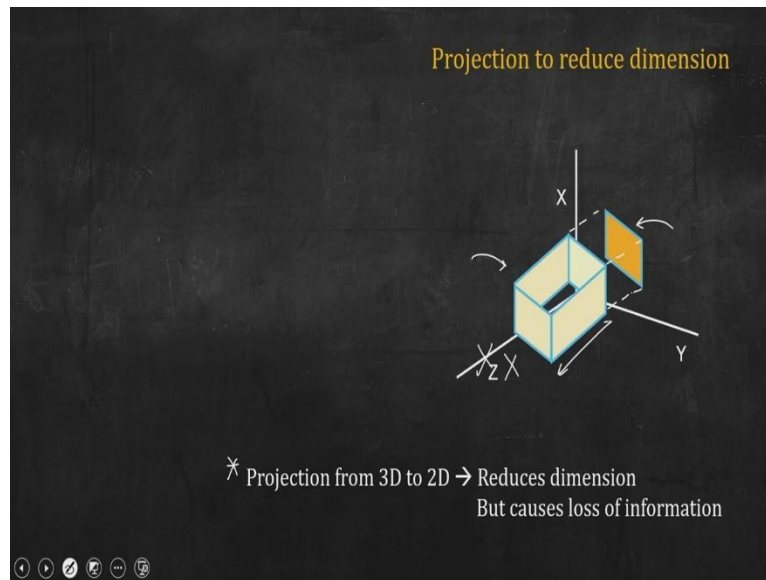
In the last lecture, we discussed about higher dimensional data in biology for example, a microarray data or RNA-seq data. And we discussed also the issue of analysis of such a higher dimensional data and visualization of that type of data. If you remember at that time, we discussed that we have technique to reduce the dimension of data before visualization or analysis.

We also mentioned that one of the one of such techniques is principal component analysis or PCA. I hope you have seen that video. If not be please pause this one and go back to that video. Because, in this particular video, I will discuss about the logic and the mathematics behind PCA. And subsequent one I will show how you can use R functions to perform PCA on a data set.

So, if you remember, what we want to do in dimension reduction is that we want to take a higher dimensional data, suppose which may have hundreds of dimension and then we want to project it on lower dimension in such a way that we retain the relationship between the data the trends and

the pattern within the data largely intact, that is the goal of principal component analysis. Now, before I move into the logic and the math behind PCA, let us see what do I mean by projection of a data and reduction in dimension.

(Refer Slide Time: 2:08)



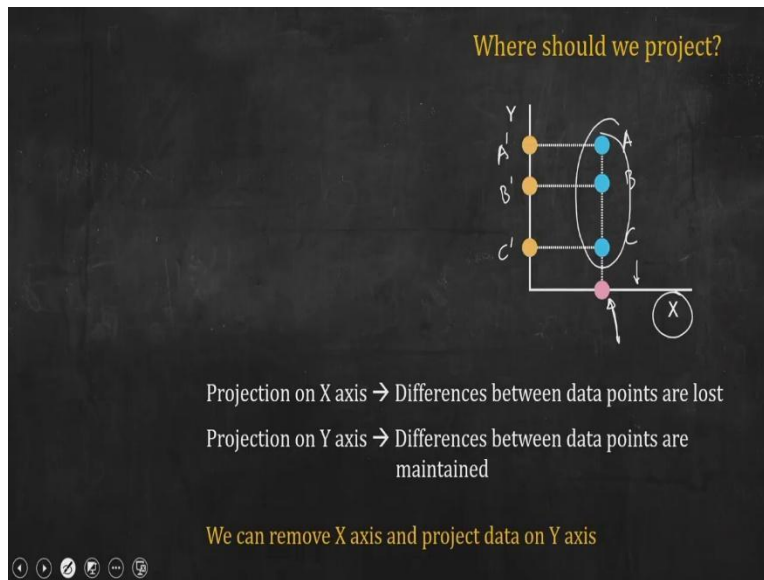
So, what I have here as an example is a three dimensional object, this is a 3D object a rectangular cuboid and it is resting in a 3D coordinate system X, Y and Z. Now, it has dimension equal to three, I want to reduce the dimension, I want to make it dimension three to dimension two that means, I have to discard either X or Y or Z axis. So, suppose I discard Z axis then what do I get?

If I have to still retain the information about this particular object in the XY plane, what I can do, I can take a projection of that object on the XY plane and if I do the projection of this object on XY plane and discard Z axis, I will get a simple rectangle. So, my previous data was a three dimensional object. Now, I have discarded one dimension that is the Z axis and I have projected the object on two dimension.

So, I have reduced the number of dimension and I have got what, I have got a projected data which looks like a rectangle. Now, you must have noticed one important thing what I have written here that whenever I project from 3D to 2D that means, I am reducing the dimension, I am actually losing some information. Obviously, in this example, we have lost the information in this direction, we lost the information about the width of that object.

So, whenever you will do some sort of projection to reduce the dimension of your data, there will be some sort of loss of information. So, that means, I have to judiciously reduce the dimension. So, I have to use a procedure by which I have least loss of information about my data set. So, let us take an example to understand what do I mean that judiciously discarding one dimension, maintaining most of the information, a simple example.

(Refer Slide Time: 4:16)



I have three data point, a two dimensional data, data one, data point one, two, and three. I want to convert these from 2D to 1d. One way of doing that is I can project these datas on either on the X axis or on the Y axis and then cancel one of the axes. So, how can I do that? If I project these three data, let us mark it as A B and C on X axis, I will get this pink dot note one important thing. All these three data are now collapsed at one single data point that is this pink one.

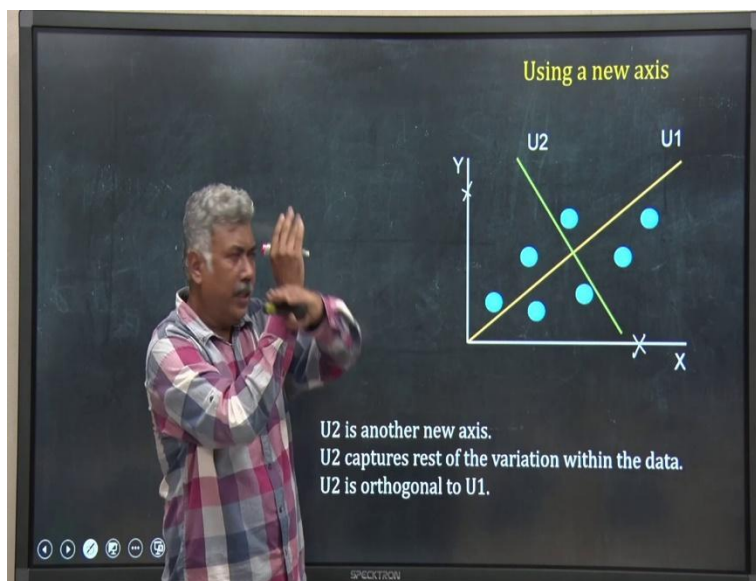
So, if I removed the Y axis, just see this projected data, I will not be able to know that there are actually three data points, because I will see only 1 data point the pink one. Now, look what is happening if I discard X and project the data only on Y. So, then what will happen for each A,B,C I will get a projected position A prime, B prime and C prime and they are distinctly different.

