

**Essential Mathematics for Machine Learning**  
**Prof. Sanjeev Kumar**  
**Department of Mathematics**  
**Indian Institute of Technology, Roorkee**

**Lecture – 17**  
**Principal Component Analysis - II (Derivation and Examples)**

Hello friends, so welcome to module 17 of this course Essential Mathematics for Machine Learning. This particular module is in continuation of the last module in which we have introduced to you about Principal Component Analysis. And then we have seen that the principal component directions are nothing just the direction given by the eigenvectors of the covariance matrix which we have taken from the data.

Then we have to reduce the dimension of a data from let us say  $n$  to  $k$  where  $k$  is less than  $n$ ; then what we need to do? We will take first  $k$  largest eigenvalues of the covariance matrix and then their corresponding eigenvectors will expand the sub space by projecting the  $n$  dimensional data on to that  $k$  dimensional sub space we will reduce your data with dimension  $k$ . At the same time it will preserve as much randomness or variance as possible in your data.

So, let us derive it first mathematically.

(Refer Slide Time: 01:35)

Derivation of PCA:- The objective of PCA is to perform dimensionality reduction while preserving as much of the randomness in the high dimensional space as possible.

Let  $X$  be a  $n$ -dim random vector such that

$$X = \sum_{i=1}^n y_i \phi_i \quad \text{--- (1)}$$

where,  $\{\phi_1, \phi_2, \dots, \phi_n\}$  forms an orthonormal basis of  $n$ -dimensional space, and the coordinates  $y_i$  are given as

$$y_i = \langle X, \phi_i \rangle = X^T \phi_i \quad \forall i=1, 2, \dots, n$$

Now suppose, I want to represent  $X$  with fewer basis vectors, i.e. say  $m$  ( $m < n$ ). We can do this by replacing the coordinates  $y_{m+1}, \dots, y_n$  with some pre-selected constants  $b_i$  as

So, it is a bit mathematical, but very interesting. So, the objective of PCA is to perform dimensionality reduction while preserving as much of the randomness in the high dimensional space as possible.

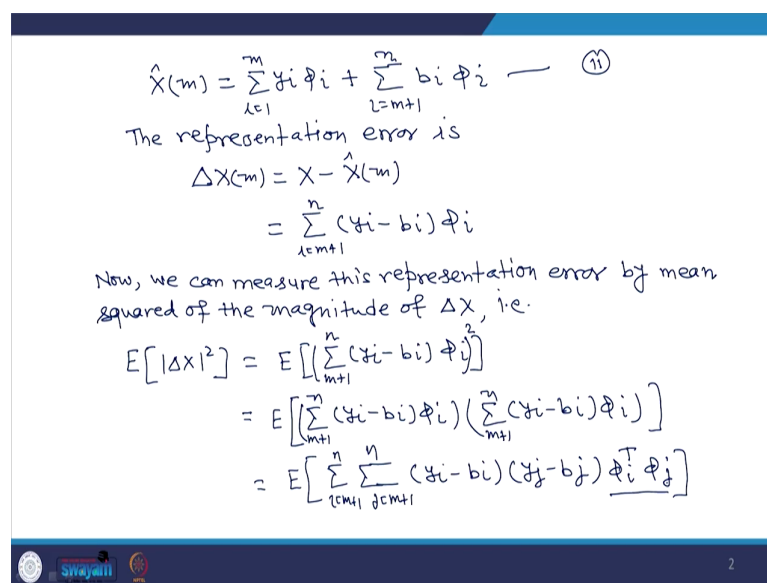
So, for this let  $X$  be a  $n$  dimensional vector such that;  $X$  equal to  $i$  equals to  $1$  to  $n$   $y_i, \phi_i$ ; where  $\phi_1, \phi_2, \phi_n$ , forms an orthonormal basis of  $n$  dimensional vector space in which  $X$  lie and the coordinates or the waiting coefficients  $y_i$  are given as; so  $y_i$  will become your inner product of  $X$  with  $\phi_i$  we have seen it earlier.

