Indian Institute of Technology Madras
Presents

NPTEL
NATIONAL PROGRAMME ON TECHNOLOGY ENHANCED LEARNING

Pattern Recognition

Module 02

Lecture 13

Principal Component Analysis [PCA]

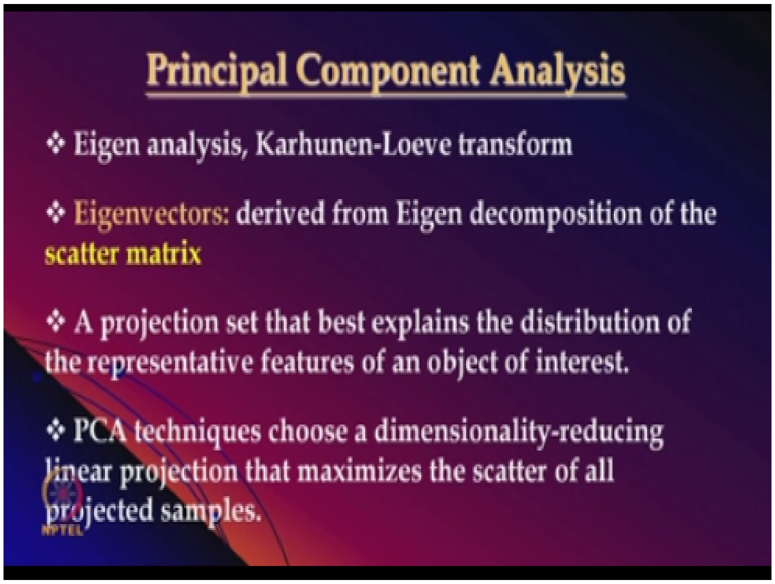
Prof. Sukhendu Das
Department of CSSE, IIT Madras

So welcome back to the lecture series on pattern recognition. We have discussed many details about different type of classifiers, both supervise, non supervise methods. Typically if you take the names, we started with base classifiers, the supervised one, then we also discussed other methods of classification and clustering to take a few examples DV scan, cane and neighborhood rule, the K means algorithm and so on.

There are two other types of variance of classifiers which are very commonly used in many pattern recognition application, one of them is of course a supervised, the other is unsupervised method. We will start with one of them, and these method can be called an unsupervised method of classification as well as clustering which is called the principal component analysis, the PCA.

It is a widely used method, its main application has been in many other applications like data dimension and data deduction, then compression and so and so forth. But we will see if it can be used for classification as well. And we will call it as unsupervised classification, because in this case, the class samples of the data are not know. There are many variance of these terms of the PCA which is used one is called the KLT or the Karhunen-Loeve transform.

(Refer Slide Time: 01:52)



Principal Component Analysis

- Eigen analysis, Karhunen-Loeve transform
- Eigenvectors: derived from Eigen decomposition of the scatter matrix
- A projection set that best explains the distribution of the representative features of an object of interest.
- PCA techniques choose a dimensionality-reducing linear projection that maximizes the scatter of all projected samples.

The KLT has given the slide. People also casually sometimes called PCA as an SVD, the similar valid decomposition which we studied much earlier in the course under basics of linear algebra and vector spaces. Sometimes PCA and SVD are used in dischangebly although, I must mention it is not correct to use that, but one must remember why it is so, because SVD is, or the singular value decomposition is the main tool or the method used for obtaining a PCA.

So what are principal components, and what is the principal component analysis and why SVD, because we need to do some Eigen decompositions. If you look back into the slide it is based on an Eigen analysis, the KLT or the PCA and the main purpose of this is to obtain a set of eigenvectors which are derived from the Eigen decomposition of a scatter matrix.

We will define what is this scatter matrix; it may have been also used in other applications throughout this course, the other concepts discussed. And we will have descriptions in this slide, so that in spite of whatever I tell you, you always go back and look into this video with the slide disproving concepts of PCA as well as SVD. And we would also take a toy example at the end to describe what is this PCA okay.

As we have been doing many other examples, we do take toy examples in 2-D and 3-D, so we will do that and, but remember one thing that this is involving Eigen decomposition and matrix analysis. So if you have not gone through that couple of lectures on matrix decomposition, please go through that. Al though I will have at least one or two concepts discussed with the help of sides on Eigen decomposition or SVD.

So we aim to actually obtain a projection set that best explains distribution of the representative features of an object of interest. This object of interest could be re particular pattern or any particular object which we are trying to retrieve from the image, but of course this PCA is applicable for any other type of data. People have applied it for speech signals, for other type of multidimensional signals which happen in practice.

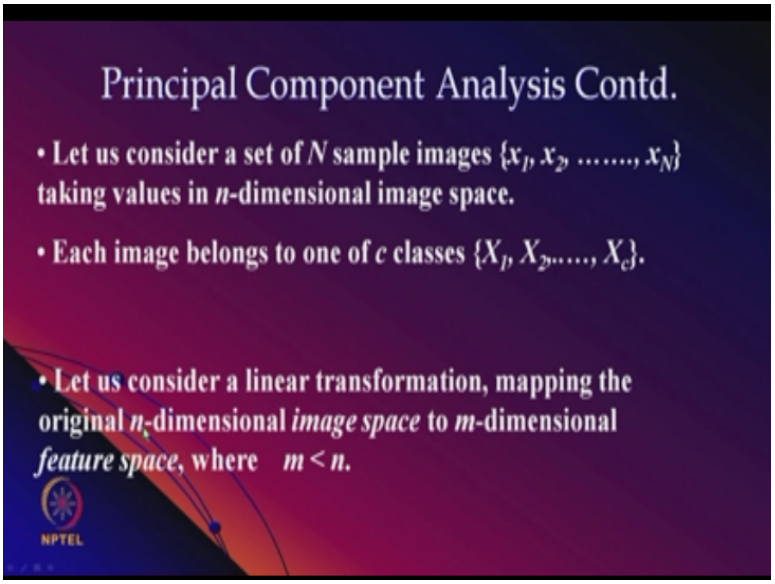
Image, video, acoustics, speech music, video, etc.., and many other applications as well. And what is this projection set, and what is this representation for the distribution we will actually look at it. In some sense, you must remember although we are talking about representation, this is a non-parametric or parameter fee method of obtaining a representation of the data.

PCA chooses a dimensionality reduction projection that maximizes the scatter of all projected samples. So we will see that with the example what we mean by this scatter of the examples, you know the scatter is a very important term, you need to do that using, we have done that is in the covariance matrix, when we talked of distance functions. And any method of clustering or classification which you do involves trying to look at the scatter of the data.

So we will look at the scatter of the data, so it becomes very, very important to model that and use it for our analysis. And although the main aim is to reduce the dimensionality of the problem by PCA and also in that process choose the best set of dimensions in a different domain. It is not like now, if you have an N dimensional data, just choose some arbitrary based M dimensions, where M is lass that N, that is not the main aim of PCA.

But we transform it to some other domain and that dimension of that projected domain will be less than the original dimension of the data. And in the projected domain we choose certain directions where the scatter is maximized, we will see how this is done. So look at the last line, this idea is important, where we are talking about trying to maximize the scatter of the samples which are projected onto another dimension.

(Refer Slide Time: 06:04)



Principal Component Analysis Contd.

- Let us consider a set of *N* sample images $\{x_1, x_2, \ldots\ldots, x_N\}$ taking values in *n*-dimensional image space.
- Each image belongs to one of *c* classes $\{X_1, X_2, \ldots, X_c\}$.
- Let us consider a linear transformation, mapping the original *n*-dimensional *image space* to *m*-dimensional *feature space*, where $m < n$.

So carrying on the discussion on PCA, I have used a set of N sample images as an example, but it could be any other data. So what this basically means is, X1, X2, X3…… Xn are different image samples. Instead of image samples, it could be any other data. I am choosing an example of images, because in many of our examples we have also chosen that, and some of the toy examples and analysis which we have done and which we are going to show also we have used lots of image samples.

Because that is an easy method of illustration and visualization which can help you to understand concepts better. So what we are trying to see here, if you are given N different images, it could be any other data mind you, but let us say N different images. This N different images pertain to one type of an object within that we will have class. Let us say you have image samples of fruits, or image sample of cars, or even let us say faces for the example of a person authentication in case of biometry.

So you are given N different image such samples, all belonging to the same or similar type, but of different classes. That means if you have N sample images of M different individuals. So M could be 10, M could be 100, that means 100 sample images of 10 different individuals, that means typically on an average you will have about 10 samples per person, or per class. So if you have 10 different fruits, 10 different cars, 10 different bicycles for each such class you have 10 different samples, so totally you will have 100 different image samples.

This could be any other data which would, on image like it could be speech, it could be audio, it could be some vibrations signals observed using some sort of a measuring device for any other measurements. So you have N different images and of course you have a dimensionality for each data. In this particular case of an image, if you ask me what is the dimensionality of each and what is the feature or the feature vector here, each image is of its particular size.

Let us say WxW or DxD, in that – so in such a case you will have the dimensionality for each data is D^2 or W^2. So if you look back there are N sample images X1, X2, X3 up to XN, each of this X is a vector of certain dimension depending upon the dimension of the feature space you are working in. instead of images, if you work with a certain feature dimension, because certain types of data set samples which are given for the purpose of pattern recognition, pattern classification or machine learning applications are given for a certain dimension.

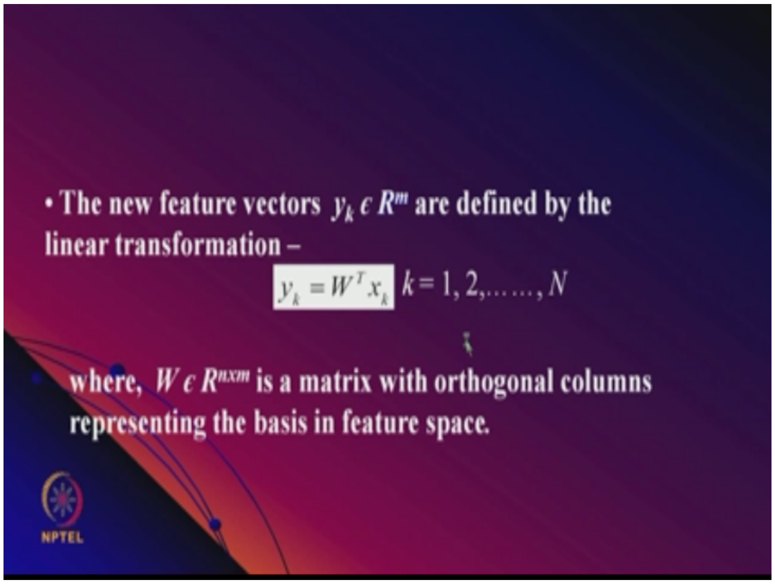
It could, the dimension could be 30, it could be 100 or a few 1000. And there are different sample is given for all the classes put together or for each particular class the total number of samples is capital N.

The dimension will be D and each image belongs to one of this small c classes okay so this capital C indicates the class level okay and typically the capital C will be of course much, much less than capital N because for each class you will have a certain number of samples so average number of samples per class multiply by the total number of classes should actually give this capital N.

So remember once again the capital X is the class level small x indicates the data point of a certain dimension okay and what we want to do with the help of PCA is to do a linear transformation mapping the original n dimensional image space data is capital N is this d^2 small n is not same as capital N be careful with the notation this small n indicates the dimension of the sample okay so if d cross d d^2 is the small n if it is w cross w image size then $n=d^2$ and if it is some other feature data research whatever the dimension of the problem d is typical d we have used d in many of our previous lectures as the dimension of the sample or the domain in which we are working.

And here we just be careful that we are using capital small n as a notation so we want to map or project either mapping or a projection you can think of from the original n dimension image space to m dimensional feature space where typically m is less than n sometimes it could be much, much less than small n okay.

