Course Name: Machine Learning and Deep learning - Fundamentals and Applications

Professor Name: Prof. M. K. Bhuyan

Department Name: Electronics and Electrical Engineering

Institute Name: Indian Institute of Technology, Guwahati

Week-7

Lecture-27

Welcome to NPTEL MOOCs course on machine learning and deep learning fundamentals and applications. In my last class I discussed the concept of principal component analysis. Today I am going to discuss the same concept the principal component analysis, but from different perspectives. That is from the transformation point of view, I will be explaining the concept of principal component analysis and that transformation is called the KL transformation. So in the KL transformation, the original data that is highly correlated and after the transformation the transform data will be uncorrelated. And during the transformation, I will be getting a new coordinate axis and the transformed data will be aligned along the direction of this new coordinate axis.

And after the transformation the transform data will be uncorrelated. We know different types of transformation like DCT, the discrete cosine transformation, DFT, discrete Fourier transformation, Hadamard transformation, discrete sine transformation. So for all these transformation, the transformation kernels are fixed.

But in the KL transformation, the transformation kernel is not fixed.

It depends on the statistics of the input data. So now let us discuss about the KL transformation. And after the discussion, I will be explaining the concept of the PCA, the principal component analysis. So let us first discuss the concept of the KL transformation. So as I explained what is the difference between the KL transformation and other transformation like DFT, DCT or

maybe the discrete sine transformation or the Hadamard transformation or maybe a Haar transformation.

In this transformation, the transformation kernels are fixed. So now let us discuss about the KL transformation. That is the KL transformation. So from KL transformation, I will be explaining the concept of that is the principal component analysis.

What is KL? KL means Karhunen-Loe transformation.

So what is the basic difference between the KL transformation and other transformation the DFT, discrete Fourier transformation, DCT-the discrete cosine transformation, DST-discrete sine transformation. Like this, we have a number of transformations in all these transformations, the transformation kernel is fixed. So for DFT, the transformation kernel is fixed. For DCT, the transformation kernel is fixed like this transformation all these transformation the transformation kernels are fixed.

So that means the kernel is independent of data.

But in the KL transformation, the transformation kernel is derived from data. The transformation kernel depends on the statistical properties of the input data. So that is the difference between the KL transformation and other transformation like DFT, DCT, DST. So in case of the KL transformation, I am repeating this, that transformation kernel is derived from the statistics of the input data.

So now I am going to explain the concept of the KL transformation.

So let us consider a population vector. The population vector is suppose X-. So it is n dimensional vector x_1, x_2. These are the components of this vector. So this is n dimensional vector.

So this is a population vector n dimensional vector. From this population vector, what I can determine, I can determine the mean, mean of X- I can determine. So it is nothing but $\mu_{x\text{-}} = E\{x\text{-}\}$. So from this I can determine the mean, mean of the vector X-. And also I can determine the covariance matrix of the population vector.

So that size of this covariance matrix is nxn. This is the size of the covariance matrix. That is nothing but $C_{x\text{-}} = E\{(x\text{-} - \mu_{x\text{-}})(x\text{-} - \mu_{x\text{-}})^T\}$. So we can determine the covariance matrix like this. So it is a n by n matrix.

So in the covariance matrix in $C_{x\text{-}}$, the elements like this, suppose if I consider C_{ii}, that is the element is x_i and C_{ij}, that is the covariance between x_i and x_j. So the elements of the covariance matrix C_{ii} is x_i and C_{ij} is mainly the covariance between x_i and x_j. So this covariance matrix that is real and the symmetric, it is a real matrix and symmetric matrix. So from this matrix, we can get a set of n orthonormal vectors. So we can get from this covariance matrix what I can get?

I can get a set of n orthonormal vectors and these are called eigen vectors.

So these are eigen vectors. So these eigen vectors are represented like this e_i. So these eigenvectors. So you can see from the covariance matrix, we can determine the eigen vectors. So after this, let us move to the next slide.