In case of  $r < n$  it will become the dot product for all  $i$  equals to  $1, 2, n$ . So, this is the representation of a vector  $X$  in  $n$  dimensional space. So, what we are taking a an orthonormal

basis. So,  $\phi_1, \phi_2, \dots, \phi_n$ , are basis vectors and we are taking the linear combination of these basis vectors to represent the  $X$ .

Now, suppose I want to represent  $X$  with fewer basis vectors that is say;  $m$  where  $m$  is less than  $n$ . So, how we can do this? So, we can do this by replacing the coordinates  $y_{m+1}, y_{m+2}, \dots, y_n$  with some of preselected constants  $b_i$  as.

(Refer Slide Time: 06:48)



$$\hat{X}(m) = \sum_{i=1}^m y_i \phi_i + \sum_{i=m+1}^n b_i \phi_i \quad (11)$$

The representation error is

$$\Delta X(m) = X - \hat{X}(m)$$

$$= \sum_{i=m+1}^n (y_i - b_i) \phi_i$$

Now, we can measure this representation error by mean squared of the magnitude of  $\Delta X$ , i.e.

$$E[|\Delta X|^2] = E\left[\left(\sum_{i=m+1}^n (y_i - b_i) \phi_i\right)^2\right]$$

$$= E\left[\left(\sum_{i=m+1}^n (y_i - b_i) \phi_i\right) \left(\sum_{j=m+1}^n (y_j - b_j) \phi_j\right)\right]$$

$$= E\left[\sum_{i=m+1}^n \sum_{j=m+1}^n (y_i - b_i)(y_j - b_j) \phi_i^T \phi_j\right]$$

So,  $X_m$  means I am representing the same  $X$ , but in lower dimensional space having dimension  $m$ . It will become  $i$  equals to 1 to  $m$   $y_i \phi_i$  plus  $i$  equal to  $m+1$  to  $n$   $b_i \phi_i$ .

Now, the representation error is so let me denote this representation error by  $\Delta X_m$  then this  $\Delta X_m$  becomes  $X$  minus  $X_k$  means this is the  $X$  represented in the higher dimensional space and this is in lower dimensional space that is with  $m$  basis element.

And if you use this equation let us say equation 2 and this is my equation 1; so by 1 and 2. This will become  $\sum_{i=m+1}^n y_i - \sum_{i=m+1}^n b_i \phi_i$ ; because first term will be cancel with first  $m$  basis elements.

Now, we can measure this representation error by mean square of the magnitude of the difference that is  $\Delta X$ . That is I am defining this error is  $E \Delta X^2$  that is the means square of the magnitude of  $X$  and this will become  $E \sum_{i=m+1}^n (\sum_{j=m+1}^n y_j - \sum_{j=m+1}^n b_j \phi_j)^2$  and summation on  $i$  and square of this.

So, it will become  $E \sum_{i=m+1}^n (\sum_{j=m+1}^n y_j - \sum_{j=m+1}^n b_j \phi_j)^2$  since square is there so I will write  $\sum_{i=m+1}^n (\sum_{j=m+1}^n y_j - \sum_{j=m+1}^n b_j \phi_j)^2$ . And this will become  $E \sum_{i=m+1}^n \sum_{j=m+1}^n (y_i - b_i \phi_i)(y_j - b_j \phi_j)$ . And then what I am having  $y_i - b_i \phi_i$ ,  $y_j - b_j \phi_j$  transpose  $\phi_j$ . So, I have open this summation and I got this one.

Now  $\phi_1, \phi_2, \phi_3$  all are orthonormal basis; so when  $i \neq j$  this dot product of this two basis elements will be 0. Otherwise if with the same if both are equal  $i = j$  it will become 1.