(Refer Slide Time: 10:42)



- The new feature vectors $y_k \in R^m$ are defined by the linear transformation –

$$\boxed{y_k = W^T x_k} \quad k = 1, 2, \ldots\ldots, N$$

where, $W \in R^{n \times m}$ is a matrix with orthogonal columns representing the basis in feature space.

So the new feature vectors okay remembered so hence forth will not discriminate image pixels or original data points from feature vector dimension if we take the original sample point that6 is also true so the feature vector versus of size of dimension n small n and we are projecting down to lower dimensions small m and we will represent that by yk so xk is projected to yk each of this yk is a belonging to dimension m and liner transformation is given by this particular expression here okay.

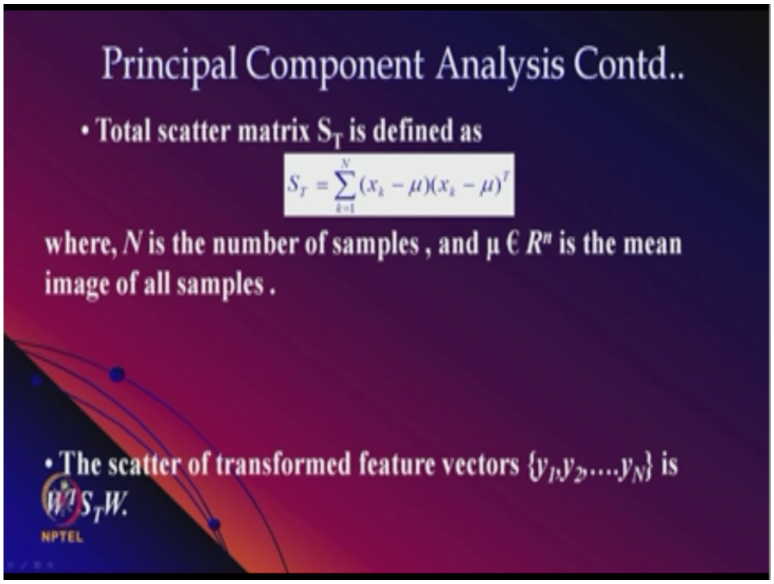
We talk of certain matrix for the timing you can think of this as sort of eight matrix okay but this is the one which is responsible for the mapping or the transformation from an higher dimensional description or data space n to a smaller dimension m so if we look back here xk is actually of dimensions small n which is larger than the dimensions yk which is m here and since small n is more than m we have seen that in the previous slide so the w is a responsible for the transformation.

So what will you the dimension of the size of the w it will be m cross m because this is a sixe n this is sixe m so this is matrix with orthogonal columns representing the basis in some features space so you want to find out this w that is the purpose of PCA is to find out what is this w which will map x to y such that you not only have reduction of the dimension but you also have some nice properties like in the new dimension m of size m to the sample yk will have some properties of maximum scatter along those dimensions okay.

It is possible that certain dimensions of n are reluctant or certain dimension if n do not have a scatter which represent the samples so in some sense they will be given less importance or you can consider them to the eliminated in some sense not directly elimination but they will be giving you very, very less importance in the transforms space and this transform space of size m again I repeat less than m will be holding the maximum scatter in some sense.

So we will define the criteria which will be actually be minimized with the help of PCA so remember this is the expression target w is the weight matrix which has to be obtained with the help of PCA which will help us to implement this projection here and of course the number of samples of k will be running from so this will be applicable for all the samples capital N number of samples which are available in the larger or original dimension.

(Refer Slide Time: 13:40)



Principal Component Analysis Contd..

- Total scatter matrix S_T is defined as

$$S_T = \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T$$

where, N is the number of samples , and $\mu \in R^n$ is the mean image of all samples .

- The scatter of transformed feature vectors $\{y_1, y_2, \ldots. y_N\}$ is $W^T S_T W$.

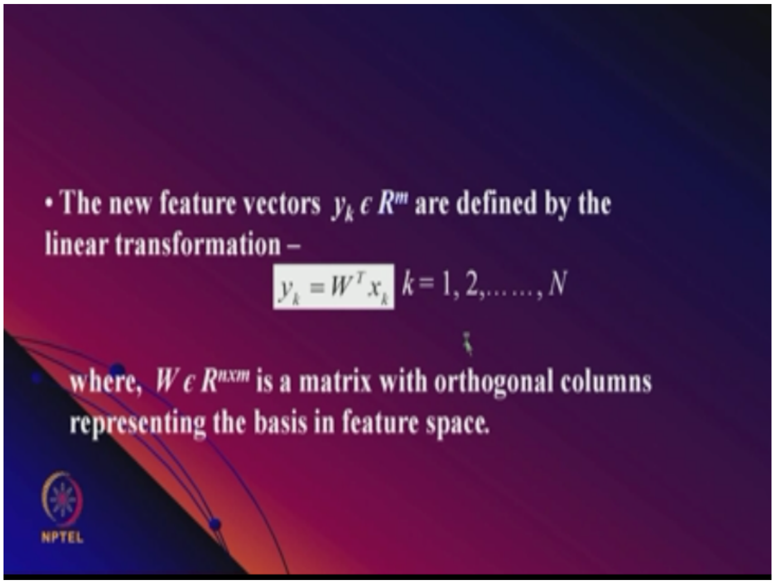
And to implement this we look at an expression of an scatter matrix St here which is given by this particular example we have already defined what is this sk k is index for the number of samples running from 1 to n and μ indicates the overall calls μ, μ indicters the overall class μ for all the samples so xk-μ basically means you are subtracting the overall class mean from the number of samples correct so you are subtracting the overall class mean from the sums we will look at the expression here $xk-\mu$ is can be considered to be column vector and $xk-\mu$ transpose will be consider to the row vector.

And this resultant multiplication is also outer product okay the outer product that means this product will actually give you n cross n sorry small n cross n because the dimension that is what you will get some over all the samples so you subtract the mean from each samples subtract the mean from each staple create the outer product some over all of them are that will give you this scatter matrix for the overall sample okay.

What is scatter matrix properties will have? It will have some properties related to the distribution of the samples remember it is non parametric value it just a matrix indicating the samples which you have and an another expression of this σ_{ij} which can be consider as this is nothing new to you it is if this is a element of the scatter matrix it can be visualized to be the expectation of this so we have seen this scatter matrix which is another term of so if you look at the expression you will actually going to get the co variance matrix.

Along the diagonal you will have the variances and along the off diagonal terms you have the correlation between two particular dimension I and J for an arbitrary element I common j okay so capital N is the number of the samples in the image and μ is the class mean for all samples of which is any other data and the scatter of the transformed feature vectors will prove this very soon after you have done the transformation using W it will be actually having the scatter as given here. So this is the scatter after you have done the transformation what is the transformation?

(Refer Slide Time: 16:06)



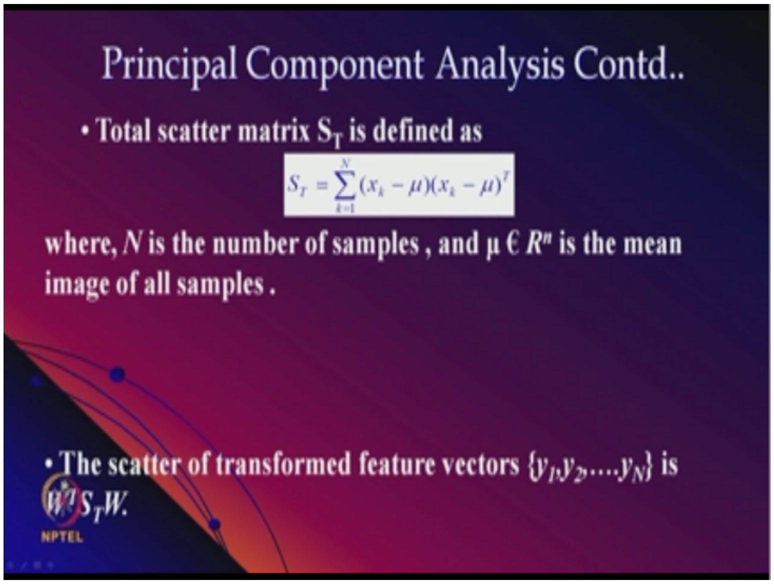
- The new feature vectors $y_k \in R^m$ are defined by the linear transformation –

$$\boxed{y_k = W^T x_k} \quad k = 1, 2, \ldots\ldots, N$$

where, $W \in R^{nxm}$ is a matrix with orthogonal columns representing the basis in feature space.

Let us go back this is the transformation we are talking about so if capital S is the scatter of X then the scatter of yk will be given by this particular expression here.

(Refer Slide Time: 16:16)



Principal Component Analysis Contd..

- Total scatter matrix S_T is defined as

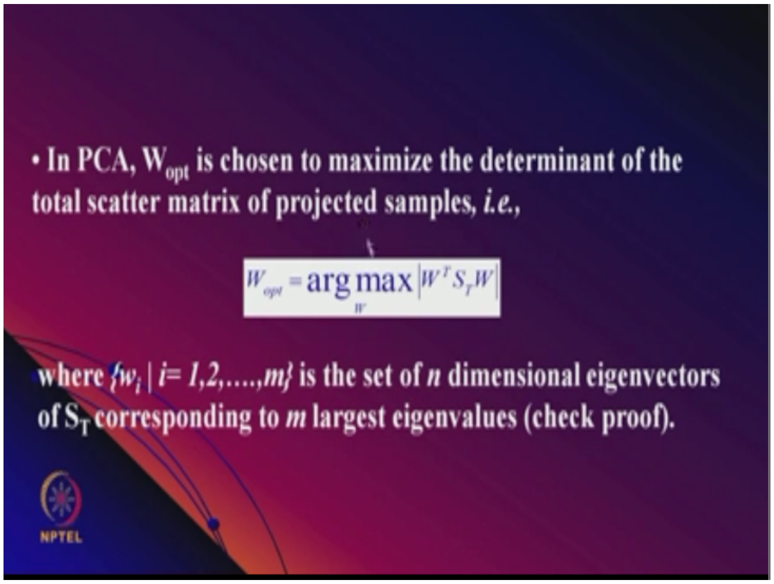
$$\boxed{S_T = \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T}$$

where, N is the number of samples , and $\mu \in R^n$ is the mean image of all samples .

- The scatter of transformed feature vectors $\{y_1, y_2, \ldots. y_N\}$ is $W^T S_T W$.

This is the scatter of the samples yk if S_t is the scatter of samples in the original domain of xk okay so this we will prove this I and show why this is so little bit of analytics and we will show that this type of w transpose S_t multiply by W actually maximize the scatter along the certain directions in a certain order based on PCA.

(Refer Slide Time: 16:41)



- In PCA, W_{opt} is chosen to maximize the determinant of the total scatter matrix of projected samples, *i.e.,*

$$W_{opt} = \arg\max_{W} \left| W^T S_T W \right|$$

where *{w_i | i= 1,2,....,m}* is the set of *n* dimensional eigenvectors of S_T corresponding to *m* largest eigenvalues (check proof).

So we have to choose W that is our main aim in principal component analysis in PCA W optimal this indicates remember you can choose any arbitrary W but it does not will not give you a PCA it will give some transformation if you take a random instead of values for the matrix W as size m cross n and you can project the samples you can give lower dimension mapping anyway but it will not satisfies the criteria of maximum scatter along certain directions in reducing order will see that.