So, even now, even if I discard X axis, I will have only 1 axis Y so, it is one dimensional data, but still I can I will be able to differentiate between three data points and I will know they are

separated. So, as I said I have to judiciously discard dimensions. So, in this case, I will not discard Y, I will remove X and project that data for A, B, C on Y axis. So, I will do a data dimension reduction from two dimension to one dimension without losing much of the information.

So, now, let us look into it in a different perspective. In this example, I have kept the original axis and discarded one of the original out of x and y dimension I have removed the X coordinate X dimension. Now, if I do the same procedure, but not on the existing coordinates, rather I create new axes, new coordinate system completely. Let me give you an example, to explain what we mean by that.

(Refer Slide Time: 6:37)



This is my data, I have six data point, it is a two dimensional data, its original coordinates are X axis horizontal one and the vertical one I named as Y. Now, if you look into this data, you can see that along this direction the data has a spread. So, what do I do, I create a coordinate axis along that direction, which mean the direction where I have the highest variation that is dispersal in the data point.

So, suppose I name that u1, this yellow line is that coordinate. So, along this coordinate most of my my data has the highest dispersal, you can easily visually see that and I can actually do the projection of the data on this axis. So these will be the projections coming like this. Now, if I forget about all other original coordinates, that is Y and X, then still if I project this data on this

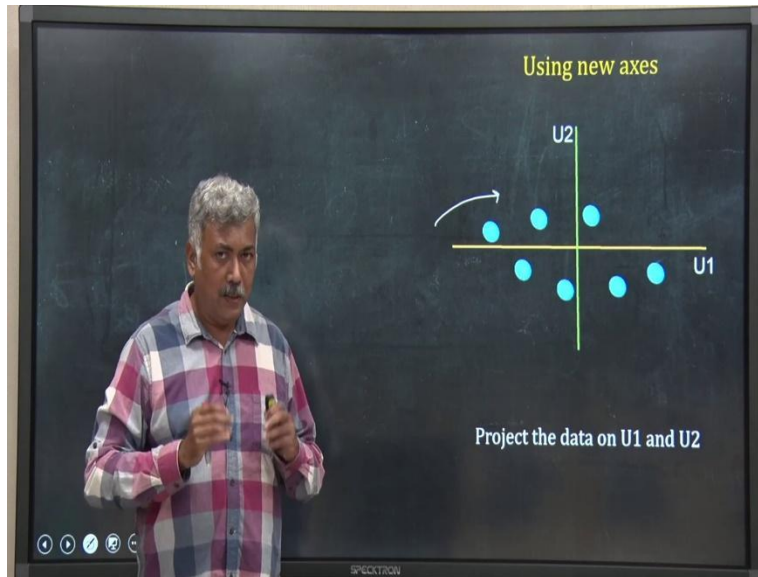
u_1 , I will still be able to differentiate between these six data points, because they are separated they are dispersed.

And I have chosen that direction, the direction of u_1 in such a way that the dispersion is maximum there. Now, if you carefully look into this data, you can still see there is some dispersal also in this direction. Now that direction's dispersal, the variation of data in that direction is not captured by the projection along u_1 . So what can I do?

What I will do here, I will create another new axis, another new coordinate axis, or dimension, which will be orthogonal to this yellow one u_1 and I will call that u_2 . So that is what I have done. So now I have two new axis u_1 and u_2 or rather, you can say I have new two dimensions. And I can now discard these X and Y.

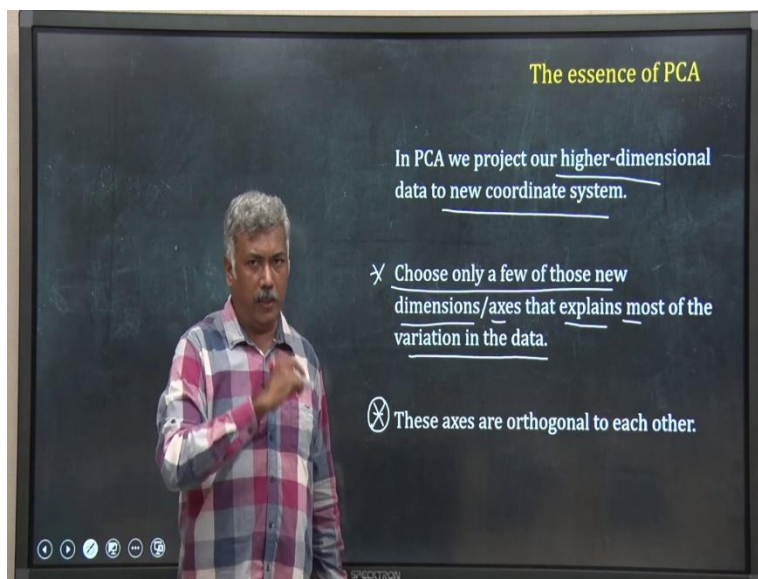
Because if I project the data on these u_1 and u_2 , they will be able to capture the difference between different data. So if I discard X and Y, then I do not need to keep u_1 and u_2 like this, I can rotate it. So that is what I have done here. I have rotated u_1 and u_2 , discarded X and Y.

(Refer Slide Time: 9:02)



So now I have my data on a new coordinate system u_1 versus u_2 and these two coordinate captures the diversity, the dispersion of my data adequately. So these two examples, for the first one where I have all data point in collapsing on X axis in one point, that is why I discarded X. And in this case where I have created completely two new coordinate these two examples brings us to the essence of PCA principal component analysis. What is that?

(Refer Slide Time: 9:42)



So PCA is a dimension reduction technique where we are projecting our original higher dimensional data to a completely new coordinate system. You have new axes and these axes are orthogonal to each other. So, if I have a D dimensional data, five dimensional data six dimensional data like that, then I can create this D number of new dimensions.

Now, obviously, I will not choose all those D dimension, rather what I will do, I will pick few of those new dimension, which can capture the most of the variation in the data, as I have written here, choose only a few of those new dimensions or axis that explains most of the variation in the data that is the whole goal of principal component analysis. This is how we reduce the dimension of a data set. Now, let us put these words into mathematical equations and see how the mathematics of PCA works.