(Refer Slide Time: 11:15)

$$\begin{aligned}
 E[|Ax|^2] &= \sum_{i=m+1}^n E[(x_i - b_i)^2] \\
 \textcircled{1} \text{ Find } b_i \\
 \frac{\partial E[|Ax|^2]}{\partial b_i} &= 0 \Rightarrow -2(E[x_i - b_i]) = 0 \\
 &\Rightarrow \boxed{b_i = E[x_i]} \\
 E[|Ax|^2] &= \sum_{i=m+1}^n E[(x_i - E[x_i])^2] \\
 \text{where } x_i &= x^T \phi_i \\
 E[|Ax|^2] &= \sum_{i=m+1}^n E[(x^T \phi_i - E[x^T \phi_i])^2] \\
 &= \sum \phi_i^T E[(x - E(x))(x - E(x))^T] \phi_i
 \end{aligned}$$

So, in that way what we can have we can write the mean square error of the representation error equals to  $i$  equals to  $m$  plus 1 to  $n$  and expectation of  $y_i$  minus  $b_i$  square. In PCA what we need to do?

We need to find out  $b_i$  as well as  $\phi_i$  which minimize this particular error because it is the representation error and we want to preserve as much as information in lower dimensional space. So, what we need to do? We have to minimize the representation error.

So, now first find  $b_i$ . So, for finding  $b_i$  it will be having  $\frac{\partial E}{\partial b_i}$  equals to 0 and this will give me minus 2 times  $E$  of  $y_i$  minus  $b_i$  equals to 0; this means  $b_i$  equals to  $E$  times  $y_i$ .

So, now substitute this value of  $b_i$  here. So, what I will be having;  $E$  of  $\Delta X$  square equals  $2 \sum_{i=m+1}^n (y_i - E[y_i])^2$  and then square of this and where  $y_i$  is given by  $X^T \phi_i$ .

So, substitute this value here so then I will be having  $E$  of  $\Delta X$  square equals to  $\sum_{i=m+1}^n (X^T \phi_i - E[X^T \phi_i])^2$  and then I am putting this value here. So,  $E[X^T \phi_i - E[X^T \phi_i]]^2$  and square of this becomes  $\sum_{i=m+1}^n \phi_i^T E[XX^T] \phi_i - E[XX^T] \phi_i$  and then  $\phi_i$ .

Now, if you see carefully; what is this? It is the data covariance matrix as per the definition of covariance matrix. So, it is the data covariance matrix. So, what I can write.

(Refer Slide Time: 14:48)

$$E[\Delta x^2] = \sum_{i=m+1}^n \phi_i^T \Sigma_x \phi_i \quad \text{--- (3)}$$

① To find  $\phi_i$

$$\frac{\partial E[\Delta x^2]}{\partial \phi_i} = 0 \quad \text{subject to } \phi_i^T \phi_i = 1$$

$$\min_{\phi_i} E[\Delta x^2] \quad \text{s.t. } \phi_i^T \phi_i = 1$$

$$J(\phi_i) = \min_{\phi_i} \sum_{i=m+1}^n \phi_i^T \Sigma_x \phi_i + \sum_{i=m+1}^n \lambda_i (1 - \phi_i^T \phi_i)$$

then

$$\frac{\partial J(\phi_i)}{\partial \phi_i} = 0 \Rightarrow \boxed{\Sigma_x \phi_i = \lambda_i \phi_i} \quad \text{--- (4)}$$

Here,  $(\phi_i, \lambda_i)$  are the eigenpairs of  $\Sigma_x$ .



Swajam



The mean square of the representation error equals to  $\frac{1}{n} \sum_{i=1}^m \| \mathbf{x}_i - \mathbf{U} \mathbf{U}^T \mathbf{x}_i \|^2$ . So, let me give question number 3.

Now, what I have to find out  $\mathbf{U}$ . So, to find  $\mathbf{U}$  what I need to do  $\frac{d}{d\mathbf{U}} \text{Error}$  upon  $\mathbf{U}$  equals to 0 and subject to  $\mathbf{U}^T \mathbf{U} = \mathbf{I}$ . Since I am having this condition also I have to minimize this.