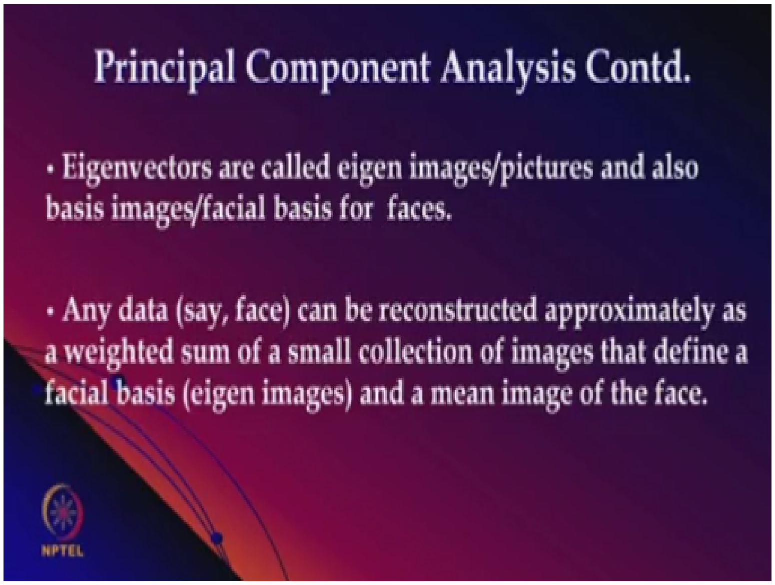
Those properties with the after projections to like that one satisfy remember so we will look at a optimal value of W which is chosen to maximize the determinant of total scatter matrix of projected samples so this is what we are looking at that if you choose certain W such that this particular value is maximized St what we have seen in the previous slide is the scatter matrix in the form the original sample X in the dimension N small n.

And if you choose a W such that this expression this determinant of this matrix is maximized we will call that this W and W_i which are basically the columns of the W is the set of N dimensional Eigen vectors of scatter matrix S_t corresponding to m large n Eigen values we can check this proof in many books.

Including by so I m avoiding the proof here what this basically means is that if you look at the n dimensional Eigen vectors of ST, so there comes our decomposition based on Eigen analysis or as VT which we have to do we get the n dimensional Eigen vectors of ST using that if you construct the W and if you do that corresponding to the m largest Eigen vectors.

Then you will actually form your W which will be optimal to give you these maximizing criteria of scatter in the transform domain, so this is what we are maximizing so continuing the discussion.

(Refer Slide Time: 18:42)



Principal Component Analysis Contd.

- Eigenvectors are called eigen images/pictures and also basis images/facial basis for faces.

- Any data (say, face) can be reconstructed approximately as a weighted sum of a small collection of images that define a facial basis (eigen images) and a mean image of the face.

On the PCA before we get little bit more analytics okay, so a typically if you talk of Eigen vectors or Eigen images or certain Eigen the data of certain pictures and they are also called the basis images are facial basis functions if the example of images but remember the images could be of any other samples such as cars bicycles even humans let us say or buildings are any other scenario, but they should be typically samples belonging to the same class okay because for which we are trying to get.

These categories it is not that you take all these several images which are there in your gallery or you make a repository of your own from Google images you have and then start you can do a scatter and do a PC on that but actually in some sense that will be meaningless it is done for the particular set of samples belonging to a particular type of images may be of different faces of individuals different car models different bikes different humans okay it could be of different buildings.

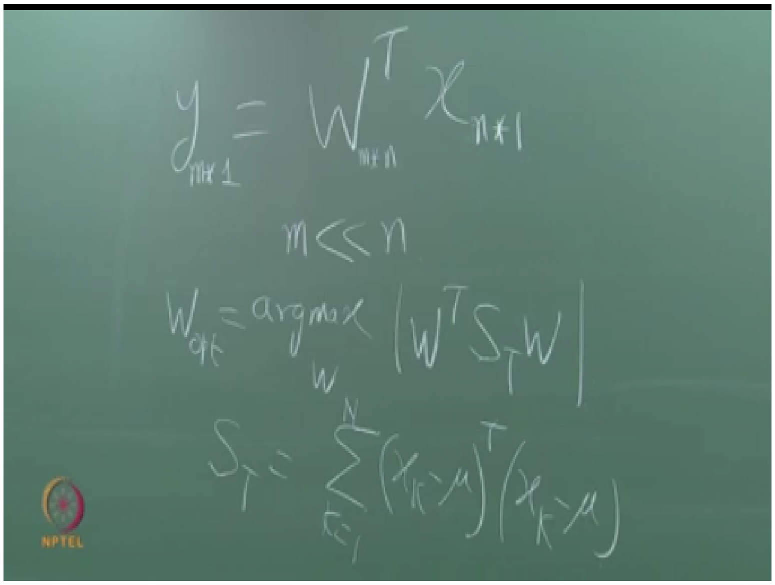
This well required or different type of box let us say which is as travel back okay, so in other some data says which have this different type of samples belonging to a set of similar classes, so you are talking about images so they form the basis vectors and weight data's example of face it can be constructed approximately as a weighted some of all the collections which is that define the facial basis or Eigen images and mean image all the face, so what it basically means the sentence is remember.

You had the set of samples x in some n dimension and assuming that you have Eigen analysis and find out what is the optimal W correct offer getting this W you project the samples x to y in a lower dimension from m to m or m is less than n so you get a samples y these form your basis images if the images are faces they are called the facial basis PC is a very common method used for doing some sort of the classification or Eigen clustering with facial images it was proposed by the in a paper if I am not wrong with the.

Date round 80s or may be late 70s by Turken Pin land PCA where it talks about Eigen faces then there of course there are other types of beautiful papers like bell hammer about official faces versus Eigen faces and all that so then PCA become very popular for representing faces of course there are better methods now to do facial classification of clustering okay, so faces an example but it could be again for any particular images so what you were doing is when you project the samples of the face.

Or any other particular type of data to a lower dimension y these form here basis images for representation in a lower dimension and you also have a mean image of the face which you cam compute from the original data that you can do remember we have done a mean subtraction before doing computing the scatter matrix, so scatter matrix itself you are in subtraction so can you get back your x given y can you get back your x given by remember what is y, y is W transpose multiplied by x may be I will write that expression on the back side of the board because that is very important projection which you are trying to do.

(Refer Slide Time: 22:09)



$$y_{m*1} = W^{T}_{m*n} x_{n*1}$$

$$m << n$$

$$W_{opt} = \arg\max_{W} \left| W^{T} S_{T} W \right|$$

$$S_{T} = \sum_{k=1}^{N} (x_{k} - \mu)^{T} (x_{k} - \mu)$$

Okay so we got that all of this equation so far this is what PCs is suppose to do projected from a higher dimension to a lower dimensions space with m less than and of course but sometimes it is taken much less than m W is obtained by a PCA which is basically solve this you know satisfy this criteria that you are it is going to maximize the scatter in the projected space in the space given by this y and this W is obtained from the Eigen vectors of this scatter matrix typically this is an un bias scatter matrix.

Sometime you many are normalizing termed here okay for and the corresponding terms in this scatter matrix is same as the covariance matrix and we will see how and SVT turn on the scatter matrix we will help us to obtain so there is a proof which I am skipping which will you this optimal and then using this Wu to the corresponding projection from x to y, so given this expressions now which we are just seen on the board and we also seen early in the slides, so you can get y from x the next question comes is can you get back from y S to some extent because y is representation it satisfying some criteria of maximum scatter along certain directions okay the choose the directions which of the maximum scatter that is all right okay and of course there are few other properties.

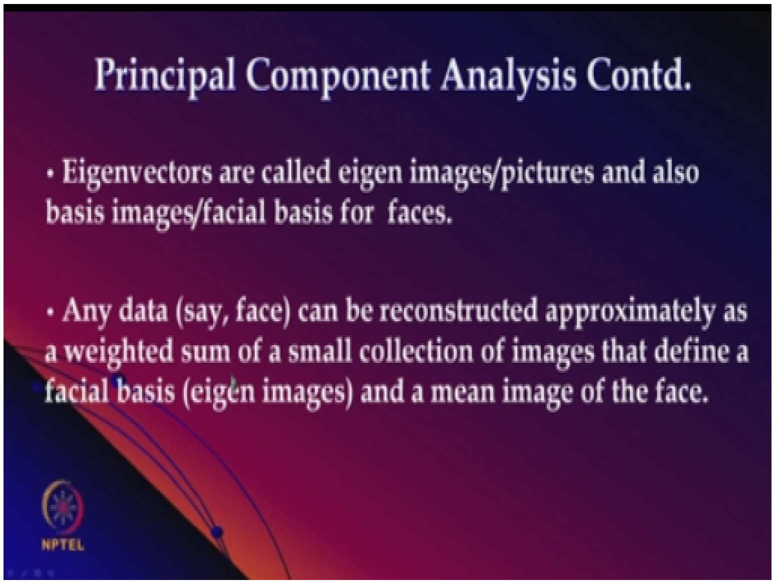
But that it is a main property which we have to consider now given those y which you are obtained by doing a projection using W can you get back x that is the biggest question you should be able to get back because W is an invertible matrix okay it is square it is meshed it is based on orthogonal components okay and if you choose a square W you should be able to invert it and you should be able to get back x by back projecting y on to the other dimension the question is only give back the original x of course.

The other thing which you keep need to keep in mind that you want to add the mean vector which you have subtracted from this I am just to get the original samples back, the reconstruction is complete if you keep m = n that means keep correspondingly all the direction are what are the called the Eigen vectors in y then you will get a perfect reconstruction back but in general we are looking for dimensional reduction also with the help of PCA, so depending upon the number of dimensions you use you will get.

Back a reconstructed signal x which will be a very close approximation of x but not the same x remind you that okay it may very close, so face images if you take a two PCA go to a lower dimension come back the faces may not exactly represent the same faces are look like the same data samples which you have started with in the original dimension it will look something different will try to give some examples in the next class of how what are called Eigen faces look like.

When you project the samples and then we construct back okay, so just remember this mine it is a low dimension representation it will give an approximation when you project it back and you must remember that we are talking about the mean image.

(Refer Slide Time: 26:51)

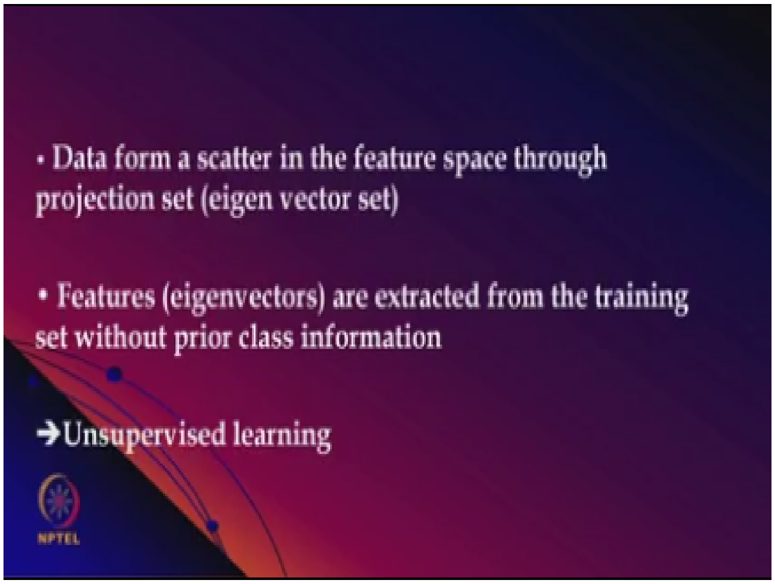


Principal Component Analysis Contd.

- Eigenvectors are called eigen images/pictures and also basis images/facial basis for faces.
- Any data (say, face) can be reconstructed approximately as a weighted sum of a small collection of images that define a facial basis (eigen images) and a mean image of the face.