So what we have determined from the covariance matrix, we have the covariance matrix C_{x^-} and from this what we have determined? We have determined eigen vectors. That means we can get a set of n orthonormal vectors, n number of orthonormal vectors and that is nothing but the eigen vectors. And corresponding to this e_i, corresponding to this e_i, I have λ_i and that is nothing but the eigen values. This is nothing but the eigen values. So you can see what is the step?

The step is from the covariance matrix, we can determine n number of orthonormal vectors.

These are the eigen vectors and corresponding to these eigen vectors, I have the eigen values. So these eigen values are arranged in the descending order of the magnitude. So I can write these eigen values are arranged in the descending order of magnitude.

So all these eigen values are arranged in the descending order of magnitude like this. $\lambda_j \geq \lambda_{j+1}$, like this I am arranging.

So $j = 0,1,2,\ldots,n-1$. So you can see we are arranging the eigen values in the descending order of magnitude. Now we have to construct the transformation matrix. So from the set of eigen vectors, we can from A matrix, the matrix is the A, A is the transformation matrix. Transformation matrix A I can form with the help of the eigen vectors.

So how to form this? The first row of A matrix is the eigenvector corresponding to the largest eigenvalue. Like this we have to consider. So A is a transformation matrix. So you can see I am making the transformation matrix. So first row is the eigen vector.

So suppose eigen vector is e_1 and this corresponds to the largest eigen value. And similarly e_2 is the second eigen vector corresponding to the second largest eigen value. And what is e_2 last row of the transformation matrix? It is the eigen vector corresponding to the smallest eigen value. So the last row I will be getting, last row of the matrix A is the eigen vector corresponding to the smallest eigenvalue. So like this we can construct the transformation matrix.

After this I will go for the transformation and that is nothing but the KL transformation. What is the KL transformation? The transformation I can write like this. And this transformation is actually the KL transformation.

KL transformation is $\vec{Y} = A(\vec{X} - \mu_{\vec{X}})$. A is the transformation matrix.

$\vec{X}$ is the population vector and the mean, mean of this vector $\vec{X}$. So here you can see what is $\vec{X}$? $\vec{X}$ is nothing but the data vector. And this $\mu_{\vec{X}}$ is the mean vector.

This is the mean. So I am getting the transformation.

The transformation is $\vec{Y} = A(\vec{X} - \mu_{\vec{X}})$. So the transformed data is nothing but the $\vec{Y}$. $\vec{Y}$ is the transformed data. The original data is the $\vec{X}$. So what is the properties of $\vec{Y}$?

The properties of $\vec{Y}$.

So what are the properties of $\vec{Y}$? That is the transformed data. So one property is very important. The mean of the transformed data is zero.

So $\vec{Y}$ is equal to zero. It is zero. And $C_{\vec{Y}}$ that is the covariance matrix of $\vec{Y}$. $C_{\vec{Y}}$ is the covariance matrix of $\vec{Y}$ which is generated from the covariance matrix of $\vec{X}$ and the transformation matrix and the transformation matrix is A. So that means the $C_{\vec{Y}}$ is nothing but $AC_{\vec{X}}A^T$. So this is the covariance matrix of the transformed data. Co-variance matrix of the transformed data is this is the expression.

So how to get this expression if you see $C_{\vec{Y}} = E[\vec{Y}\vec{Y}^T]$ because the mean of $\vec{Y}$ is equal to zero. So, $C_{\vec{Y}} = E[\vec{Y}\vec{Y}^T]$. So there is a covariance matrix. So this can be written like this $A(\vec{X} - \mu_{\vec{X}})$. And after this I have to write Y transpose.