I have to find out  $\mathbf{U}$  means I have to minimize and for over  $\mathbf{U}$   $\frac{1}{n} \sum_{i=1}^m \| \mathbf{x}_i - \mathbf{U} \mathbf{U}^T \mathbf{x}_i \|^2$  subject to  $\mathbf{U}^T \mathbf{U} = \mathbf{I}$ ; since they are the orthonormal basis. So, by using Lagrange multiplier what I can write I can include this condition here.

So, minimize  $\mathbf{U}^T \text{covariance matrix} \mathbf{U}$  into  $\mathbf{U}^T \text{covariance matrix} \mathbf{U} + \lambda (\mathbf{U}^T \mathbf{U} - \mathbf{I})$  times 1 minus  $\mathbf{U}^T \text{covariance matrix} \mathbf{U}$ . And then if this is  $J(\mathbf{U})$  then to minimize it  $\frac{d}{d\mathbf{U}} J(\mathbf{U})$  over  $\mathbf{U}$  equals to 0. And this will give me  $\text{covariance matrix} \mathbf{U} = \lambda \mathbf{U}$ ; so, 4.

So, now this value of  $\text{covariance matrix} \mathbf{U}$  substitute here. So, basically what is  $\mathbf{U}$  and  $\lambda$  first? They are the eigenvalues; so here  $\mathbf{U}$  and  $\lambda$  are the eigenpair of the covariance matrix.

(Refer Slide Time: 17:56)

From (3) and (4), we have

$$E[\|\Delta x\|^2] = \sum_{i=m+1}^n \phi_i^T \lambda_i \phi_i$$

$$= \sum_{i=m+1}^n \lambda_i$$

In order to minimize the representation error,  $\lambda_i$ s need to be smallest eigenvalues.

→ Therefore, in PCA, we choose  $m$  eigenvectors corresponding to the  $m$  largest eigenvalues  $\lambda_i$  of the covariance matrix  $\Sigma_x$  as the principal directions.

inf. preserve =  $\frac{|\lambda_1 + \lambda_2 + \dots + \lambda_m|}{|\lambda_1 + \lambda_2 + \dots + \lambda_m + \dots + \lambda_n|} \times 100$

⇒

Now, from 3 and 4; we have  $E \Delta X$  square equals to summation  $i$  equals to  $m$  plus 1 to  $n$  and then  $\phi_i^T$  into  $\lambda_i \phi_i$  and this becomes  $i$  equals to  $m$  plus 1 to  $n$   $\lambda_i$  because  $\phi_i^T \phi_i$  will become 1 and for when  $i$  not equals to there they will become 0.

So, now mean square of the representation error is sum of the eigenvalues. So, what I need to do here; so in order to minimize the representation error which is mean square  $\lambda_i$  is need to be smallest eigenvalues; means what I need to do?

I am having  $n$  eigenvalues out of  $n$  I have to choose  $n$  minus  $m$  minus 1 eigenvalues and sum of those would be the minimum. So, what those  $n$  minus  $m$  minus 1 eigenvalue I will choose; I will choose the smallest one and I will ignore them. And hence to minimize the representation error or preserving the maximum variance maximum randomness in my data after projecting



into lower dimensional space I have to select only eigenvectors corresponding to larger eigenvalues; so this is the meaning of this

So, means what we conclude from this derivation; therefore, in PCA we choose  $m$  eigenvectors corresponding to the  $m$  largest eigenvalues  $\lambda_i$  of the covariance matrix  $\Sigma_X$  as the principal directions.

So, this is the prove; why we are taking principal directions or principal components in the direction of eigenvectors corresponding to largest eigenvalues of the covariance matrix.