(Refer Slide Time: 10:51)

Understanding the Math of PCA

Data: $X = \begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1d} \\ \vdots & & \vdots & & \vdots \\ x_{i1} & \dots & x_{ij} & \dots & x_{id} \\ \vdots & & \vdots & & \vdots \\ x_{n1} & \dots & x_{nj} & \dots & x_{nd} \end{bmatrix}$

Project the data (X) on a vector (u) such that:

- a) The variance of the projected data is maximum.
- b) u is a unit vector

So, I have a data set this is represented by this matrix X and this n into d matrix, note carefully I have d variables which are in this direction. So, the columns are variables in this X matrix, whereas I have n number of samples, so, samples are the rows. So, this is my data matrix X . And what I want to do?

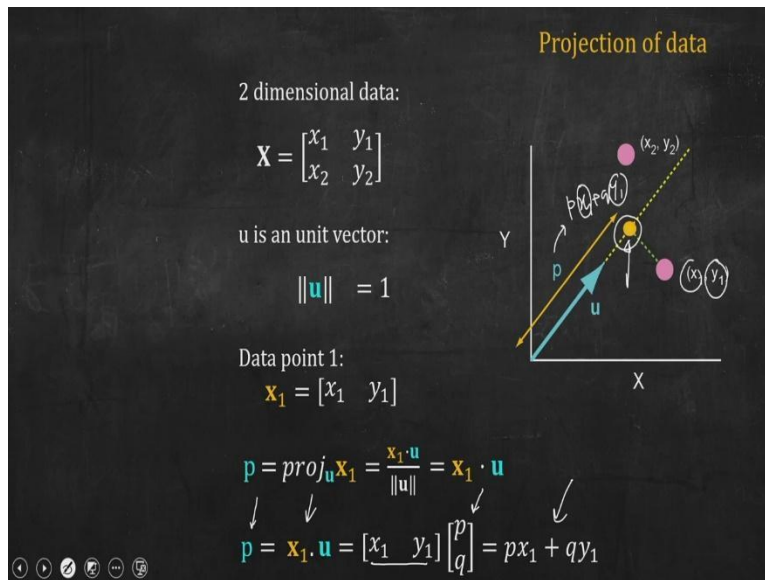
I want to create a new coordinate, new coordinate axis new dimension. So, if I have a coordinate something like this X and Y the way we originally write usually most of the time. You can always say that this X is actually a vector, is it not? Vector and Y is also a vector. So, when I say

I want to create a new axis or new dimension, essentially, I am looking for a new vector, what type of vector?

I want a vector which will be unit vector and let me call it u such that, I want to project my data on that vector u . And on projection, the variance of the projected data will be maximum. Let me repeat, I want to get a new vector u and project my data X on that new vector such that this new vector that I have chosen is a unit vector and the variance of the projected data along this u vector is maximum.

That is what I want to do mathematically in PCA. Now, in this formulation, there are two key points, one is the variance the other one is projected data. So, let me explain both of them first, before we go into details of further steps of PCA.

(Refer Slide Time: 12:54)



2 dimensional data:

$$\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}$$

$$X =$$

Unit vector u :

$$\|u\| = 1$$

Data Point 1:

$$x_1 = [x_1 \quad y_1]$$

$$p = \text{proj}_u x_1 = \frac{x_1 \cdot u}{\|u\|} = x_1 \cdot u$$

$$\begin{vmatrix} p \\ q \end{vmatrix}$$

$$p = x_1 \cdot u = [x_1 \quad y_1] = px_1 + qy_1$$

What is projection of a data on a vector, that I will explain suppose, in case of n by d matrix, I have a two by two matrix for X. So, I have two samples these two row and I have two variables x and y. So, I can show this, this is X axis, this is Y axis and I have two samples. So, I have the sample 1, this is sample 2, these pink points.

Now, I have a unit vector right now, I do not know details of that, but imagine I have a unit vector u, this will be eventually my new coordinate axis, this in blue colored arrow is a unit vector u and the yellow one is the span of that. So, I will project my data, these two data points on this vector u or on the span of this vector u.

So, to do that, let me explain with the first data point this one the first data point this one which is x1 and y1. That means, I want to get a projected data this yellow point on the span of vector u. How can I get that? I want to project this data point 1 on this vector a unit vector u. How can I get that?

I hope you remember our last lecture where we have discussed about vectors and also discuss our projection of a vector onto another vector. So, I can represent this data point, this first data point this pink dot as a vector, that vector is x1, its components are x1 and y1. So, I want to project this x1 vector on this u and how can I do that? I know the formula.

The projection of x1 on u will be given by the dot product between x1 and u these two vector divided by the magnitude of u. Now, u is a unit vector that is how we have defined that means that the magnitude of a given unit vector u is 1. So, what I get? I get x1 dot u dot product of x1 the data point vector and the unit vector and what it will give me?

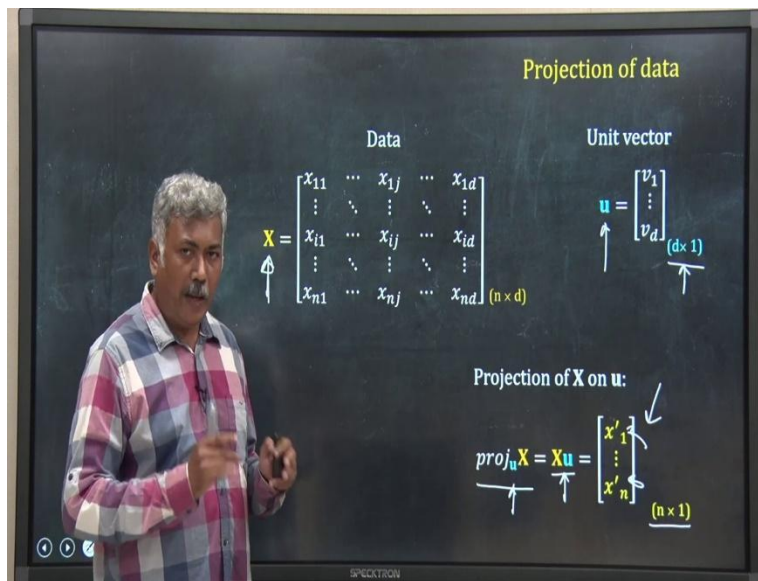
It will give me the position of this projected point on this unit vector u , and this length, this is that position from the 0, 0 position. So, that is the p . So that p , I get by this formula of projection. Now, let me imagine, u is, you have some values suppose, u is p and q arbitrarily I have taken.

So, let me do this calculation dot product between u and x_1 . So, I have to calculate the position of this projected data, data 1 that will be equal to the dot product between x_1 and u . x_1 is this vector and u is this vector. So, if I do multiplication between these two vectors one is row vector and another is column vector, I get $p_1 x_1$ plus $q y_1$. Notice one important thing this p is now p_1 , $p x_1$ plus $q y_1$.