So that is nothing but $(A(\vec{X} - \mu_{\vec{X}}))^T$. So which is equal to $AE\{(\vec{X} - \mu_{\vec{X}})(\vec{X} - \mu_{\vec{X}})^T\}A^T$. So if you see this is nothing but A and this $E\{(\vec{X} - \mu_{\vec{X}})(\vec{X} - \mu_{\vec{X}})^T\}$that is nothing but the covariance matrix of $\vec{X}$ $C_{\vec{X}}$ and A transpose. So from this actually we obtain this. So this is the expression for the covariance matrix of the transformed data.

So let us move to the next slide. So what will be the nature of the covariance matrix of $C_{\vec{Y}}$? The nature of the covariance matrix of $C_{\vec{Y}}$ will be like this. So this is a matrix. So diagonal elements are like this λ_1, λ_2 like this up to λ_n. And if you see this row these are all zeros. So you can see the diagonal elements are λ_1, λ_2, λ_3 like this up to λ_n and off diagonal elements are zero.

So you can see this off diagonal elements are zero. So that means it perfectly de-correlates data. So that means perfectly de-correlates the input data. So elements of the vector $\vec{Y}$ is uncorrelated. That means the transformed data is uncorrelated.

So I can write the elements of the vector $\bar{Y}$ are uncorrelated. So that is the meaning of this covariance matrix. And one thing is that Eigen values of $C_{\bar{Y}}$ is same as that of $C_{\bar{X}}$. So that I can write the Eigen value, value of the covariance matrix $C_{\bar{Y}}$ is same as that of $C_{\bar{X}}$.

So that is also one property. And similarly another is the Eigen vectors of $C_{\bar{Y}}$ is same as that of $C_{\bar{X}}$. So these are properties. So you can see that the KL transformation is $\bar{Y} = A(\bar{X} - \mu_{\bar{X}})$. We consider this one. So you can see the original data, they are highly correlated.

But after the transformation, I am getting $\bar{Y}$ that is the transformed data are uncorrelated. Because I have the diagonal covariance matrix. Since I have the diagonal covariance matrix, off diagonal elements are all zero.

So that means I can say the transformed data are highly uncorrelated, totally uncorrelated.

The transformed data are totally uncorrelated. So this is the observation after the KL transformation. So let us discuss one example how to determine this transformation matrix from the input data. So let us consider one example.

So how to determine the transformation matrix.

So a binary image is considered. So let us consider this binary image. And suppose these are the pixel positions. So let us consider this image. So at a particular point suppose (3, 4) x is 3 and y is 4. So suppose this x and this y corresponding to this position the pixel is present.

So that means when the pixel is present, I am considering it as 1. If the pixel is not present, when the object is not present, then it is the element is zero. So it is a binary image. And similarly I can consider another (4, 3).

And suppose (4, 4). So a particular object is considered and that is represented by a binary image (4, 4), (4, 5), (5, 4). So this object is considered. Suppose I am considering this object and I have the coordinates x and y. So what is the population vector in this case? The population vector I can consider is $\bar{X}$.

So (3, 4) in the (3, 4) position the pixel is present. (3, 4). If you see x is 3, y is 4 the pixel is present. Another one is at the point (4, 3) the pixel is present 4 and 3. Another point is (4, 4) the pixel is present. (4, 5) the pixel is present.

And another one is (5, 4) the pixel is present. (5, 5) the pixel is present. (5, 6) the pixel is present. And (6, 5) the pixel is present. So I am considering these cases and based on this I can determine the population vector. So that means what I want to determine?

I want to determine the KL transformation.

KL transformation I am determining of the pixels where the object is present. So you can see the object I am considering and these pixels at the position (3, 4), (4, 3), (4, 4), (4, 5), (5, 4), (5, 5), (5, 6), (6, 5) the object is present corresponding to these positions. So 8 numbers of 2D vectors that is nothing but the 8 numbers of 2 dimensional vectors. And from this population vector I can easily determine the mean.

So there is a mean vector I can determine. So the mean vector will be something like 4.5 and 4.5 you can determine. And also you can determine the covariance matrix. The covariance matrix is the $C_{X\text{-}} = E\{(x\text{-} - \mu_{x\text{-}})(x\text{-} - \mu_{x\text{-}})^T\}$.