(Refer Slide Time: 21:50)

PCA and SVD:-

Let the data matrix be  $C$  of size  $n \times p$ . Then the principal directions (components) are coming from the eigenvectors of  $\Sigma_X = \frac{1}{n-1} C^T C$  ( $p \times p$  matrix)

Now if the SVD of  $C$  is given as  $C = USV^T$ , then

$$\Sigma = \frac{1}{(n-1)} C^T C = \frac{1}{(n-1)} (USV^T)^T \cdot USV^T = \frac{1}{(n-1)} V S^T U S V^T$$

$$= \frac{1}{n-1} V S^2 V^T$$

The columns of  $V$  are the eigenvectors of  $\Sigma$ .  
Therefore if the SVD of the data matrix is  $C = USV^T$ , then columns of  $V$  give the principal direction;  $\mathcal{C}V = USV^T V = US \rightarrow$  principal comp

Now, one more important thing what is the relation of PCA with SVD. So, PCA and SVD; so let the data matrix be;  $C$  of size  $n$  by  $p$ . So, means I am having  $n$  number of samples and each one is having  $p$  direction. So, my data original dimension is  $p$ .

Then the principal directions and therefore, components are coming from the eigenvectors of the covariance matrix. So,  $\Sigma^C$  and that is  $1 \text{ upon } n \text{ minus } 1$ . So, sometimes we can write  $1 \text{ upon } n$  also, but in variate generally we write  $1 \text{ less}$ ;  $C^T$  transpose into  $C$ ; which is a  $p$  by  $p$  matrix.

Now, if the SVD of  $C$  is given as  $C$  equals to  $U S V^T$  then the covariance matrix is  $1 \text{ upon } n \text{ minus } 1$   $C^T$  transpose into  $C$ . So,  $1 \text{ upon } n \text{ minus } 1$  and then  $C^T$  transpose is  $U S V^T$  transpose transpose into  $U S V^T$ .

And this comes out to be  $1 \text{ upon } n \text{ minus } 1$ . So, this will become  $V S U^T$  into  $U S V^T$  so  $V S^2 V^T$ . So, since this covariance matrix is a symmetric matrix and  $V$  is an orthogonal matrix. So, what I am having now this is orthogonal diagonalization of  $\Sigma$ .

So, what are the eigenvectors of  $\Sigma$ ? The columns of  $V$  are the eigenvectors of  $\Sigma$ . And what are the eigenvectors of  $\Sigma$ ? They are the principal directions they are giving you the principal components. Therefore, if the SVD of the data matrix is  $C$  equals to  $U S V^T$  transpose, then columns of  $V$  give the principal direction.

And then if you want to project then it will become  $X V$  and  $X V$  will be  $U S$ ; so not  $X$ ,  $C C^T V$  sorry  $C V$ ,  $U S$ ,  $V^T V U S$  gives the principal components. Furthermore; how much information I have preserve; that you can see from here. So, this much information we are discarding.

So, information preserve is if you are coming from  $n$  dimensional space to  $k$ . So,  $\lambda_1$  plus  $\lambda_2$  up to  $\lambda_k$  upon  $\lambda_1$  plus  $\lambda_2$  plus  $\lambda_k$  plus up to  $\lambda_n$  and all are  $S$  Uth value into 100; will give you the percentage of information preserve in your data ok.

(Refer Slide Time: 27:26)

Ex.1:- Compute the PCs for the following 2D data:

$$X = (x_1, x_2) = (1, 2), (3, 3), (3, 5), (5, 4), (5, 6), (6, 5), (8, 7), (9, 8)$$