So, that means, the position on the span of this unit vector for this data point in the first data point includes the information on the original values for the data point. See it includes the value of y_1 and x_1 . So, this position is a linear combination of the original data that is the essence, one important issue in PCA, in PCA, when you project in new coordinate that position of the projected data retains the linear information linear combination of the original data.

So, PCA is a sort of linear transformation. So, now, I have explained this the projection of a data point on a unit vector using a two by two matrix, but our data matrix is not two by two is a quite big one, n into d .

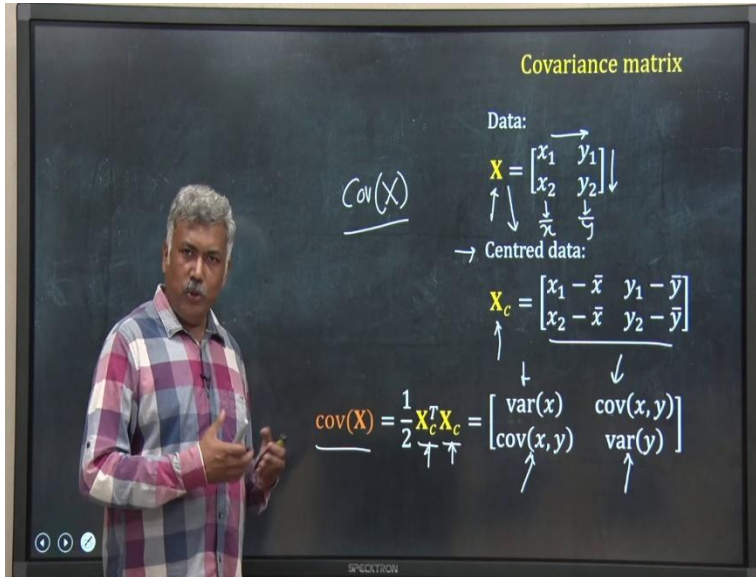
(Refer Slide Time: 17:34)



So, this is my data matrix X . This is suppose a unit vector whose dimension should be d into one otherwise I cant multiply. Now, if I project this data X on u using the formula we just derived in last slide, I will get the projection of X onto u will be multiplication of X and u and that should give me a column vector of dimension n into 1, n by one and each of these prime values are the projected values of my samples on that particular vector u or on that particular new coordinate.

So, that is for projection of data. As I said the other important issue in PCA is the issue of variance. For that we have to understand something called covariance matrix.

(Refer Slide Time: 18:30)



Data:

$$\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}$$

$$X =$$

Centred Data:

$$\begin{vmatrix} x_1 - \bar{x} & y_1 - \bar{y} \\ x_2 - \bar{x} & y_2 - \bar{y} \end{vmatrix}$$

$$X_c =$$

$$\begin{vmatrix} \text{var}(x) & \text{cov}(x,y) \\ \text{cov}(x,y) & \text{var}(y) \end{vmatrix}$$

$$\text{cov}(X) = \frac{1}{2} X_c^T X_c =$$

Again, I will take a two by two data matrix as an example. So, this is my data matrix I have two samples I have two variables. Variables are X and Y. I want to calculate the covariance of X remember X here is a matrix we have done covariance and variance earlier for random variables, but in those are scalar, in this case, X is a matrix. Using the same definition of variance and

covariance if I have to calculate the covariance of X , that is we have to calculate the covariance matrix of X what I have to do first?

I have to center this data. What do I mean by centered data? So, you calculate the mean of each of these variables, the columns. So, suppose the mean is \bar{X} and for the variable second variable it is \bar{Y} . So, then for the first column data for X variable, subtract the mean of that column and then for the second variable the second column subtract the mean of that column. So, I get a centered matrix X_c , X underscore c and this is the formulation of that.

So, now, if I have to calculate the covariance of X , covariance of X we can show that it will be it is a two by two I have two samples here. So, it will be one by two into the transpose of X_c multiplied by X_c . I have not gone into detail how we have reached there, but you can easily see it, check it by linear algebra and this multiplication will give me a matrix two by two matrix where the diagonal elements will be the variance of X and Y whereas, the off diagonal element will be the covariance.

(Refer Slide Time: 20:32)

Covariance matrix

Data

$$X = \begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1d} \\ \vdots & & \vdots & & \vdots \\ x_{i1} & \dots & x_{ij} & \dots & x_{id} \\ \vdots & & \vdots & & \vdots \\ x_{n1} & \dots & x_{nj} & \dots & x_{nd} \end{bmatrix}$$

$(n \times d)$

Covariance matrix $S = \text{cov}(X) = \frac{1}{n} X^T X$

Covariance matrix

Data

$$X = \begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1d} \\ \vdots & & \vdots & & \vdots \\ x_{i1} & \dots & x_{ij} & \dots & x_{id} \\ \vdots & & \vdots & & \vdots \\ x_{n1} & \dots & x_{nj} & \dots & x_{nd} \end{bmatrix}$$

$(n \times d)$

Covariance matrix $S = \text{cov}(X) = \frac{1}{n} X^T X$

③ All eigenvalues of S are non-negative

④ All eigenvectors of S are orthogonal

$$S = \text{cov}(X) = (1/n) X^T X$$

Again this is for a two dimensional system two by two data system whereas, my real data is n into d but we can actually use the same rule logic formulation to calculate the covariance for the data matrix also. So, my data matrix X has n samples and d variables and you can show that the covariance matrix for X which I will call now as capital S is equal to $1/n$ because I have n samples into X transpose X .

One important point here, this X here in this equation, in this equation are actually centered matrix, centered data, but I have not written the subscript c because, in general in PCA by default, we are supposed to use centered data. So, we are supposed to start with the data itself which is centered. So, most of the literature and textbook we simply do not use the subscript c anymore to reduce the equation.

So, what I have got is the covariance matrix S for X is equal to 1 by n into X transpose X. Now, let me go through some very crucial properties of this S covariance matrix which are useful in our PCA. The first one is that S will be a square matrix it will be d by d because I have d variables in my data set and it will be symmetric matrix that means, S is equal to S transpose.

The third important point is that all Eigen values, all Eigen values of S are non-negative that means, they can be zero or positive and the fourth one all the Eigen vectors of S are orthogonal, that means, they are independent to linearly independent to each other. If you remember, when I was discussing the essence of PCA, we say that, we want a new coordinate system where the axes will be orthogonal or linearly independent to each other. So, connect the dot what I am saying here is eigenvectors of S will be orthogonal, we will come back to later on.

(Refer Slide Time: 22:55)

Covariance matrix

Data

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{id} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nd} \end{bmatrix}$$

(n × d)

Covariance matrix $\mathbf{S} = \text{cov}(\mathbf{X}) = \frac{1}{n} \mathbf{X}^T \mathbf{X}$

If $n < d$:

⊗ At least one $\lambda = 0$

Then there are a few other important properties. If n is bigger than d that means, the number of sample is bigger than the dimension the number of variable in the data or all columns of X are

linearly independent, if these two properties are met, then all the Eigen values all the Eigen values will be positive.