So this covariance matrix you can determine. So for determining this covariance matrix what you can do? You can just determine $x_1 - \mu_x$ and after this $x_2 - \mu_x$, $x_3 - \mu_x$. So you have to determine like this and from $x_1 - \mu_x$ also you can determine $(x_1 - \mu_x)^T$ you can determine. So from this you can determine the covariance matrix. The covariance matrix is nothing but this is $x_1 - \mu_x$ you can determine and $(x_1 - \mu_x)^T$ you can determine. So like this you can determine like this you can determine all these values you can determine.

After this you have to take the average of all these to get the covariance matrix. The covariance matrix is the $C_{X\text{-}}$ you can determine the covariance matrix. So after determining the covariance matrix you have to determine the Eigen values and the corresponding Eigen vectors. What you need to determine? You have to determine the Eigen values and the corresponding Eigen vectors that you can determine from the covariance matrix.

So after this you can apply the transformation. The transformation is already I have defined $Y\text{-} = A(X\text{-} - \mu_{X\text{-}})$ you can determine this transformation. So you will be getting the transform data. So after the transformation what you can see the origin of the object is located at the centroid of the object.

So that means I will be getting the origin of the new coordinate system.

So origin is suppose this is the origin. So origin is located at the centroid of the object and the axis will be parallel to the direction of the Eigen vectors. So corresponding to this problem I have two Eigen vectors. So Eigen vectors I have two Eigen vectors one is e_1 and another one is e_2. So that means the origin of the new coordinate system I am getting a new coordinate system and origin is located at the centroid of the object and the axis will be parallel to the direction of the Eigen vectors. So you can see I am getting a new coordinate system and this new coordinate these are the this is the axis.

So one is e_1 and another one is e_2. So e_1 is the Eigen vector and e_2 is the Eigen vector. So that means after the transformation I am getting a new coordinate system and the transform data will be aligned in the direction of the new coordinate axis that is in the direction of the Eigen vectors. So that is why this KL transformation is also called the rotation transformation because I am getting a new coordinate system because I am getting a new coordinate system and the new axis are the Eigen vectors of the covariance matrix. So that means data will be aligned to the direction of the Eigen vectors and because of these alignments different elements of $\vec{Y}$ will be uncorrelated. So I am repeating this because of this alignment, alignment means I am aligning the transform data along the direction of the Eigen vectors and because of these alignments different elements of $\vec{Y}$ will be uncorrelated.

That is the interpretation of this KL transformation $\vec{Y} = A(\vec{X} - \mu_{\vec{X}})$. So I am repeating it again because it is very important. So after the transformation I will be getting a new coordinate system. The axis of the new coordinate system will be parallel to the direction of the Eigen vectors.

And because of these alignments different elements of $\vec{Y}$ will be uncorrelated. So that is the interpretation of the KL transformation. Now let us consider the reconstruction of the original data from $\vec{Y}$ that is how to reconstruct $\vec{X}$ from $\vec{Y}$. So that is the reconstruction problem. So let us move to the next slide reconstruction.

So how to reconstruct $\vec{X}$ from $\vec{Y}$ that is to reconstruct $\vec{X}$ from $\vec{Y}$. Already we have done the transformation. So we know that $\vec{Y} = A(\vec{X} - \mu_{\vec{X}})$. So this is the transformation. So this $\vec{Y}$ is what? $\vec{Y}$ is nothing but it is n dimensional.

So this transformation matrix it is orthogonal transformation matrix. So $A^{-1} = A^T$ that is the orthogonal matrix. So what is the inverse KL transformation? Inverse KL transformation I can write like this inverse KL transformation is nothing but $\hat{\vec{X}} = A_K^T \vec{Y} + \mu_{\vec{X}}$. This $A^{-1} = A^T$ that is nothing but the orthogonal transformation. So you can see I can reconstruct the original data. So you can see this transformation matrix A that is formed by all the eigenvectors of the covariance matrix $C_{\vec{X}}$.