$x_1$	$x_2$
1	2
3	3
3	5
5	4
5	6
6	5
8	7
9	8

8x2

$$\Sigma_x = \frac{1}{7} C^T C = \begin{pmatrix} 6.25 & 4.25 \\ 4.25 & 3.5 \end{pmatrix}$$

$$\lambda = 0.4081 \quad \text{and} \quad 9.3419 \checkmark$$

$$\vec{v} = \begin{pmatrix} 0.5883 \\ -0.8086 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 0.5883 \\ 0.8086 \end{pmatrix} \checkmark$$

$$y = 0.5883 x_1 + 0.8086 x_2$$

→ 1-D

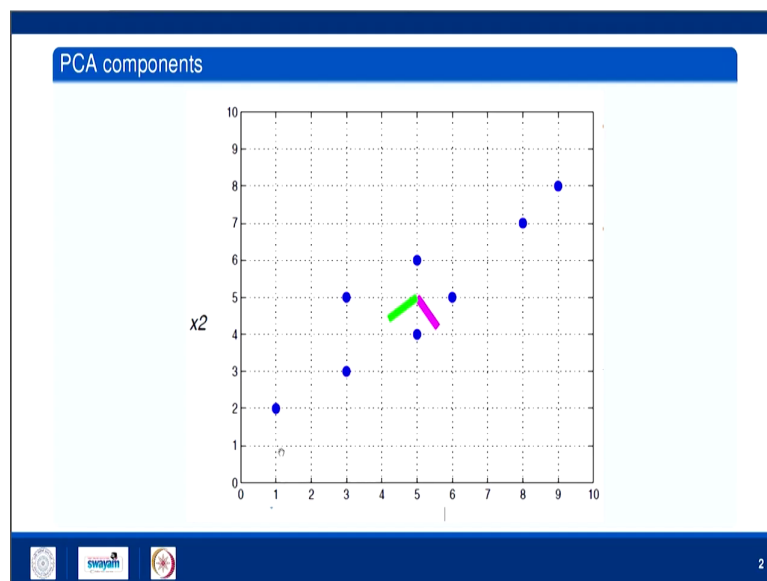
So, now let us take some example. So, example 1 so compute the principal components for the following; 2 dimensional data. So, data is  $X$  equals to  $x_1 \times x_2$  and these are points given by 1, 2, 3, 3, 3, 5, 5, 4, 5, 6, 6, 5, 8, 7, and then 9, 8.

So, means what I am having  $x_1$  column  $x_2$  column and then data is; 1, 2, 3, 3, 3, 5, 5, 4, and so on 9, 8. So, first what I need to do this is my data matrix; so 1, 2, 3, 7, 8, 6, 7, 8. So, it is 8 by 2 matrix this is the matrix  $C$ .

First I have to find out  $\Sigma C$ ; that is the covariance matrix that will become 1 upon 7 means 8 minus 1, 7; 8 are the sample point  $C$  transpose into  $C$  and this comes out to be 6.25, 4.25 4.25 3.5 Now, eigenvalue of this is  $\lambda$  equals to 0.4081 and 9.3419. So, eigenvector corresponding to this eigenvalue means 0.4081 is 0.5883 and for second one is 0.8086.

Similarly, eigenvector for corresponding to this is 0.5883, 0.8086. So, these eigenvectors are giving you the direction of principal components or principal directions. So, if I want to reduce this data into one dimension then that one dimension if data is y will become because this is the larger eigenvalue. So, this will be the principal direction. So,  $0.5883 \times 1$  plus  $0.8086 \times 2$ ; all these pairs will projected to 1 D that will be our y.

(Refer Slide Time: 30:36)



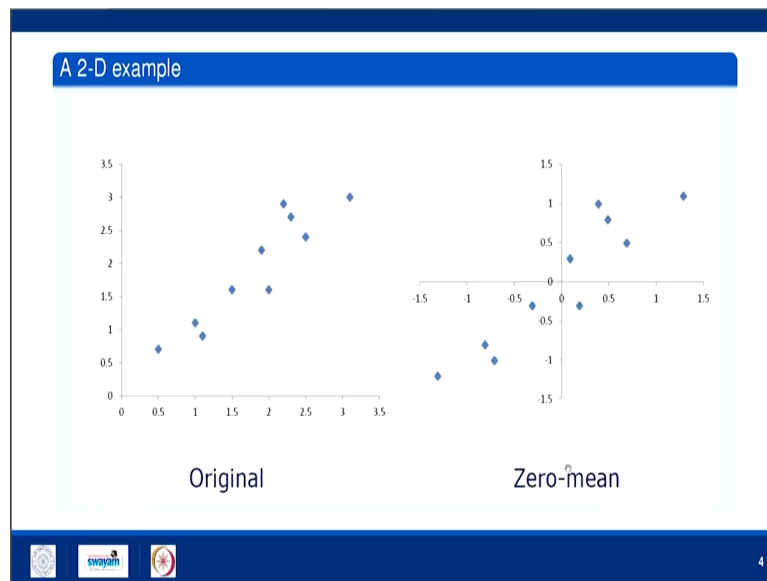
So, if you want to see this is the example these are these points; I have plotted in 2 D and this green one is giving the direction of principal component. While the second one is the another eigenvector.