And as these eigenvalues are distinct, I will have d number of Eigen vectors. So, if I have d number of variables, I will have d number of Eigen vectors. But, if n is not greater than d, it is less than d, when the number of sample is less than number of variables. In that case, one important issue is there that at least one of the Eigen values will be zero.

Now, that is enough for the basic concept of variance, covariance and the projection of a data on a vector. Now, let me get back to what we were doing earlier the mathematics of PCA.

(Refer Slide Time: 23:57)

PCA: an optimization problem

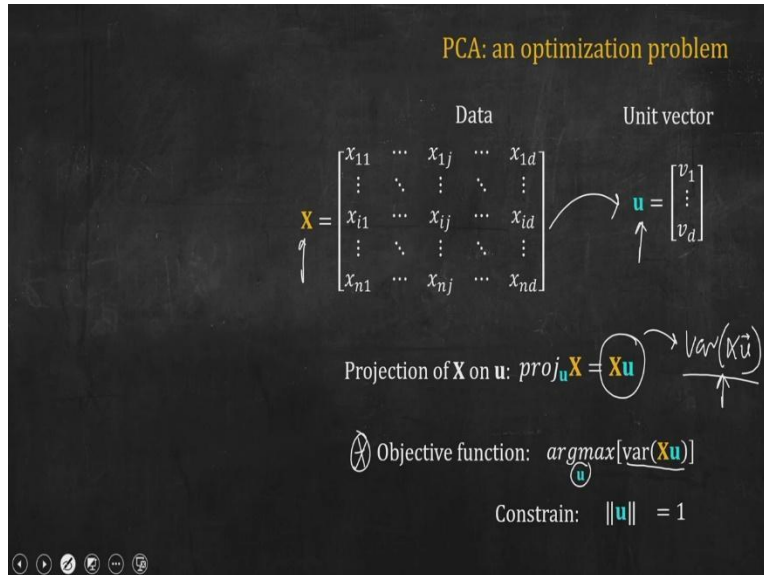
$\mathbf{X} = \begin{matrix} & \text{Data} & \\ \begin{matrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{id} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nd} \end{matrix} \end{matrix}$	$\mathbf{u} = \begin{bmatrix} v_1 \\ \vdots \\ v_d \end{bmatrix}$
--	---

$\text{var}(\mathbf{X}\vec{u}) = \vec{u}^T \mathbf{S} \vec{u}$

Projection of X on u: $\text{proj}_u \mathbf{X} = \mathbf{X}\mathbf{u}$

Objective function: $\underset{\mathbf{u}}{\text{argmax}} [\text{var}(\mathbf{X}\mathbf{u})] = \underset{\mathbf{u}}{\text{argmax}} [\mathbf{u}^T \mathbf{S} \mathbf{u}]$

Constrain: $\|\mathbf{u}\| = 1$



$$\text{Projection of X on u: } \text{proj}_{\mathbf{u}} \mathbf{X} = \mathbf{X}\mathbf{u}$$

$$\text{Objective function: } \underset{\mathbf{u}}{\text{argmax}} [\text{var}(\mathbf{X}\mathbf{u})] = \underset{\mathbf{u}}{\text{argmax}} [\mathbf{u}^T \mathbf{S} \mathbf{u}]$$

$$\text{Constrain: } \|\mathbf{u}\| = 1$$

So, what is our purpose what we are doing? I have n into d matrix X, which holds my data in sample d variables, I want to project the data X on a vector u such that that u is a unit vector and the variance of the projected data is maximum. That means, we want to maximize something is it not? So, I can say this is an optimization problem and let me write it in that form in optimization problem form.

So, X is given, u I do not know. I have to find the u. How I have to find that u? I have to find the u that will satisfy certain requirements, what is the requirement? So, if I project this data on u, the projected data will be Xu and its variance will be variance of Xu. So, I want to find a unit vector in such a way that this variance of Xu is maximum.

So, I am formulating an optimization problem where the objective function is to maximize variance of Xu by changing the u, selecting a suitable u with the constraint that u must be a unit vector the magnitude of u should be one.

Now, without going into details of the math people have shown a very nice property, this variance of Xu is actually equal to u transpose Su and here variance of Xu, u is a vector is equal

to u^T , transpose of u vector, S is the covariance matrix of the data X and u . So, I can reformulate my objective function, I have to find u such that $u^T S u$ is maximized with a constraint that u is a unit vector.

You can approach this maximization problem from different perspective, we will not go in details of that derivation and mathematical derivation I will say tell only that by using a common method called Lagrange multiplier, its very beautiful relations and a beautiful solution comes out of it.

(Refer Slide Time: 26:28)

PCA: an optimization problem

Objective function: $\underset{\mathbf{u}}{\operatorname{argmax}}[\operatorname{var}(\mathbf{X}\mathbf{u})] = \underset{\mathbf{u}}{\operatorname{argmax}}[\mathbf{u}^T \mathbf{S} \mathbf{u}]$

Constrain: $\|\mathbf{u}\| = 1$

S. Matrix \rightarrow Solution: $\mathbf{S}\mathbf{u} = \lambda\mathbf{u}$ \rightarrow Vector

λ : Eigenvalue of \mathbf{S} \leftarrow Scalar

\mathbf{u} : Eigenvector of \mathbf{S}

∇ Variance of the projected data would be maximum when the unit vector (\mathbf{u}) is an eigenvector of the covariance matrix (\mathbf{S}) of the data.

$$\mathbf{S}\mathbf{u} = \lambda\mathbf{u}$$

$\lambda \rightarrow$ eigenvalue of \mathbf{S}

$\mathbf{u} \rightarrow$ eigenvector of \mathbf{S}

The solution is very interesting, this is the solution, when you try to meet this objective function with this constraint, you reach this interesting solution pay attention to the solution. This is \mathbf{S} , this \mathbf{S} is covariance matrix of \mathbf{X} . So, this is a square matrix, \mathbf{u} is a vector and this lambda is actually scalar. Look at this relation \mathbf{S} into \mathbf{u} equal to lambda into \mathbf{u} .

Can you recognize this relationship? You must have already done that right? You must have recognized it. Yes, like this scalar quantity lambda is actually the Eigen value of \mathbf{S} , this is the Eigen value, Eigen vector definition. The square matrix into its Eigen vector is equal to Eigen value into the corresponding Eigen vector.

So, lambda is the Eigen value of \mathbf{S} , the covariance matrix of \mathbf{X} and \mathbf{u} is the Eigen vector. Remember, we are searching the \mathbf{u} . So, I have got the \mathbf{u} , \mathbf{u} is the Eigen vector of the covariance matrix of the data that is \mathbf{S} . So, let me make the statement I can say that the variance of the projected data would be maximum when that unit vector \mathbf{u} is an Eigen vector of the covariance matrix of the data.