That means I am repeating this the transformation matrix A is constructed by considering all the eigenvectors of the covariance matrix $C_{X\text{-}}$. But if I only consider k number of eigen vectors then I will be getting the transformation matrix A_K. In the A_K that means what we are considering? We are only considering the K number of eigen vectors that is we are only considering K number of eigen vectors. In the case of A we have been considering all the eigen vectors but now we are considering K number of eigen vectors. So that means A_K we are considering that is the transformation matrix K number of eigen vectors of the covariance matrix $C_{X\text{-}}$.

What we are considering the K largest eigen vectors we are considering. So what will be the dimension of A_K? The dimension of A_K is K x n because we are considering only K number of eigen vectors. So dimension will be K x n. So corresponding to this my transformation will be Y- is equal to A_K. So in place of A I am considering the A_K and this A_K I can consider as a truncated transformation matrix because I am not considering all the eigen vectors for constructing the transformation matrix.

A_K I can consider as the truncated transformation matrix. So I can write truncated transformation matrix because we are only considering the K number of eigen vectors for constructing the transformation matrix. So $A_K(X\text{-} - \mu_{X\text{-}})$. So what is the dimension of Y-? The dimension of Y- will be K. What is the dimension of A_K? The dimension of A_K is K x n.

And what is the dimension of X-? The dimension of X- is n x 1, n dimensional. So we are getting this. So in this case for that transformation we are considering only the K number of eigen vectors. So that means these are I can consider as the principal components. So these are actually I can consider the principal components. That is the largest eigen values we are considering and corresponding eigen vectors and these are called the principal components.

So in the principal component analysis this is the case. That means we are considering the largest eigen values and corresponding eigen vectors. So this should be K largest eigen values. So now how to do the reconstruction? So move to the next slide. So how to do the reconstruction? So if I want to reconstruct X- from Y-, reconstruction of X- from Y-. So in this case if I want to reconstruct X- from Y-, we will not be able to get the perfect reconstruction because we are not considering all the eigen vectors for constructing the transformation matrix.

So perfect reconstruction is not possible. So we are getting the approximate reconstruction. So this is the approximate reconstruction of X-, $A_K^T Y\text{-} + \mu_{X\text{-}}$. So here you can see the dimension of X- will be n, dimension of A_K^T that will be K x n, dimension of Y- will be K.

So this the dimension of X- that will be the same dimension of X- but approximate value of X-. So we are getting the approximate value of X- because we are not considering all the eigen vectors for constructing the transformation matrix.

So this ak you can see the A_K is dimension is what is the dimension of A_K? It is K x n. What is the dimension of Y-? The dimension of Y- is K and the approximate X- not the perfect reconstruction is possible. So dimension of X- will be n. So this is the reconstruction of X- from X- and we are only considering K number of eigen vectors corresponding to K number of largest eigen values. Now in this case what is the mean square error? The mean square error is defined like this $e_{ms} = \sum_{j=1}{}^{n} \lambda_j - \sum_{i=1}{}^{K} \lambda_i$.

So in the first case you can see what is λ_j that is the eigen values. So in the eigen value we are considering $i = 1{:}n$. So in the eigen value you can see $j = 1{:}n$ we are considering that means we are considering all the eigen values minus λ_i that is also the eigen value from i is equal to K that means we are considering only K number of largest eigen values. So that is equal to $\sum_{j=K+1}^{n} \lambda_j$.

So that is nothing but the sum of the neglected eigen values. So sum of the neglected eigen values. This KL transformation is called optimum transformation because it minimizes the mean square error of reconstruction error between X- and $\hat{X}$-. So that means I can say it is optimum transformation. Why it is called the optimum transformation? Because we have to minimize the MSE the mean square error of reconstruction error reconstruction error between X- and $\hat{X}$-. So the KL transformation is called the optimum transformation. So if I consider all the eigen vectors for the construction of the transformation matrix then the reconstruction error will be 0.