So look at the last slide this as it has any data can be reconstructed approximately this is what I was trying to highlight as the weighted some of small collection of images that define the facial basis are Eigen images, so these are my voice so you representing x as a weighted some of small collection of images that define the facial basis or Eigen images so these are my voice so you representing x as a weighted sum of all these voice and the mean image of the face which you are subtracted to compute the scatter matrix, okay.

(Refer Slide Time: 27:12)



- Data form a scatter in the feature space through projection set (eigen vector set)

- Features (eigenvectors) are extracted from the training set without prior class information

→Unsupervised learning

So the data form is scatter in the feature space through the projection set which is called the Eigen vector set and the features that are extracted from the training set without prior class information, I do not have any class information available here hence it is also called a method of unsupervised learning it is like learning without a teacher we know the difference between supervised and unsupervised.

Supervised you will have class samples here you do not have class samples it is called unsupervised so some time that is why PCA is also called a method of clustering okay, it will not group the data as such but it will tend to form groups depending up on the direction of the maximum scatter, okay and it may try to discriminate between classes but you will be wondering I will show an example what this means.

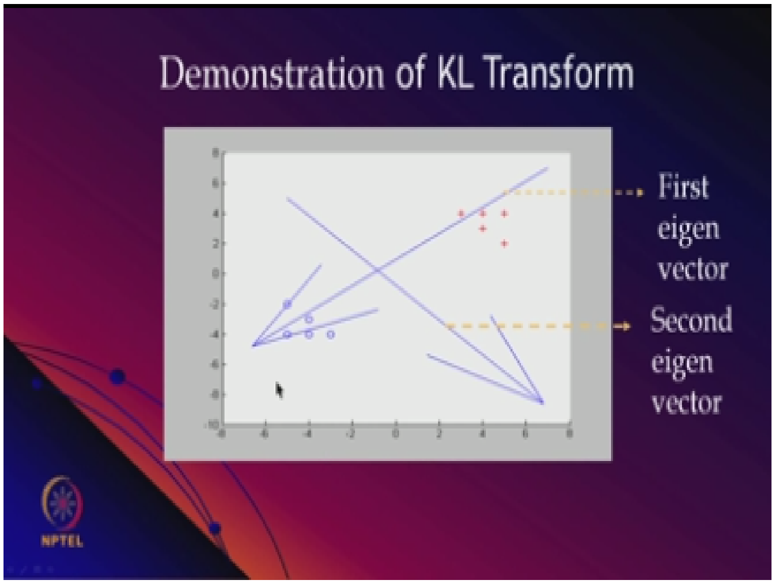
When the class information is not there what you will learn from the data that is the big question you can ask now, so you are talking of unsupervised learning and for learning you need to learn what are the class samples if that itself is not given to you what will you learn and what will be the output so we will see that with examples that the main purpose of PCA is actually to extract out and give a low dimensional letter which satisfies some criteria of maximum scatter along certain set of dimensions.

Or the other applications is dimension reduction in terms of representation also, in some cases this is driven towards a method of unsupervised learning okay, with the hope that if you are extracting dimensions along maximum scatter along the direction of maximum scatter to be very precise. The data will form groups along with their clusters or they will form groups along the direction of maximum scatter as per the number of class.

So if we have two or three different classes they tend to form groups along those direction and with that hope we can call PCA very loosely as a method of unsupervised learning although you must remember that is not the main aim that is only a secondary though and an application of PCA which one can use actually you cannot learn anything because in the true sense of the terms in the field of pattern recognition.

Because the class samples are not known, but anyway we will keep this is mind that the features are extracted without prior class information and hence it is also call unsupervised learning.

(Refer Slide Time: 29:43)



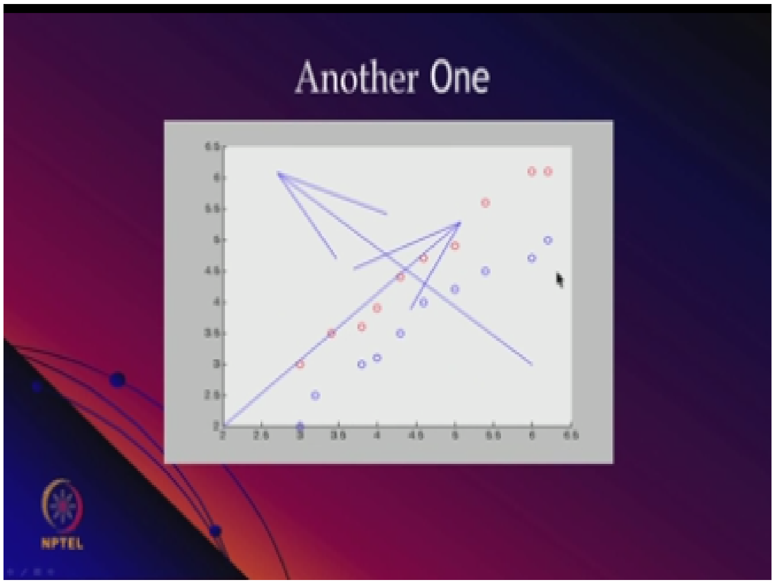
Demonstration of KL Transform

This is an example which shows what this scatter is, look at the set of samples here so you have a set of samples indicated by certain color and a symbol here you have another class second group indicated by another set of symbols with different color, and if this is a two dimensional example in 2D what you expect PCA to give is the first Eigen vector will give a direction as given by these arrow which will say this is the direction and which I have the maximum scatter or the maximum spread or the maximum width or the maximum separation.

Scatter, spread, separation whatever term you want to use to give it a meaning, but we will use the word scatter hence forth which could mean separation or spread as well okay, in that direction you can see that is the direction which is given the maximum scatter. There is orthogonal direction in fact you have to the figure does not reveal you but this direction is orthogonal to the first Eigen vectors so these two vectors are orthogonal to each other. The second Eigen vector where you have the second highest or less scatter compared to the first one.

And remember since the two dimensional problem so you will have only two directions in which you can project so you can basically or projecting into one direction and which you have the first Eigen vector giving you the maximum scatter.

(Refer Slide Time: 31:05)

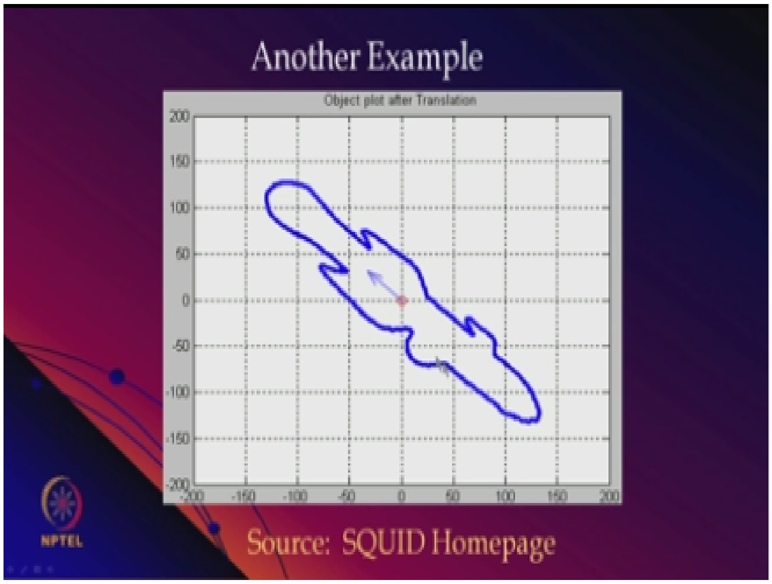


Another One

This is another example so this is an interesting example where it shows that the PCA is not able to produce direction which we will separate the two classes if you look at this, if you look around the first Eigen vector for the first example and project the samples on to this particular axis this will form a clutter here, this will form a cluster here and it should help you to do some unsupervised method of clustering or classification, okay.

This is not the case anymore you look at the direction of the maximum scatter which is this one okay, along this if you project all the samples there will be huge extend overlap. In fact the second direction which will actually give you the separation.

(Refer Slide Time: 31:49)



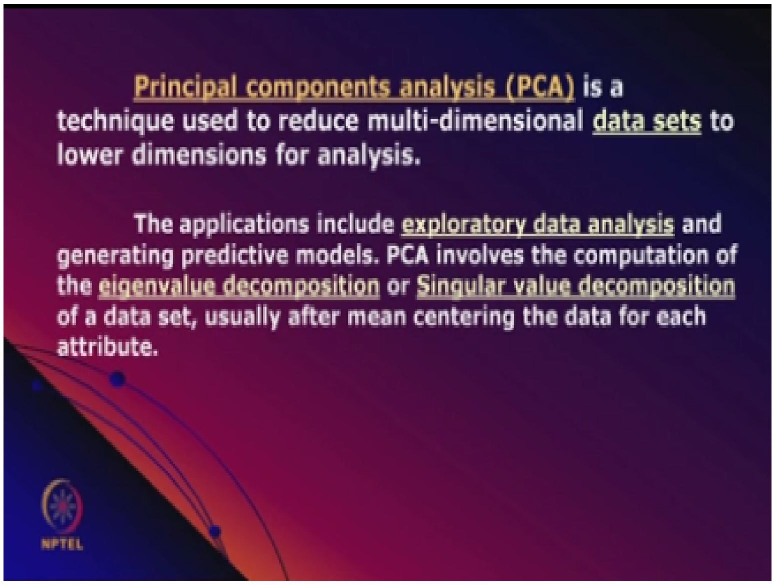
Another Example

Source: SQUID Homepage

This is another example using an image set of contour two points if you take x and y as coordinates if you take x and y as coordinates and do a PCA this is the direction in which you will get the plot, this is an example from the squid home page for object shapes you will get the maximum scatter along this particular direction of the set of x and y. What are the samples here, the data is 2D but the samples are x and y coordinates.

So x and y one object point of the contour is one sample, second sample is x to y to the next point and so on, so you will set of n samples take them do a PCA it will you give some direction in 2D the first Eigen vector will give you the maximum scatter that is what this will give you. So this illustrations are now showing you what is the direction of the maximum scatter or spread or width of the data. Let us go to some analytics of the PCA now.

(Refer Slide Time: 32:37)



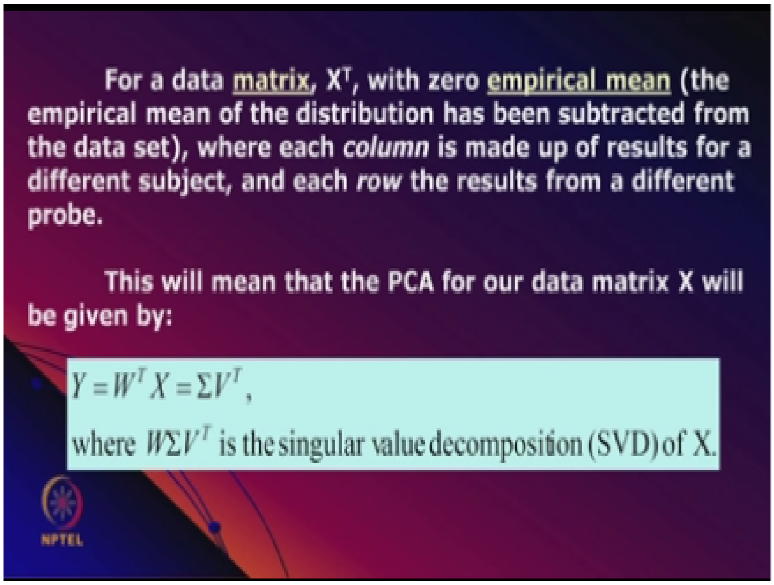
Principal components analysis (PCA) is a technique used to reduce multi-dimensional data sets to lower dimensions for analysis.

The applications include exploratory data analysis and generating predictive models. PCA involves the computation of the eigenvalue decomposition or Singular value decomposition of a data set, usually after mean centering the data for each attribute.