So, we have found the unit vector which I can use as a coordinate a new coordinate on which I can project that data. If I project the data on this vector u which is the unit vector and is a Eigen vector of my covariance matrix for my data, then the variation the variance of the data along this axis along this vector will be maximum. Job done, but little bit still there that we have to sort out.

(Refer Slide Time: 28:37)

The principal components

$$S\mathbf{u} = \lambda\mathbf{u}$$

S is a $d \times d$ matrix

$$\lambda_i \quad i = 1, 2, \dots, d$$

$$\mathbf{u}_i \quad i = 1, 2, \dots, d$$

Variance of projected data:

$$\text{var}(\mathbf{Xu}) = \mathbf{u}^T S \mathbf{u} = \mathbf{u}^T \lambda \mathbf{u} = \lambda \mathbf{u}^T \mathbf{u} = \lambda$$

$\lambda_1 > \lambda_2 > \dots > \lambda_d$

largest smallest

The principal components

$$S\mathbf{u} = \lambda\mathbf{u}$$

S is a $d \times d$ matrix

$$\lambda_i \quad i = 1, 2, \dots, d$$

$$\mathbf{u}_i \quad i = 1, 2, \dots, d$$

Variance of projected data:

$$\text{var}(\mathbf{Xu}) = \mathbf{u}^T S \mathbf{u} = \mathbf{u}^T \lambda \mathbf{u} = \lambda \mathbf{u}^T \mathbf{u} = \lambda$$

$$\lambda_1 > \lambda_2 > \dots > \lambda_d$$

Eigenvector of λ_1 : \mathbf{u}_1

Data projected on \mathbf{u}_1 will have highest variance.

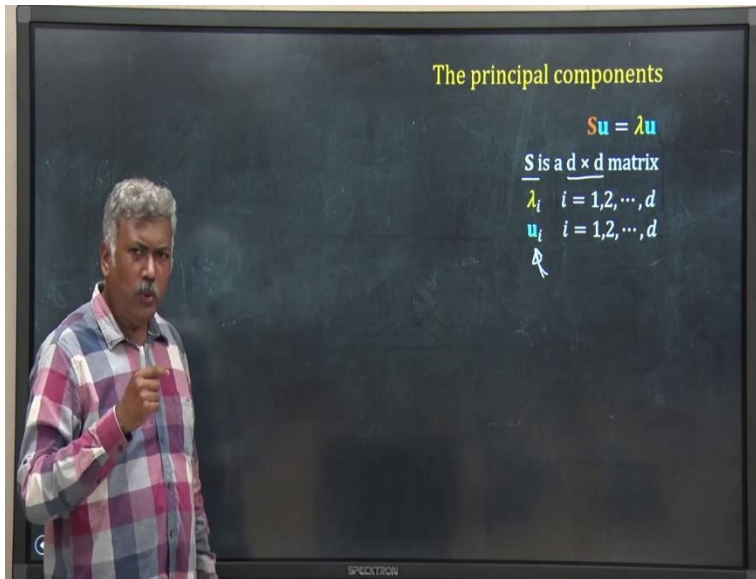
$\rightarrow \mathbf{u}_1$ is the Principal Component 1

Variance of projected data:

$$\text{var}(\mathbf{Xu}) = \mathbf{u}^T S \mathbf{u} = \mathbf{u}^T \lambda \mathbf{u} = \lambda \mathbf{u}^T \mathbf{u} = \lambda$$

$$\lambda_1 > \lambda_2 > \dots > \lambda_d$$

Eigenvector of λ_1 : u_1



Because, if my system is n into d , then S the covariance matrix is d by d that means, at max I can have d Eigen values, distinct d Eigen values and d Eigen vectors. So, these Eigen vectors we will call principal component, but remember our original goal we are starting from d dimensional data, d is very large and I want to reduce the dimension and now, I have got new coordinate. What are these new coordinate?

These new coordinates will be called principal component and those are these Eigen vectors and how many Eigen vectors I have? As it is a d by d matrix I have d Eigen vectors. So, I am starting with a data with d dimension, n into d and sample d variable, I am creating new vectors, new coordinates which are also d in number.

So, where am I reducing the dimension? That means, I have to discard some of these d Eigen vectors. I have to choose only those which will suffice for my work. How should I choose them? I will go back to my main logic. I want to create new coordinates in this case actually vectors because the vector will represent the direction of the coordinate axes such that the variation of the data along those axes when projected is high maximum.

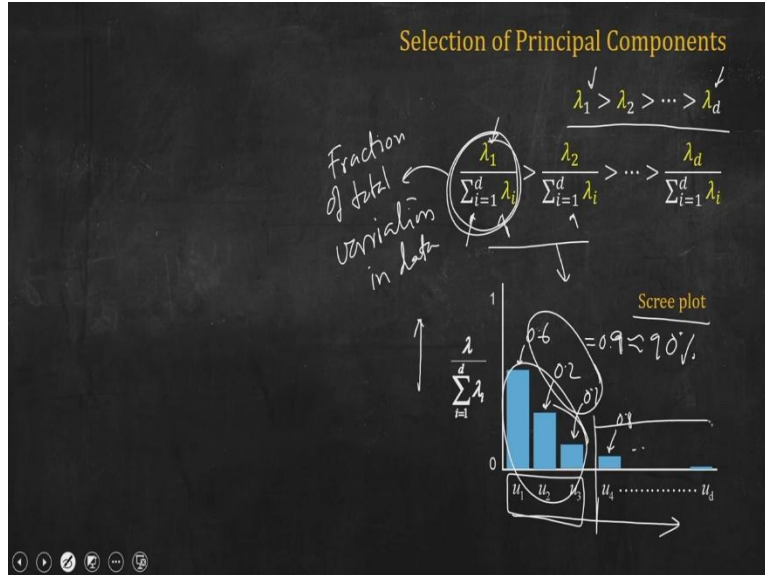
So, what I will do, I look into those Eigen vectors which will fulfill this criteria and how can I do that? I have a beautiful relationship, the variance of the projected data on a particular Eigen vector suppose u can be shown is equal to λ the corresponding Eigen value, that is what I have written here, the variance of X into u is equal to the Eigen value λ .

So now, I have d Eigen vectors and corresponding d Eigen values. So, arrange these Eigen values from the highest to the lowest one. So, this is the largest one and this is the smallest one. So, if this is the largest one that means the Eigen vector u_1 corresponding to this λ_1 must be a vector on which if I project my data, the variance of the predicted data will be highest.