I can do perfect reconstruction. But if I consider only the K number of eigen vectors then it is not possible to perfectly reconstruct the original data that original data is $\hat{X}$-. So this is about the reconstruction of the vector $\hat{X}$- from Y-. So now let us consider how actually you can apply this transformation for the image the two dimensional image and how it can be used for data compression. So let us move to the next slide.

So how to apply KL transformation that is the PCA the principal component analysis. How to apply KL transformation in an image. So image is a 2D array of numbers. So let us consider one N x N image and suppose X_0, X_1, X_2 suppose this is one is suppose X_i like this.

So we can consider every column as a vector. So you can see we have N number of vectors. So image is nothing but the 2D array of numbers. So image I can write it is a 2D array of quantized intensity values. So you can see what we are considering the data is represented as a vector. So you can see I have N number of vectors X_0 X_1 X_2 like this I have N number of vectors. So from N number of vectors I can determine the mean $\mu_{X\text{-}} = \frac{1}{N}\sum_{i=0}^{N-1} X\text{-}_i$ and also I can determine the covariance matrix $C_{X\text{-}} = \frac{1}{N}\sum_{i=0}^{N-1}(X\text{-}_i - \mu_{X\text{-}})(X\text{-}_i - \mu_{X\text{-}})^T$.

So we can determine the covariance matrix Cx and the mean mu x we can determine. So dimension of mu x is n dimensional. So you can see the dimension n and what is the dimension of $C_{X\text{-}}$ the dimension is N x N dimension of $C_{X\text{-}}$ is N x N. So from $C_{X\text{-}}$ I can determine the eigenvalues and the corresponding eigenvectors. So I can determine eigenvalues and eigenvectors. So after determining the eigenvalues and eigenvectors that means we are determining λ_i that is the eigenvalues and corresponding to $i = 0,1,\ldots,N-1$.

So N number of eigenvalues we can determine and corresponding eigenvectors so $i = 0,1,\ldots,N-1$ we can determine. So N number of eigenvalues and N number of eigenvectors. So after determining that this eigenvalues and eigenvectors we can determine the transformation matrix. So let us move to the next slide how to determine the transformation matrix.

The transformation matrix is the A transformation matrix transformation matrix is A. So that is determined from the eigenvectors. So first row is the eigenvector e_0^T that corresponds to the largest eigenvalue. Next one is e_1^T like this e_{N-1}^T transpose. So because we are considering this transpose because the vector is normally represented as column vector. And in this case what we are considering $\lambda_0 \geq \lambda_1 \geq \lambda_2 \ldots \geq \lambda_{N-1}$.

So corresponding to λ_0 what is my eigenvector eigenvector is e_0 corresponding to λ_1 my eigenvector is e_1. So you can see I am constructing the transformation matrix with the help of the eigenvectors. So the first row is the eigenvector corresponding to the largest eigenvalue. Second row is the eigenvector corresponding to the second largest eigenvalue.

So like this I am constructing the transformation matrix. So after this suppose we are considering the truncated transformation matrix because in the truncated transformation matrix we have to consider K number of largest eigenvectors. So what is the truncated transformation matrix? So we are considering only the K number of eigenvectors. So the truncated transformation matrix is A_K e_0^T e_1^T like this only we are considering the K number so it is e_{K-1}^T transpose and this is the truncated transformation matrix. So first K number of eigenvectors we are considering corresponding to K number of largest eigenvalues.

So now we have to apply the transformation, transformation of every column vector of the image. I have to apply the transformation for all the columns of the image. So for every x_i of the image we are getting y_i. So we are applying the transformation column wise. So suppose I am getting the transformation y_i and A_K is the truncated transformation matrix x_i is a particular column of the image we are considering like this.