(Refer Slide Time: 30:50)

A 2-D example			
x	y		
		0.69	0.49
2.5	2.4	-1.31	-1.21
0.5	0.7	0.39	0.99
2.2	2.9	0.09	0.29
1.9	2.2	1.29	1.09
3.1	3	0.49	0.79
2.3	2.7	0.19	-0.31
2	1.6	-0.81	-0.81
1	1.1	-0.31	-0.31
1.5	1.6	-0.71	-1.01
1.1	0.9		

Another example consider this again two dimensional data x and y. And I am having something like 2, 3, 4, 5, 6, 7, 8, 9, 10 sample points. First what I am doing; I am shifting the center of this data to the origin at 0 0. So, what I need to do for these? Means I want zero mean of the data; so what I need to do? I have to subtract the mean of x from all these entry and mean of y from all these entry. So, after doing this you are getting this data.

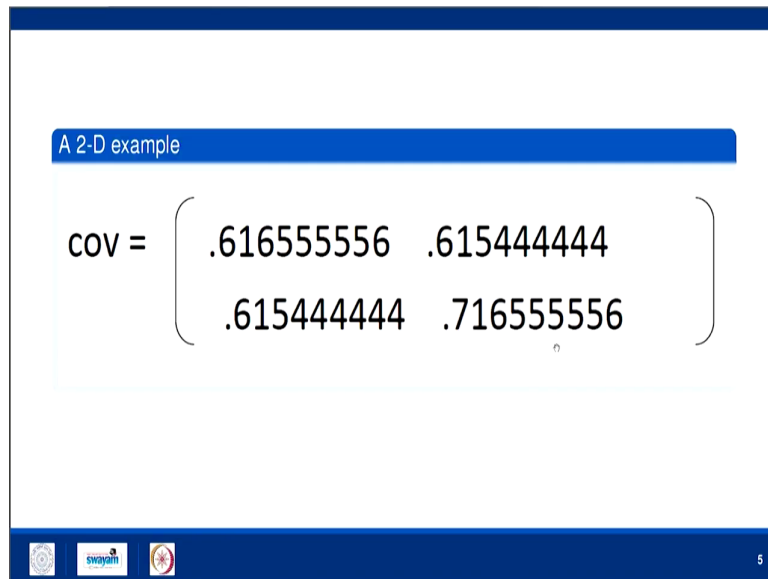
(Refer Slide Time: 31:21)



So, this is the original data this is the data with zero mean just subtracting from each column their respective mean. Now from this data what I will do? I will find out the covariance matrix.

(Refer Slide Time: 31:34)




A 2-D example

$$\text{cov} = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$


And covariance matrix comes out to be this one. Now I will find out the eigenvalues and eigenvector of these.