So, I will call that Eigen vector u_1 as my principal component. So, the Eigen vector corresponding to λ_1 has the highest value, is u_1 and I will call that u_1 , the principal component one, because when the data is projected on u_1 , will have the highest variance of that projected data. Now, once you have selected the first principal component, go to the second highest λ , λ_2 , and the corresponding vector, Eigen vector is u_2 .

So, if I project my data on this vector u_2 , so this is a new axis on u_2 , I will have the second highest variance. So, I will pick this vector also. And I will call that second principal component PC 2. So, in this way, I can pick principal component 1, 2, 3, 4 up to d . But remember, I do not want to choose all these d vectors or d principal components. I want to truncate, remove most of them and to select a handful of them, how can I do that, that is also very simple.

(Refer Slide Time: 32:22)



$$\lambda_1 > \lambda_2 > \dots > \lambda_d$$

$$\frac{\lambda_1}{\sum \lambda_i} > \frac{\lambda_2}{\sum \lambda_i} > \dots > \frac{\lambda_d}{\sum \lambda_i}$$

So, this is the order of Eigen values lambda one is the highest and lambda d is the lowest. So, these are the raw values what I will do absolute value. So, what I will do, I will normalize them by the total Eigen value, means the sum of all Eigen values. So, that is what I have done here, I have solved all Eigen values and divided lambda one by that.

Obviously, this one will be still bigger than this one, but now, these values are related. So, they will vary from 0 to 1 and you can easily imagine actually this part this relative value will represent fraction of total variation in data. So, I have calculated the relative Eigen values which represent the fraction of total variation in the data.

So, obviously, the value is relative value for Eigen vector one, that is a principal component one will have the highest value, but it still will be within 0 to 1. So, what I do now, I plot these data as a histogram. So, that is called a scree plot, in that I have all the principal component in the horizontal axis and on the vertical axis, I have this relative value of Eigen value for each of these eigenvectors.

And for example, in this graph, imagine, suppose this value is 0.6. So, that means 60 percent of the variation in the data is captured by this Eigen vector, when I project the data on this Eigen

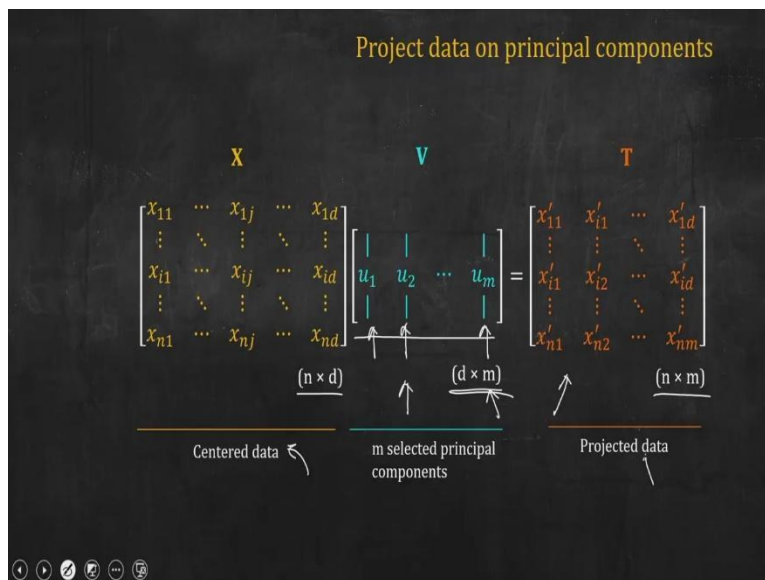
vector. Suppose, this u_2 is 0.2 and this is 0.1. Suppose this is 0.8 and so on. So, if I sum these three, I get 0.9.

So, that means if I take only these three components, principal component, or these three Eigen vectors and project the data on these three vectors, I will be able to capture 0.9 that means 90 percent of the variation in my data. That is good enough. D maybe initially 200 from that I have got three new coordinate axes, which are called principal component 1, 2 and 3.

If I project these 20 dimensional, 200 dimensional data on this three dimension, I still retain the variation, 90 percent of the variability and the variation in the data, dispersal in the data that is good enough for me, I have done a good job of reducing reduction of the dimension. So, what I will do, I will retain these three principal components and discard all others.

In this case, I have shown three you may go up to four or five or you may take the first two, it depends upon the data set that you are analyzing. So, once you have selected a subset from the first one first principal component, second principal component up to m th principal component, because they actually retain most of the variation in the data, capture most of the variation in the data, then what you have to do?

(Refer Slide Time: 35:48)



$$X V = T$$

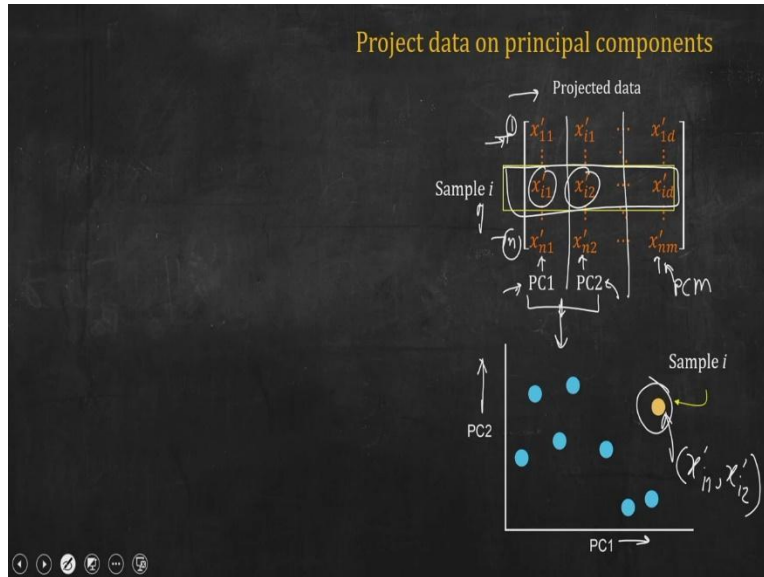
You have to project your data on these principal component because now these are the axes. So, you are projecting higher dimensional data on the dimension defined by these principal components. So, again, projection is nothing but multiplication of the original data matrix with each of these vectors.

So, rather than doing individual multiplication, what we usually do we write individual these vectors principal component one, two and the m th principal component, we stack them side by side to create a matrix. So, what I will get my ordinal data which is a centered data by default is n into d matrix, and I multiply that with d into n matrix because I have selected m , first m principal components starting from 1 to m , and each of these Eigen vectors has d dimension d into 1.

So, it is d into m matrix, and if I do the multiplication, I get the projected data and that dimension will be n into m because n is the number of sample, but now, they have projected on m dimension. So, d is bigger than m so, from d which is very large, I have come down to dimension m which is more of a handful 3, 4, 2 or something like that. So, this one is my m selected principal component, this is my projected data.