So for all the columns I have to apply this transformation this is the kl transformation. So $i = 0,1,\ldots,N-1$. So for every column that means for every x_i I have to determine y_i that is the transform vector I have to determine. So y_i what is the dimension of y_i? Dimension of Y- is K x 1 and this is the modified transformation matrix the dimension is K x N and this is the input vector it is the dimension is N x 1 this is the mean vector so it is N x 1. So that means I will be getting n number of Y-'s I will be getting and finally I will be getting N number of y_i I will be getting. So if the transformation of all the column vectors of the 2D image is done then we will be getting N transform vector y_i with the dimension K.

So what will be the transform image? So move to the next slide. So transform image will be transform image what is the dimension of this K x N and what is the original size? Original size of the image is N x N. So you can see the dimension is reduced by this transformation and you can see we are considering only the K number of eigenvectors corresponding to the K number of largest eigenvalues and that is why and this is called the PCA the principal component analysis because we are considering the K number of largest eigenvectors corresponding to the largest eigenvalues. So how to reconstruct the original image? So we have to do the reconstruction. So for reconstruction this approximate image I am getting this is the approximate image I am getting so that is nothing but $A_K^T Y_i + \mu_X$.

So these are reconstruction formula. So in this case A_K is not a square matrix so for this what we can consider maybe pseudo inverse we have to consider for determining the inverse the pseudo inverse by singular value decomposition to calculate the inverse of A non-square matrix that we can consider. So we are determining the x_i. So collection of all x_i that means the collection of all the x_i will give the dimension will give the image of the dimension N x N that is the reconstructed or the approximate image. So in this x_i what is the dimension of this x_i? The dimension of x_i was N this is the dimension of x_i.

So this collection of all the x_i will give the image the reconstructed image of the size N x N. So you can see how to reconstruct the original image. So now let us see how to get $\hat{x_i}$ that is the reconstructed value of x_i. So for reconstruction what information I need? The information I need A_K that is we need to save A_K and also what information I need y_i. So $i = 0,1,\ldots,N-1$. So these two information these two information one is A_K another one is y_i I need to reconstruct x_i these two information I need.

Now if I want to do the compression data compression the compression depends on the value of K or the compression of the data. The compression depends on the value of K. So if I consider suppose $K = 1$ that means we are considering only one eigen vector for the transformation matrix. If I consider $K = 2$ that means we are considering two eigen vectors for the transformation matrix. So if I increased the value of K that means the quality of the image will be improved with the increase of the number of eigen vectors.

So that means the quality of the image will be improved with the increase of number of eigen vectors. So if I consider all the eigen vectors of the covariance matrix for the construction of the transformation matrix then the reconstruction would be perfect. So that is the case. So this is about the KL transformation and you can see how actually the principal component concept is coming from the KL transformation. So in this class I discussed the concept of the KL transformation and finally I have explained the concept of the principal component analysis.

So what are the advantages of the KL transformation? The first advantage is that it can perfectly de-correlates the input data. The original data is highly correlated and after the transformation the transformed data will be uncorrelated and energy compaction is very high in case of the KL transformation. So what are the disadvantages of the KL transformation? In the KL transformation the transformation kernel is dependent on the statistics of the input data. So from the input data we have to determine the mean vector, we have to determine the covariance matrix, from the covariance matrix I have to determine the eigen values and the eigen vectors and from this I can determine the transformation matrix. For non-stationary data or for the non-stationary signal it is very difficult to compute the transformation matrix for each and every instance.

So that is why the computational complexity is more for the non-stationary data. So this KL transformation cannot be applied in the real time. So if I want to go for image compression or the video compression with the help of the KL transformation it is not possible because of this case. The case is if the data is non-stationary we have to determine the transformation matrix for different, different instance because the transformation kernel depends on the statistics of the input data. So that is why the real time implementation is not possible in the KL transformation. So that is the main disadvantage of the KL transformation. So let me stop here today. Thank you.