And before that I will just give you some inputs which people use you in terms of interpreting what is PCA it is also called a technique used to reduce multi dimensional data sets to a lower dimensions for analysis, we already talked about this that it is a problem of it is also considered as a tool or a mechanism to produce a low dimensional data or dimensionality reduction. The application can be used for predictive analysis models exploratory data analysis.

It involves the computation of Eigen value to composition or as VD we talked about this of a data set usually after means entering the data for each attribute we talked about subtracting the mean of the data anyway.

(Refer Slide Time: 33:19)



For a data matrix, X^T, with zero empirical mean (the empirical mean of the distribution has been subtracted from the data set), where each *column* is made up of results for a different subject, and each *row* the results from a different probe.

This will mean that the PCA for our data matrix X will be given by:

$$Y = W^T X = \Sigma V^T,$$

where $W\Sigma V^T$ is the singular value decomposition (SVD) of X.

For a data matrix X^T you know that is only a data notation where that means all the samples put together from that matrix X with 0 empirical mean because you are subtracting the mean, that the 0 empirical mean will be there anywhere. The empirical mean of the distribution has been subtracted from the data set that is what mean by the empirical mean.

Each column is made up of results from a different subject and each row the results from a different probe, these are terminologies what it basically means is that if X is a matrix indicating all the data samples put together then each particular column is basically indicating a sample point and each particular row is indicating a dimension, that means if you take the previous example object of contour points on that object which we just shown an indicated one directional the PCA.

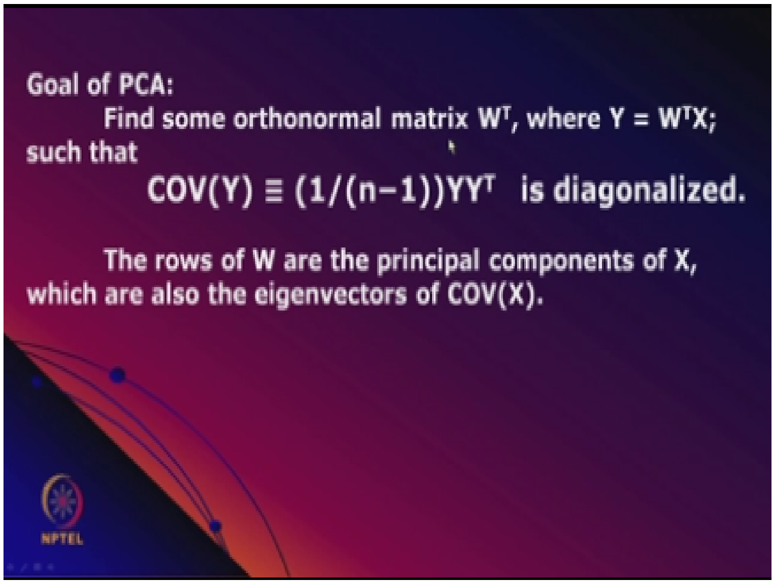
The matrix X will be of dimension to then there may two rows multiplied by n as many number of contour points which you have, okay. So this is only a notation sometimes these are called subjects and probes or samples and probe. Probe means picking a feature okay, so each probe picks up a particular value as a feature okay, so that is a terminology which is sometime used okay.

Say so in this particular case for images each gravel value or a pixel itself could be a probe it is an observation which has been made by a sensor so that could be a sample point as well. And this will mean when you are doing this on X which is our data that means the PCA of the data matrix will be given by remember you are taking the X and you are trying to actually project on to Y using a X^T and it is also given by this matrix where the SVD of X.

You can write, if you write the SVD of X as something like this okay, so you replace X by WΣ the diagonal matrix and another set of Eigen vectors V we substitute here, where W^TW it is an orthogonal matrix it will vanish and you will be left with this. So this is another representation but typical a most people use this representation, but this is the if X you take the SVD of X and obtain this.

So basically what it means is W is obtained by the SVD of X, and it is the left set of Eigen vectors because you have two sets of Eigen vectors W and V.

(Refer Slide Time: 35:55)



Goal of PCA:

Find some orthonormal matrix W^T, where $Y = W^TX$; such that

$$COV(Y) \equiv (1/(n-1))YY^T \text{ is diagonalized.}$$

The rows of W are the principal components of X, which are also the eigenvectors of COV(X).

So the goal the ortho normal matrix W is to actually find this W out from the scatter matrix X and then used to project Y.

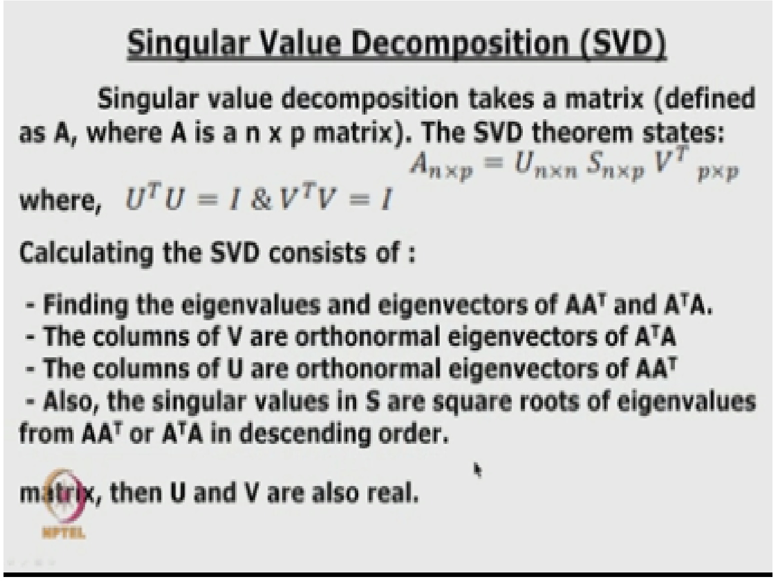
And is such that the co variants of y which is also given by this now we have a normalizing term y y^T this is diagonals okay so if you take the co variants of the w scatter of w you will have only diagonal terms that is the basic gain and the rows of the w are basically the which is the basically the matrix used for projection or pc or the principle components of x and they are also call the Eigen vectors of co variants of x.

So unlike other transform the other types of transform which also exist in literature like the descript cosign transform DCT descript Fourier transform DFT descript well a transform DWT Haddam hot transform other types of transform which exist also in literature including in the field of signal processing matrix algebra TF communications and so on, PCA does not have a fixed of basis vectors it depends on the samples remember w is composed set of Eigen vectors you derive them from the data.

What is the data sample here? The capital x matrix that means if you change x, how can I change x? Very simple takes of a few samples points or add a few sample points change a few set of samples we have different x if you have a different x you take the SVD of that you get a different w you get a different matrix w and you will have a different set of projections now okay.

So the Eigen vectors which are responsible for the radial dimensions it depends on the data itself which is different than the other type of transform which is exist literature for other type of application remember DCT DFT has major applications in many branches of science and engineering as well as descript develop transform for the case of multi resolution signal analysis so they have basically set of fixed vectors or representation matrices based on which you do the projection. But here it is it fell to data take manner.

(Refer Slide Time: 38:03)



Singular Value Decomposition (SVD)

Singular value decomposition takes a matrix (defined as A, where A is a n x p matrix). The SVD theorem states:

$$A_{n\times p} = U_{n\times n} S_{n\times p} V^{T}{}_{p\times p}$$

where, $U^TU = I \,\&\, V^TV = I$

Calculating the SVD consists of :

- Finding the eigenvalues and eigenvectors of AA^T and A^TA.
- The columns of V are orthonormal eigenvectors of A^TA
- The columns of U are orthonormal eigenvectors of AA^T
- Also, the singular values in S are square roots of eigenvalues from AA^T or A^TA in descending order.

matrix, then U and V are also real.

And since we have studied so far that SVD is the main idea behind performing the PCA is based on SVD we just have one slide here which indicates or gives you a revision of singular vault decomposition if you are actually have donna skip of the discussion which was done earlier in the class okay in one class when we discussed vector algebra and spaces.

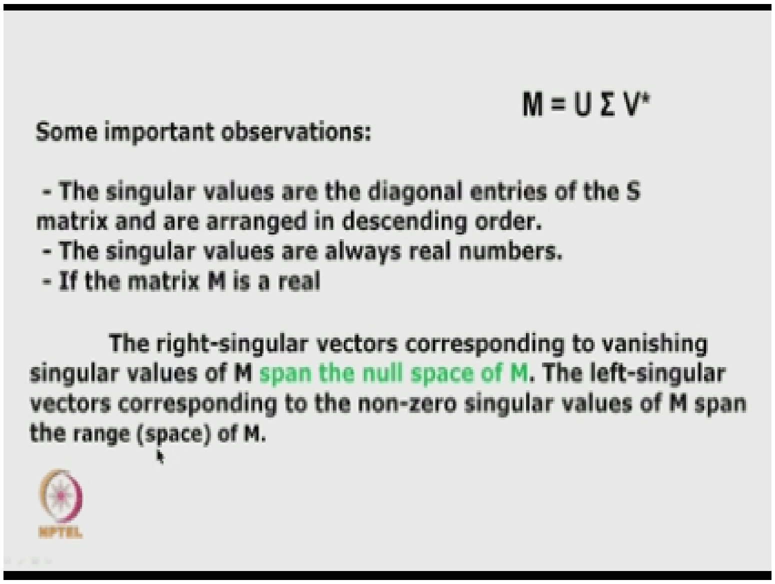
So SVD takes symmetric a of arbitrary size m x p and the theorem says that you can decompose a x a unitary matrix w and another matrix t v here both are orthogonal matrices and a diagonal matrices s given by this the calculation of SVD consist of this is just key point if this is not lecture on SVD this is a key point it basically it finds if Eigen values in s and the Eigen vectors of a a^T which will get in the u and v respectively.

So what it means the columns of v are Eigen vectors of a^T a that is what you will get in v which is here and the columns of u which is this matrix are the orthogonal Eigen vectors of aa^T okay, so aa^T is basically this matrix a, so you can form an aa^T out of a and you are talking about the Eigen vectors of that particular matrix which will be symmetric then you are talking about those Eigen vectors as u and that will form my w in fact w is taken out of u okay.

And also the singular values in s or sometimes is in a σ in some notations you will find in my slides also that this is written as a σ matrix are the square roots of Eigen values of either a transpose a or a^T a but interesting is that it will be arrange in descending order that means if is a diagonal matrix the top left diagonal element you will have the largest Eigen value then the second value will have the second largest Eigen value and so on up to the least Eigen value at the bottom okay.

At the bottom most diagonal element will have the smallest Eigen value okay in descending order and the u and v are also real matrices.

(Refer Slide Time: 40:11)



$M = U \Sigma V^*$

Some important observations:

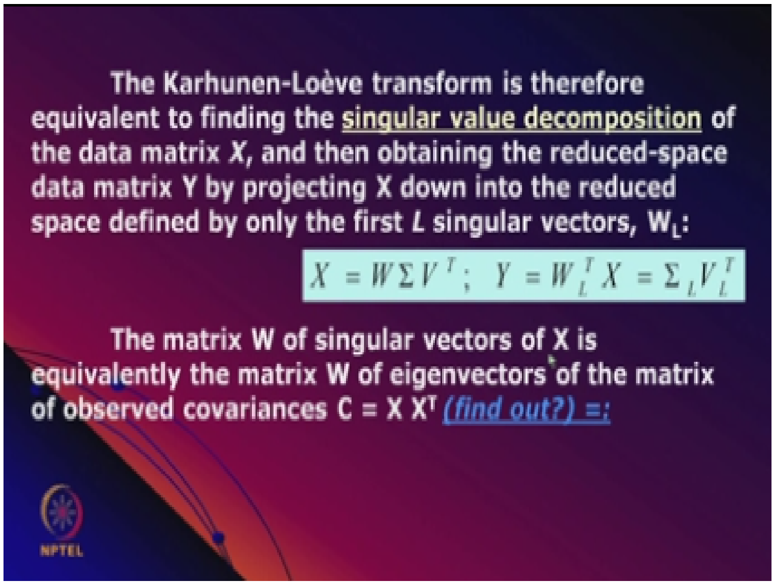
- The singular values are the diagonal entries of the S matrix and are arranged in descending order.
- The singular values are always real numbers.
- If the matrix M is a real

The right-singular vectors corresponding to vanishing singular values of M span the null space of M. The left-singular vectors corresponding to the non-zero singular values of M span the range (space) of M.