There is some terminology in PCA world what do we call, we call this matrix as the loading matrix and this projected data matrix, we often call scores matrix. So, now, what I have done, I have almost done the PCA, I have taken a d dimensional data with n sample n into d matrix X and then I project I have identified the Eigen vectors of the covariance matrix and I have selected the first few 1 to m Eigen vectors which we call principal component now, and I have projected my n into d data set into this m dimensional system.

(Refer Slide Time: 37:55)



So, what do I get, I get this projected data as I showed in the last slide. Now, I want to visualize it suppose. So, let us explore this projected data matrix. Each of these column they represent one principal component is not it. So, the first column is principal component one. So, if I project my data on this principal component, I will have the highest variation, variance in that and this is the second column is PC 2 and the last one is PC m.

And I want to project the data on only these two because I want to visualize it into dimension. I want to visualize your projection of the original data on these two dimension PC 1 and PC 2 and that is what I have done here. So, each of these row is a sample starting from first sample to the nth sample.

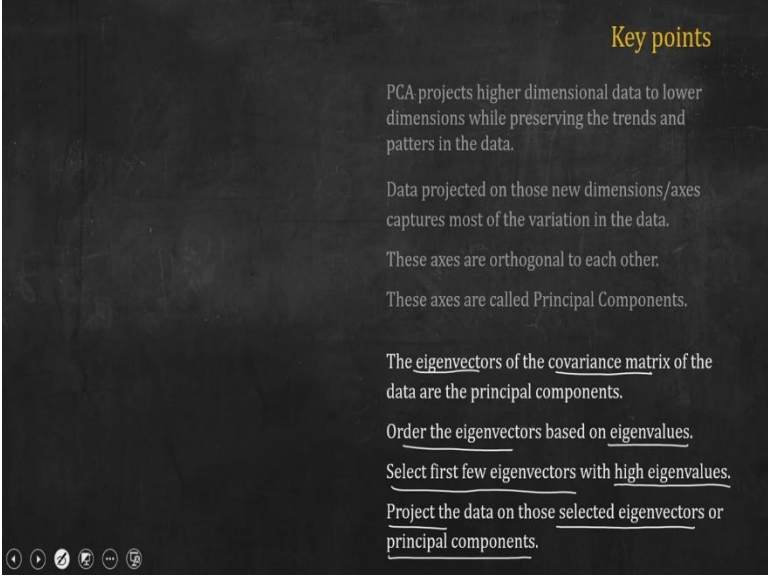
So, this yellow box represent my ith sample. So, each all the samples are shown here in this plot, where the horizontal axis is PC 1 and the vertical axis is PC 2, each of these data points are actually one of the rows and this one for example is sample i. So its coordinate is this one, this one, so this one is x'_{i1} prime, because it projected one, x'_{i2} prime.

So, in this way, I can take a part of the projected data matrix and visualize into two dimensions. If you want to do it in three dimension. Then take the PC 3. If you want to plot between PC 2 and PC 3, you take the PC 1 column and the PC 3 column and then plot it just like this scatterplot.

That is all, in the upcoming lecture, we will learn how to use R to perform PCA and create this type of scatterplot for principal component one versus two, principal component two versus

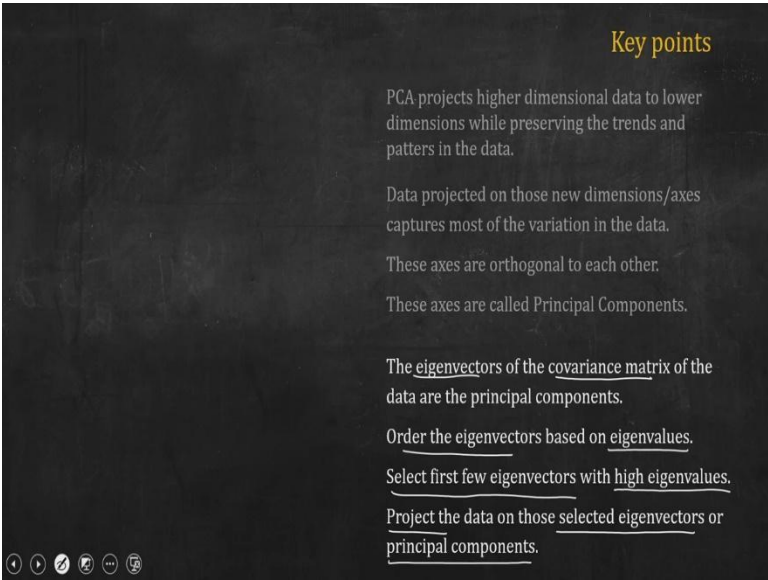
principal component, PC3 something like that. So, let me jot down what we have learned in today's lecture.

(Refer Slide Time: 40:11)



Key points

- PCA projects higher dimensional data to lower dimensions while preserving the trends and patterns in the data.
- Data projected on those new dimensions/axes captures most of the variation in the data.
- These axes are orthogonal to each other.
- These axes are called Principal Components.
- The eigenvectors of the covariance matrix of the data are the principal components.
- Order the eigenvectors based on eigenvalues.
- Select first few eigenvectors with high eigenvalues.
- Project the data on those selected eigenvectors or principal components.



Key points

- PCA projects higher dimensional data to lower dimensions while preserving the trends and patterns in the data.
- Data projected on those new dimensions/axes captures most of the variation in the data.
- These axes are orthogonal to each other.
- These axes are called Principal Components.
- The eigenvectors of the covariance matrix of the data are the principal components.
- Order the eigenvectors based on eigenvalues.
- Select first few eigenvectors with high eigenvalues.
- Project the data on those selected eigenvectors or principal components.

In this lecture, we have learned the logic and the mathematics behind principal component analysis. In principal component analysis, what we are doing, we are projecting a higher dimensional data to a lower dimension while retaining the pattern or trend in the data set. And we are projecting this data to new axes or dimensions.

And we are doing in such a way that the projected data captures most of the variation within the data. One thing you have to remember that these new dimensions or new axes should be orthogonal that means linearly independent to each other and these axes are called principal components.

Now, how do you do it mathematically? To do it mathematically, we find the Eigen vectors of the covariance matrix of the data and then these Eigen vectors are principal components. So, if I have d dimension then I have d Eigen vectors, but I do not want to return all those Eigen vector or all those principal component, I want to remove most of them and keep only a handful.

To do that what we do we order the Eigen vectors based on the Eigen values, because the Eigen values are equal to the variance of the projected data on that particular Eigen vector. So, we select first few Eigen vector with high Eigen values. For example, we choose the first vector which has the highest Eigen value then we choose the second Eigen vector here which has a second Eigen value.

So, these two are principal component one and principal component two and so on. And then we project the data on those selected Eigen vectors or principal components. That is all for this lecture. Thank you for being today with me. See you in the next one.