(Refer Slide Time: 31:42)

A 2-D example

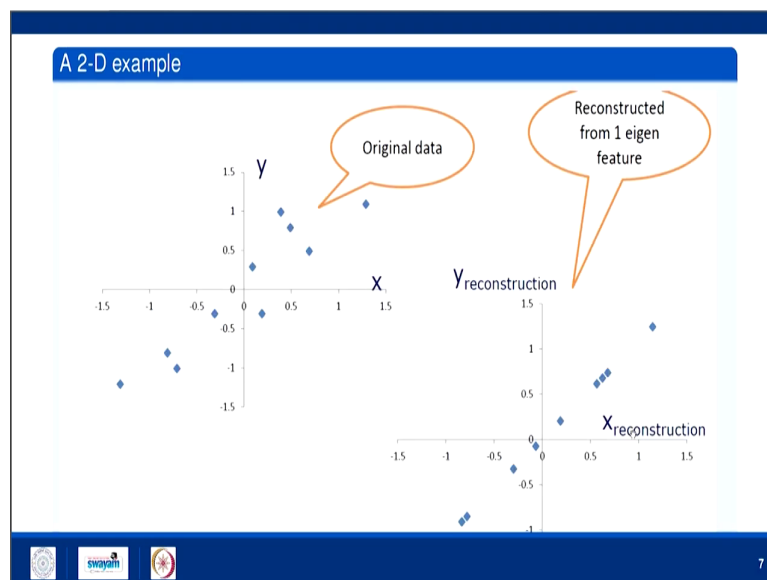
$$\text{eigenvalues} = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$
$$\text{eigenvectors} = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$


6

So, eigenvalues are 0.04908 and 1.2840 similarly eigenvectors are these one. So, this is the bigger eigenvalue; so this is the direction of principal eigenvector. So, if I want to reduce this data to 1 D. So, I have to write all the data that is x y given originally as the linear combination as  $0.677 \times 1$  or x plus  $0.735 y$ .



(Refer Slide Time: 32:16)



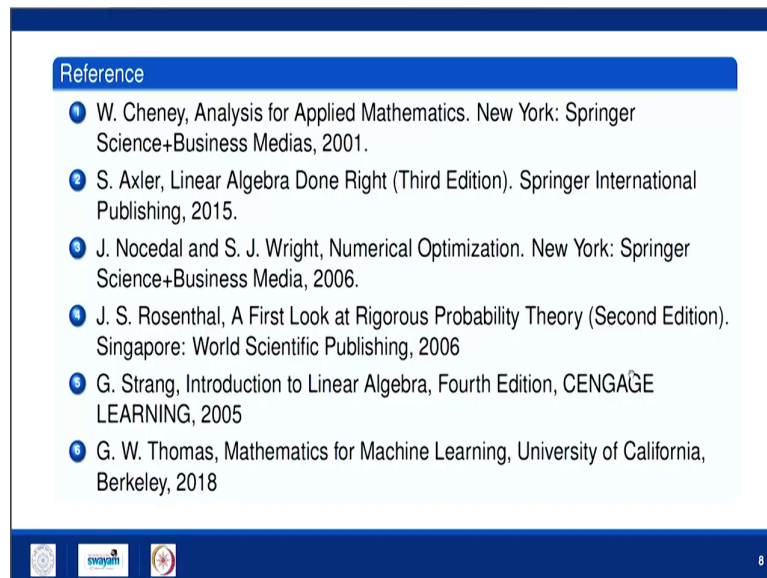
And if I do this you can see this is original data and this is 1 dimensional data; means after projecting on the principal eigenvector. So, in that way you can reduce the dimension of your data.

Suppose your data is having a dimension 100 you want to reduce it up to 20; so what you have to do? You will find out the covariance matrix that will be 100 by 100 matrix. You will find the eigenvalues of that covariance matrix; so 100 eigenvalues you will select top 20 eigenvalues corresponding eigenvectors.

So, those will be orthogonal to each other. So, those 20 eigenvectors will give you or will expand the a 20 dimensional space. And if you project your 100 dimensional data to those 20

dimensional space your data dimension will reduce to 20. So, this is all about principal component analysis.

(Refer Slide Time: 33:15)



So, these are the references. In the next lecture we will learn another very useful concept that is called linear discriminant analysis. So, why we need linear discriminant analysis when we are having PCA type of thing.

Thank you very much.