Some important observations of SVD the singular values are the diagonal entries of the s matrix and arranged in descending order which we know and they are always real numbers and the matrix is also very real but sometimes some books will use matrix m, this is the same as matrix which we have seen earlier the right singular vectors which are corresponding to the v the corresponds to the vanishing singular values of m, so if you have some null Eigen values of in this σ.

They form what is call the null space of m and the left singular Eigen vectors of which are there in the u the correspond to the non 0 singular values of m and the span the range space of m. We may not use the concept of range space and null space right now in our discussion but it will come in our discussion in the next when we discuss LDA after PCA.

(Refer Slide Time: 41:07)



The Karhunen-Loève transform is therefore equivalent to finding the singular value decomposition of the data matrix *X*, and then obtaining the reduced-space data matrix Y by projecting X down into the reduced space defined by only the first *L* singular vectors, W_L:

$$X = W\Sigma V^T; \quad Y = W_L^T X = \Sigma_L V_L^T$$

The matrix W of singular vectors of X is equivalently the matrix W of eigenvectors of the matrix of observed covariances $C = X X^T$ *(find out?) =:*

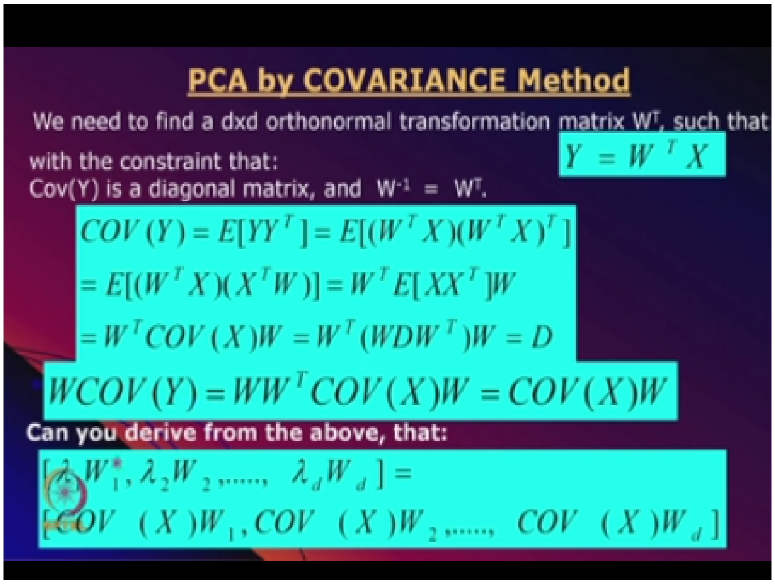
Okay so carrying on the discussion of PCA which is also call the Karhunen – Loeve transform or KLT it is equal to finding this SVD of a particular data matrix x, we know that and then obtaining the reduced space of the data matrix y by projecting x we have seen that, so x is basically w y is basically w^T multiplied by x, with the reduce space define by of the l singular Eigen vectors this l is our m remember so representing that.

So you can actually chose a few set of singular vectors not the entire dimension and assuming that the SVD gives you this and if w you have seen this expression a few slides back where y can be represented by this w is basically the left Eigen vectors corresponding this SVD of x or you can write them in this particular form as well okay.

So you can find out that the matrix w of single vector x is equal to the matrix w of the Eigen vector of the matrix of the observed covariance's xx^T so if you look back to this slide here so we are talking about covariance x^T and it is given by this particular expression so if you substitute x as this and x^T will be basically $v\sigma^T$ and substitute that here it is easy for you to see that V^T will cancel out.

And you will be left with this which can written in terms of this where d is the diagonal matrix again which is the square of the singular values which you get here on the σ so the covariance of x which you get of will be given by this so this is again a diagonal matrix and you have the left singular angle vector of w on both sides.

(Refer Slide Time: 42:49)



PCA by COVARIANCE Method

We need to find a dxd orthonormal transformation matrix W^T, such that

$$Y = W^T X$$

with the constraint that:
Cov(Y) is a diagonal matrix, and $W^{-1} = W^T$.

$$COV(Y) = E[YY^T] = E[(W^T X)(W^T X)^T]$$
$$= E[(W^T X)(X^T W)] = W^T E[XX^T]W$$
$$= W^T COV(X)W = W^T(WDW^T)W = D$$

$$WCOV(Y) = WW^T COV(X)W = COV(X)W$$

Can you derive from the above, that:

$$[\lambda_1 W_1, \lambda_2 W_2, \ldots\ldots, \lambda_d W_d] =$$
$$[COV(X)W_1, COV(X)W_2, \ldots\ldots, COV(X)W_d]$$

So carrying on the discussion by the PCA what is mean by the covariance matrix which we started with discussion so we assume that the w matrix is of certain dimension dxd such that we had this constraint and the covariance matrix is a diagonal matrix of y with W^T which can be proven by this so if you look at the expression of the derivation here the covariance of y is expectation of yy^T substitute y $=w^T$ x in the expression here I repeat gain take $w=W^TX$ substitute it here this is what you will get okay.

Then you get this and you take out the matrix w out of the expectation term here this will give you the covariance we have derived that to be wet matrix which is d inside and so that interesting that after all you have covariance of y which is obtained remember the voice samples are obtained by doing a PC on x which yield a covariance which will be strictly diagonal what does it basically need means.

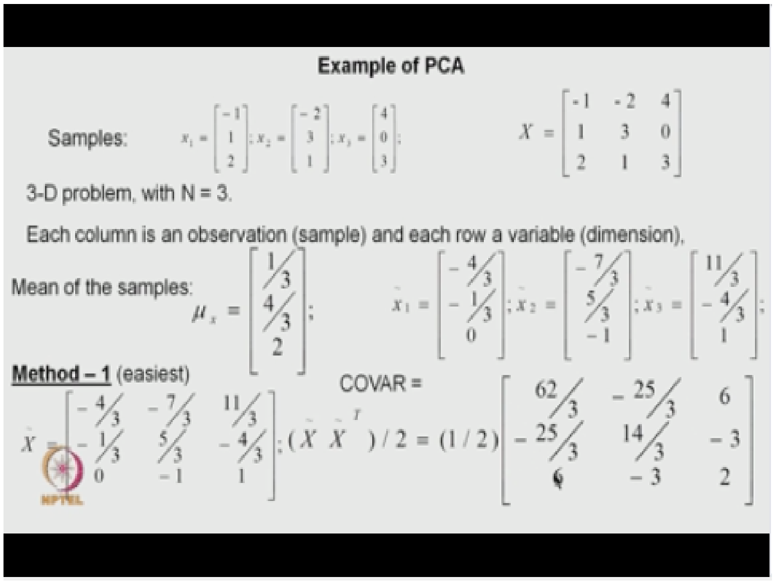
That you have the what is the d is the diagonal matrix which is the square of the Eigen values of the diagonal matrix obtained by the SVD okay and you then have the scatter of the samples of y which is maximum along the first direction because this is also happier in descending order this will appear in descending order and you will have the maximum scatter along the first dimension the second maximum dimension

And so on and that is what covariance of y indicates and if the covariance is strictly diagonal what does it mean half diagonal terms are should be zero the covariance matrix is strictly diagonal then the hold diagonal terms are zero that means what are the diagonal terms σij being 0 that means there is no covariance no relationship between the two dimensions because they are orthogonal to each other okay and they are correlation becomes 0 means that 1 data is completely independent of the other one, so you are projecting on to certain dimensions not only you have maximum scatter along in 1st and the 2nd and so on you also do not have any correlation between those 2 dimensions you have projected on 2 dimensions feature space where the dimensions are independent of the other.

1st is independent of 2nd and 3rd so on the same is to with the 2nd independent as well as the other one, so if you take any 2 dimensions I and j after pc in a transform domain 2 arbitrary dimensions a and j I0 = j they will not have any correlation and that is what I defecated in the co variance of matrix y which will give you a diagonal matrix that is what you get here, you can also prove this W x COV Y = COV /W should actually you can derive from that the corresponding singular values x corresponding weights.

Is a simple analysis prove I will leave this is an example for an assignment to derive this because the co variance matrix is diagonal that is what it will give you this particular matrix.

(Refer Slide Time: 46:29)



Example of PCA

Samples: $x_1 = \begin{bmatrix} -1 \\ 1 \\ 2 \end{bmatrix}; x_2 = \begin{bmatrix} -2 \\ 3 \\ 1 \end{bmatrix}; x_3 = \begin{bmatrix} 4 \\ 0 \\ 3 \end{bmatrix};$ $X = \begin{bmatrix} -1 & -2 & 4 \\ 1 & 3 & 0 \\ 2 & 1 & 3 \end{bmatrix}$

3-D problem, with N = 3.

Each column is an observation (sample) and each row a variable (dimension),

Mean of the samples: $\mu_x = \begin{bmatrix} 1/3 \\ 4/3 \\ 2 \end{bmatrix};$ $\tilde{x}_1 = \begin{bmatrix} -4/3 \\ -1/3 \\ 0 \end{bmatrix}; \tilde{x}_2 = \begin{bmatrix} -7/3 \\ 5/3 \\ -1 \end{bmatrix}; \tilde{x}_3 = \begin{bmatrix} 11/3 \\ -4/3 \\ 1 \end{bmatrix};$

Method – 1 (easiest)

$\tilde{X} = \begin{bmatrix} -4/3 & -7/3 & 11/3 \\ -1/3 & 5/3 & -4/3 \\ 0 & -1 & 1 \end{bmatrix};$ COVAR = $(\tilde{X}\tilde{X}^T)/2 = (1/2)\begin{bmatrix} 62/3 & -25/3 & 6 \\ -25/3 & 14/3 & -3 \\ 6 & -3 & 2 \end{bmatrix}$

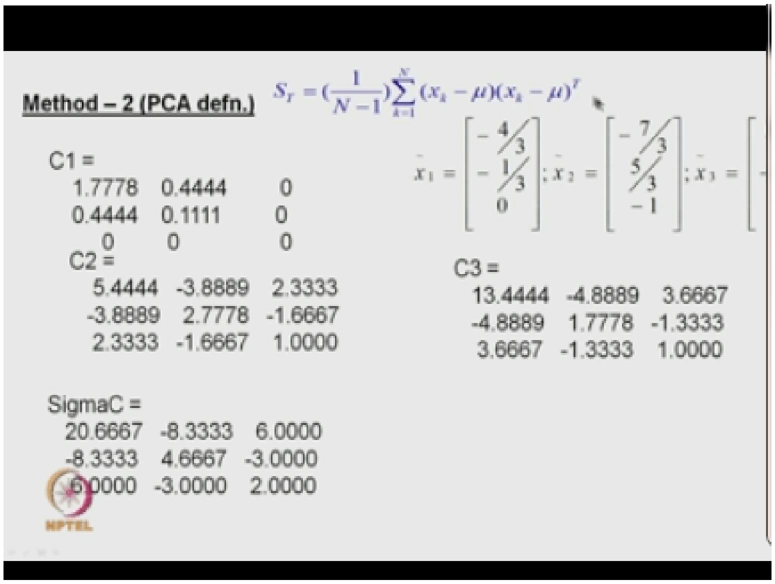
Let us take an example of a PCA a hand worked out toy example, you take an example of 2d because sorry the data is 3d just 3 sample points of course you can take many and this is the overall data sample. You can say this is a column and gives you this x, so what is the job of PCA how do you do PC? Before s variance you must not forget 1 important step which is to = the mean of the data.

So let us calculate the mean of the data, the mean of the data is this particular sample, so it is 3d problem number of samples is also 3 and each column is an observation sample and each row is the dimension, this is the mean of the samples. So that is the sample mean μ is 1/3rd what is how do you get the value of 1/3 + theses 3 okay this is the 1st dimension what is 4 -1, -2 1/3 that is 1/3 or 4/3 3+ 1 = 4/ 3.

This is 2+ 1 3 6/3 that is very simple that is how you calculate the mean – samples that means you take the each of these individual samples x1, x2, x3 – mean this is what you get I will leave this as an exercise to check it out, this is also same as taking each of the column and – vector. now you construct the SVD create the sample x okay so the data sample x is become this, how do you get this take these 3 columns one after another and from this new x.

Let us look at the co variance term this is xt /2 remember this 1/n-1 so that is how 2 comes and I am giving you the answer here you can actually user any mathematical tool box to compute the it is 3 x 3 matrix it will not be difficult for you to actually x and calculate x t which you will getting at this. so you have done the co variance of matrix for the scatter as it is called and we will do SVD of this scatter matrix in the next slide.

(Refer Slide Time: 48:51)



Method – 2 (PCA defn.) $S_T = (\frac{1}{N-1})\sum_{k=1}^{N}(x_k-\mu)(x_k-\mu)^T$

$\bar{x}_1 = \begin{bmatrix} -4/3 \\ -1/3 \\ 0 \end{bmatrix}; \bar{x}_2 = \begin{bmatrix} -7/3 \\ 5/3 \\ -1 \end{bmatrix}; \bar{x}_3 =$

```
C1 =
   1.7778   0.4444        0
   0.4444   0.1111        0
        0        0        0
C2 =
    5.4444  -3.8889   2.3333
   -3.8889   2.7778  -1.6667
    2.3333  -1.6667   1.0000

SigmaC =
  20.6667  -8.3333   6.0000
  -8.3333   4.6667  -3.0000
   6.0000  -3.0000   2.0000
```

```
C3 =
  13.4444  -4.8889   3.6667
  -4.8889   1.7778  -1.3333
   3.6667  -1.3333   1.0000
```

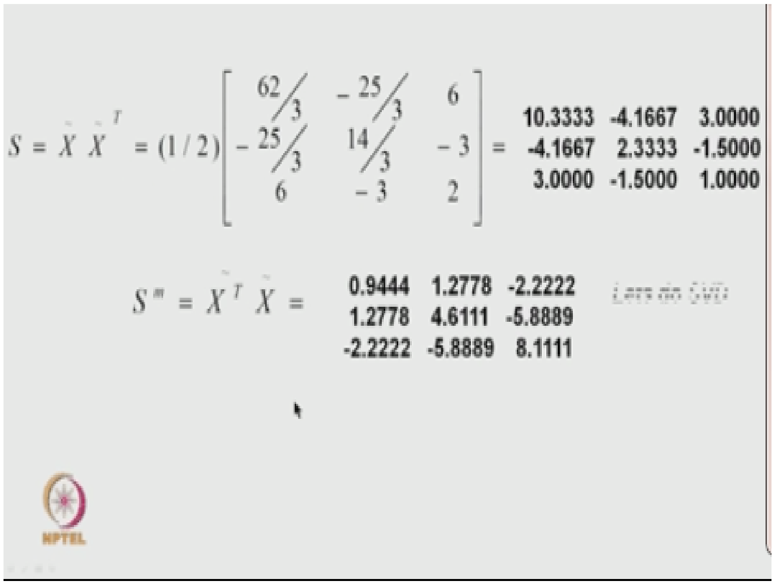
So this is what we have done so far this is the scatter matrix example okay you get the mean subtracted data and you can actually compute the scatter matrix by individual standings of this, if you actually follow this expression to compute this scatter matrix from these data samples – mean and construct what is C1 xt of this.

So the 1st term of this expression x1 – μ, is this compute this expression outer product is this similarly for x2 you get the C2 and similarly for x3 you get the C3. I will repeat again 1 st term of ∑ you get this C1, 2nd ∑ you get this and x3 for the 3rd term of this k = n to 1, 3 terms c1 from x1, c2 from x2, c3 from x3 some all of these are what you will get. You can check few examples if you want 1 + 1 + 0 how do you get -3, if you some of these two elements are up – 5/3 this is – 4/3.

So it will basically give – n/3 to -3 at these two elements are you will get 6 also, same thing i applicable for last row as well t this two element is -3, so that is very simple and then you get the co variance which is σ/2 which is this and this I repeat is a same co variance which you obtained by the σ, so get look back this is what you got. Look at the 1st term it is 62/6 just look at the last two it is easy to verify here it is symmetric matrix, so this will be 3 – 3/21.

So check whether this value is correct now, 3 – 1.51 same as this, so you just have to divide by 2 you get the answer so you can compute this scatter matrix either using expression or co variance one so it will give the same.

(Refer Slide Time: 51:24)



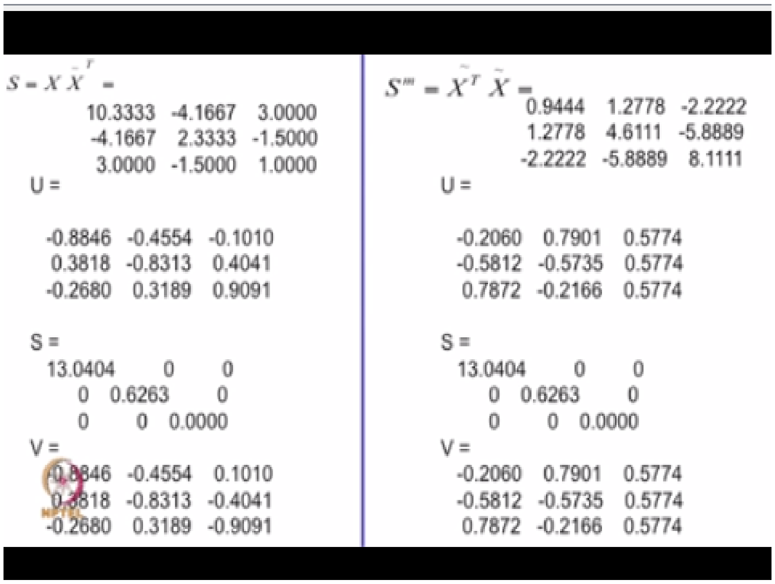
$$S = \tilde{X}\,\tilde{X}^{T} = (1/2)\begin{bmatrix} 62/3 & -25/3 & 6 \\ -25/3 & 14/3 & -3 \\ 6 & -3 & 2 \end{bmatrix} = \begin{matrix} 10.3333 & -4.1667 & 3.0000 \\ -4.1667 & 2.3333 & -1.5000 \\ 3.0000 & -1.5000 & 1.0000 \end{matrix}$$

$$S^{m} = \tilde{X}^{T}\,\tilde{X} = \begin{matrix} 0.9444 & 1.2778 & -2.2222 \\ 1.2778 & 4.6111 & -5.8889 \\ -2.2222 & -5.8889 & 8.1111 \end{matrix}$$

Lets do SVD

So this what the scatter matrix looks like in the last slide this is what we have and if we do scatter matrix given by this and sometimes people take xt you will get the matrix this is done sometimes as dimension is too high and that is to both.

(Refer Slide Time: 51:52)



$S = \tilde{X}\tilde{X}^T =$

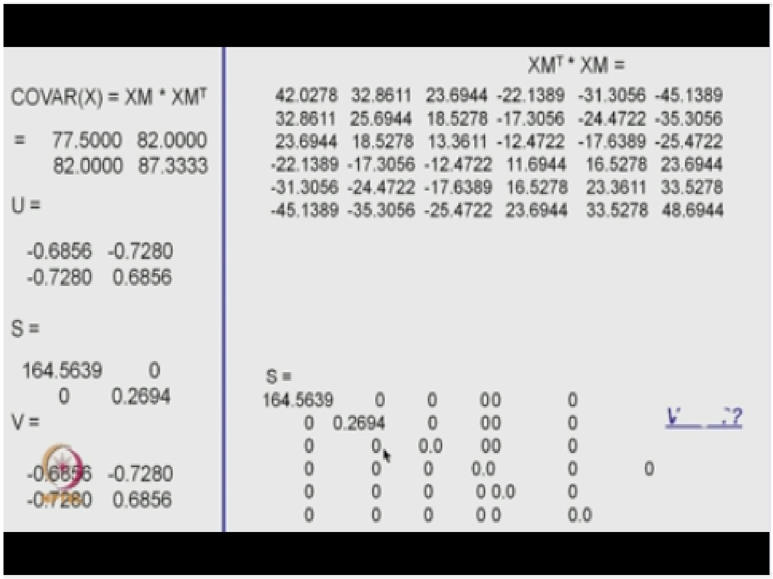
```
        10.3333  -4.1667   3.0000
        -4.1667   2.3333  -1.5000
         3.0000  -1.5000   1.0000
U =

   -0.8846  -0.4554  -0.1010
    0.3818  -0.8313   0.4041
   -0.2680   0.3189   0.9091

S =
   13.0404        0        0
         0   0.6263        0
         0        0   0.0000
V =
   -0.8846  -0.4554   0.1010
    0.3818  -0.8313  -0.4041
   -0.2680   0.3189  -0.9091
```

$S^m = \tilde{X}^T\tilde{X} =$

```
         0.9444   1.2778  -2.2222
         1.2778   4.6111  -5.8889
        -2.2222  -5.8889   8.1111
U =

   -0.2060   0.7901   0.5774
   -0.5812  -0.5735   0.5774
    0.7872  -0.2166   0.5774

S =
   13.0404        0        0
         0   0.6263        0
         0        0   0.0000
V =
   -0.2060   0.7901   0.5774
   -0.5812  -0.5735   0.5774
    0.7872  -0.2166   0.5774
```

So I am showing this x let us look at this is the actual one which has been proposed as a matrix but this is just a variant and let us look at if the examples are same, and look at u both the matrix but look at the diagonal matrix, you can either take xt or tx and co variance scatter in whatever way may be the diagonal matrix will give you the set of values. The vectors will not look the same so you must actually be very careful that you take the look into this slide x xt.

(Refer Slide Time: 52:50)

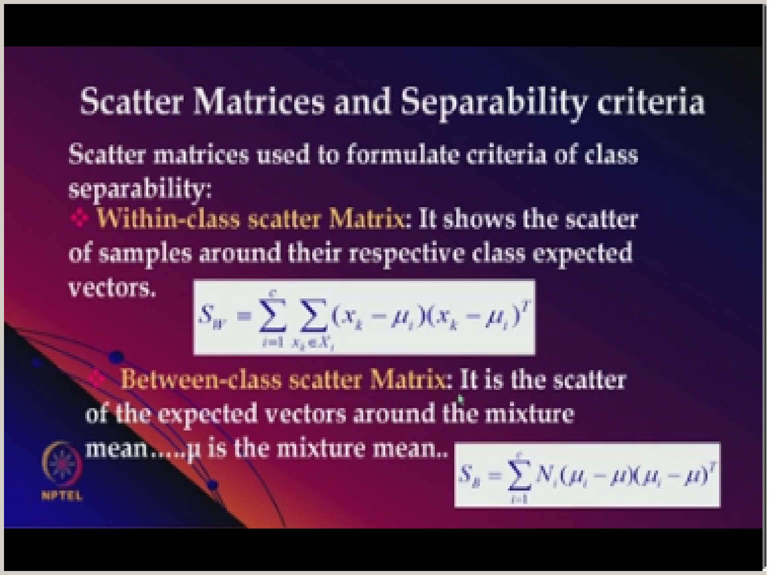


```
COVAR(X) = XM * XM^T                  XM^T * XM =

=    77.5000  82.0000          42.0278  32.8611  23.6944 -22.1389  -31.3056 -45.1389
     82.0000  87.3333          32.8611  25.6944  18.5278 -17.3056  -24.4722 -35.3056
                               23.6944  18.5278  13.3611 -12.4722  -17.6389 -25.4722
U =                           -22.1389 -17.3056 -12.4722  11.6944   16.5278  23.6944
                              -31.3056 -24.4722 -17.6389  16.5278   23.3611  33.5278
  -0.6856  -0.7280            -45.1389 -35.3056 -25.4722  23.6944   33.5278  48.6944
  -0.7280   0.6856

S =

 164.5639      0               S =
      0      0.2694            164.5639      0        0      00         0
V =                                 0    0.2694      0      00         0
                                    0        0      0.0     00         0
  -0.6856  -0.7280                  0        0        0     0.0        0        0
  -0.7280   0.6856                  0        0        0     0 0.0      0
                                    0        0        0     0 0        0.0
```

This is another example okay this is 6 points they are arranged to create this x so quickly run to this example 2d problem in 2d with 6 sample points again each column is on observation, this is the mean vector 1st row / 6 will give you this, this is what you have this is the mean subtracted value of this, that means you take all the sample point of this, that means you take all the sample points / corresponding vector.

This is the co variance xt something different, but you can also do this x^T x this is the correct one again the co variance of x^T x as well subtracted and let us look at u s and v, but look at s here as well okay. Look at the 1st wagon vectors, the wagon are same okay, the v will be different you can also see here that this being the v=u here because this is added to the symmetric matrix this is also symmetric this is also symmetric matrix so in that case you can use the Eigen vectors from any one of these

(Refer Slide Time: 54:25)



Scatter Matrices and Separability criteria

Scatter matrices used to formulate criteria of class separability:

- **Within-class scatter Matrix:** It shows the scatter of samples around their respective class expected vectors.

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

- **Between-class scatter Matrix:** It is the scatter of the expected vectors around the mixture mean.....µ is the mixture mean..

$$S_B = \sum_{i=1}^{c} N_i (\mu_i - \mu)(\mu_i - \mu)^T$$

So that is what to do with PCA there are other types of variances of scatter matrix which are also available which are also available in the literature and we will look into those and see examples of another type of Eigen analysis and decomposition which is possible and then compare both with an example where you look at this scatter just look at a scatter matrix which is called within class scatter metric.

This expression is now little different then what ever we have seen so far we had a scatter matrix where in probably we had subscribe w we had one σ and xk-µxk-µ^{T} what is different in this expression will you see the mean has an index I for a particular class okay and this index I transform 1 to c number it seems this scatter matrix now requires that you have samples belonging to each and every class separately grouped to compute.

This within class scatter matrix that means what is the scatter within a particular class that means you must know samples which belonging to a particular class that means if you have pictures of ten different persons of face images or what are called facial image samples then for each particular class or person in this particular case you must root this samples belonging to the particular sample and then compute with the expression.

Because you cannot compute µi else you know the samples which belong to the ith class so xk are samples belong to the class xi so the class level must be given with this so I am not doing a peace A you are doing some else here which will describe soon so you sum them over all classes and for each class you take the sample belong to the particular class particular class xi and compute the scatter the rest of it is same once you have the class mean.

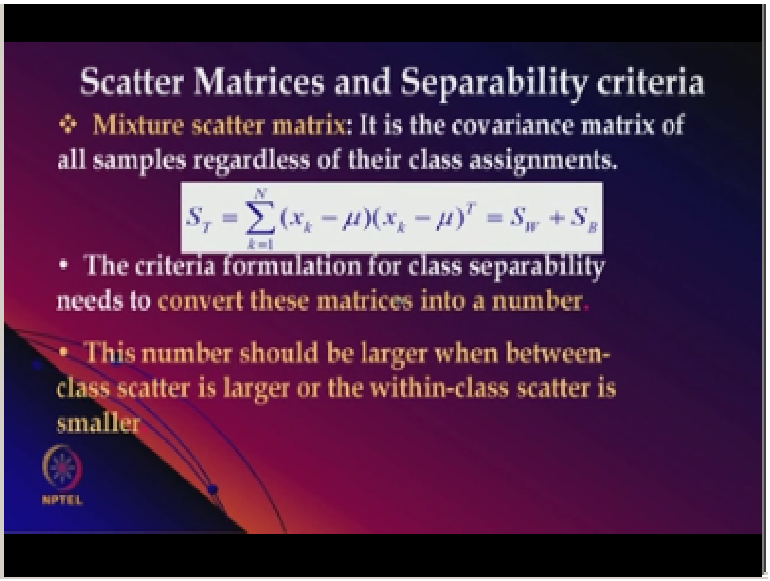
The greater sample xk belong to the particular class we can compute the within class scatter matrix the W here indicates not the wet matrix the within class is incorrect for the similarly you have between class scatter matrix it is the scatter of the expected vectors around the mixture mean of the entire mixture look at this expression now you have the overall data mean which you have used for PC earlier you also have individual class means.

And using these means you form this expression which is an outer scatter of the class outer product of the means multiplied by the number of samples summation over all classes so you have within class scatter matrix with data samples you have between class scatter matrix which is mainly dependent on the class means and sometimes causally people call that inter class and interact class scatters.

The inter class scatter basically means within class scatter interclass scatter basically means between classes what is the scatter okay so if you have this expressions computed as SW and SB it has certain very nice properties I will be able to discuss all of them due to limitations of time so what is within class scatter it shows.

The scatter of samples around the respective class expected vector class means so around particular mean what is the scatter here how the class means us scatter that is the second term SB the b stands for between will keep this notation of W indicating within class scatter b indicating between class scatter.

(Refer Slide Time: 58:29)



Scatter Matrices and Separability criteria

- Mixture scatter matrix: It is the covariance matrix of all samples regardless of their class assignments.

$$S_T = \sum_{k=1}^{N}(x_k - \mu)(x_k - \mu)^T = S_W + S_B$$

- The criteria formulation for class separability needs to convert these matrices into a number.
- This number should be larger when between-class scatter is larger or the within-class scatter is smaller

And the overall class scatter which we have seen the overall mixtures scatter or data samples is the summation of this is just a property the proof is given in many satirical books okay and the criteria formulation of class separable need to convert these matrix into a number so we will see how this is done and this number should be larger and then scatter is small why do you want to do.

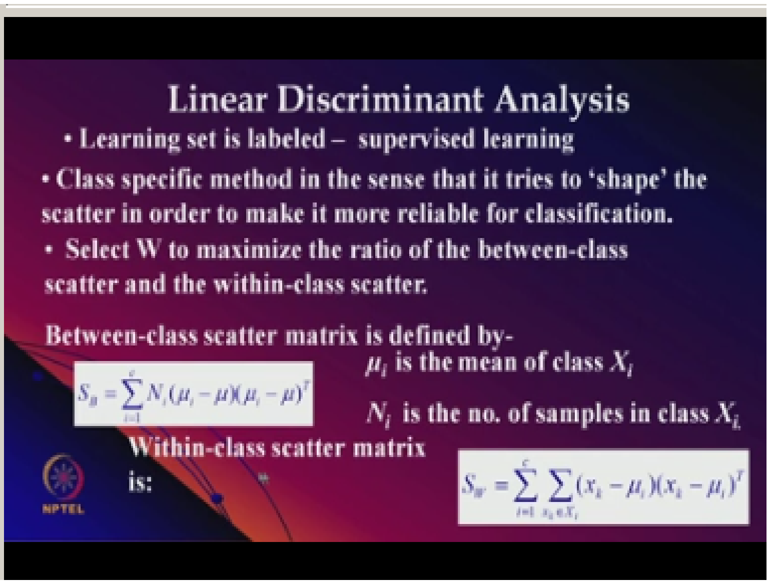
This forget this factor about converting matrix into number we will see how do this we will now form and different criteria using SW and SB and remember when we are talking about classifications versus clustering much earlier in this course what did we say that the classification problem becomes easier and job of the classifier becomes easy if the within class scatter is less and the between class scatter is larger.

If there are two class sample class 1 and class 2 class a and class b you would prefer or you would desires to have a scenario were the distance between the individual class means are much wider and the individual class scatter for before a particular class a and class b they are not large then this samples will not overlap leading to easy formulation of a very good decision boundary linear or non linear whatever the case may be between the two class scatters.

So you want a larger between class scatter and as smaller interact class or a within class so it typically if we go back to the expression of SW and SB I want smaller values of SW and larger values of SB that will be the bases of my optimization or criteria which I want to make in order to have projection dimension

And typically examples of such things are these okay let us take an example of something like this trace of S1/S2 as a criteria as these SW and SB so I want one of them to be larger one of them to be smaller the SB between class scatter should be largest when the numerator of that expression and within class scatter should be small.

(Refer Slide Time: 01:01:02)



Linear Discriminant Analysis

- Learning set is labeled – supervised learning
- Class specific method in the sense that it tries to 'shape' the scatter in order to make it more reliable for classification.
- Select W to maximize the ratio of the between-class scatter and the within-class scatter.

Between-class scatter matrix is defined by-

$$S_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T$$

μ_i is the mean of class X_i

N_i is the no. of samples in class X_i

Within-class scatter matrix is:

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

If you are heading towards this but if you cannot convert it to scatter quantity these as variances of this so this yields us to new method of classification which is called supervised which is under falls under the category of supervise learning because the learning set is level you want samples of different class available to you to compute the within class and within class scatter is called the linear discreet analyses.

And tries no shape the scatter to make it more reliable for actual classification so now we are trying to select W which will maximize the ratio of between class scatter versus with class scatter that is SW and SB and the between class scatter we have already defined them as this for mean for class I that linearism the number of samples which belong to sample xi what is the within class scatter this is defined as this earlier.

And what we want to do is maximize the ratio that means we will take SB/SW and that ratio we want to maximize we must select some projection matrix in which the in that projected domain the SB scatter becomes large and the SW not even the ratio must be larger for classification to work in fact to compute.

This scatter matrix we need the class samples hence it is called supervised learning unlikely the because you did not need to class labels and when you do not have class labels you can apply PCA and when you have class samples you can apply either PC or LD and in fact we will see why the LDA method I think we will stop with thing.

Online Video Editing /Post Production
K.R.Mahendra Babu
Soju Francis
S.Pradeepa
S.Subash

Camera
Selvam
Robert Joseph
Karthikeyan
Ram Kumar
Ramganesh
Sathiaraj

Studio Assistants
Krishankumar
Linuselvan
Saranraj

Animations

Anushree Santhosh
Pradeep Valan .S.L

NPTEL Web &Faculty Assistance Team

Allen Jacob Dinesh
Bharathi Balaji
Deepa Venkatraman
Dianis Bertin
Gayathri
Gurumoorthi
Jason Prasad
Jayanthi
Kamala Ramakrishnan
Lakshmi Priya

Malarvizhi
Manikandasivam
Mohana Sundari
Muthu Kumaran
Naveen Kumar
Palani
Salomi
Senthil
Sridharan
Suriyakumari

Administrative Assistant

Janakiraman.K.S

Video Producers

K.R. Ravindranath
Kannan Krishnamurty

IIT Madras Production

Funded By
Department of Higher Education
Ministry of Human Resource Development
Government of India

www.nptel.ac.in