Essential Mathematics for Machine Learning

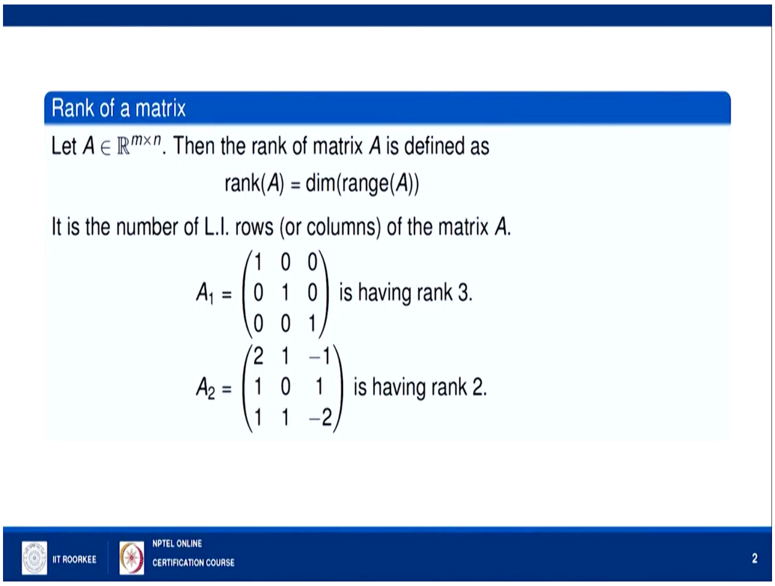
Prof. Sanjeev Kumar
Department of Mathematics
Indian Institute of Technology, Roorkee

Lecture - 13
Low Rank Approximation

Hello friends. So, welcome to lecture number 13 of this course. I hope you are enjoying all the lectures and the continuation which we are having especially in case of singular values. So, again in this lecture like the previous 2 lectures, this lecture is again related to singular value decomposition.

However, we are having a very important application of singular value decomposition which we will discuss in this lecture and that is called low rank approximation. So, first we will start this lecture with the definition of rank. However, we have taken rank in previous lectures and by this time you know very well what we mean by rank of a matrix. However, again let me define it.

(Refer Slide Time: 01:15)



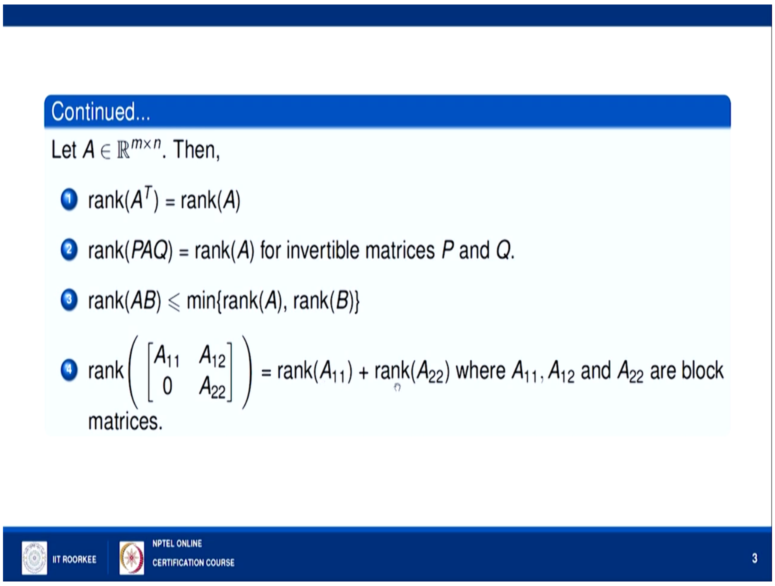
So, let A be a m by n real matrix. Then the rank of matrix A is defined as the dimension of range space of A, and the similarly dimension of null space is called a nullity.

But here we are focusing on rank. It is the number of linearly independent rows or columns of the matrix A. So for example, if you take this matrix, so it is an identity matrix and it is having 3 linearly independent row or 3 linearly independent columns. So, rank of A 1 is 3.

Similarly, if you see this A 2. So, here if you see this in this form we cannot say about the linearly independent or linearly dependent. However, if you carefully check it, if I subtract second row from the first one what I will get, 2 minus 1 1, 1 minus 0 1, minus 1 minus 1 equals to minus 2.

So, third now is nothing just it is depending on the first 2 rows that is, third row is the R 3 that is the row 3 equals to row 1 minus row 2. So, all 3 rows are not linearly independent; 1 row depending on the linear combination of other 2 rows.

(Refer Slide Time: 02:46)



Hence, the rank of this matrix is 2. So, this is all about the rank then we are having some of the important property about rank, one of the properties rank of A equals to rank of A transpose. And that is why I am saying row ranks equals to column rank. Rank of a matrix PAQ equals to rank of A for any invertible matrices P and Q.

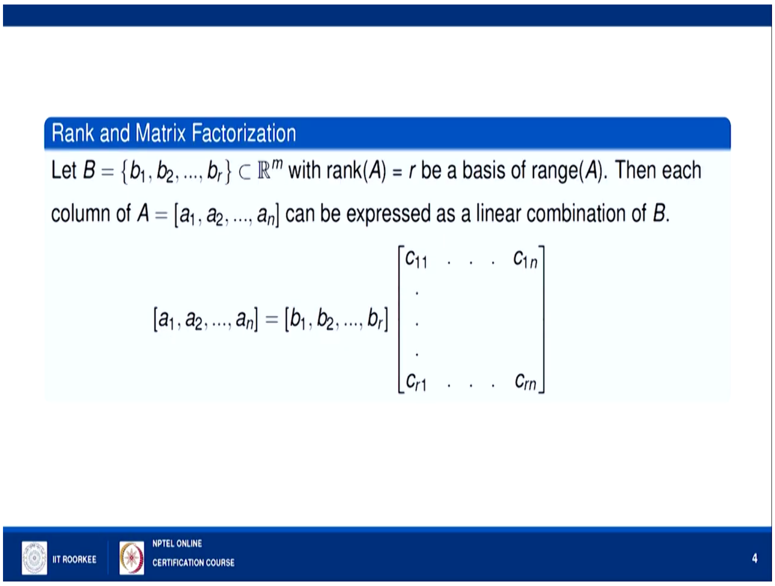
So, if P is invertible Q is invertible then rank of the product of PAQ equals to rank of A. Then we are having; so like in the case of singular value decomposition we have seen that rank of A equals to rank of sigma.

That is why we are saying the rank of A equals to the number of non-zero singular values, that is the number of non-zero rows or non-zero columns. Because, their u and v transpose are orthogonal matrices and they are invertible. My third property is rank of the product of 2 matrices A and B, that is rank of A B will be always less than equals to minimum of rank A comma rank B.

Fourth one is if you are having this kind of block matrix where, A 11 is a block matrix, A 12 is a block matrix, A 22 this equals to greater than equals to; so it will be greater than equals to. In the case when A 11 and A 22 are zero matrices.

So, right hand side will be 0; however, left hand side will depend on the rank of A 12. So, this may this connection it is greater than equals to, where all these A 11, A 12 and A 22 are block matrices. So, these are some of the important properties about rank which we can remember easily, and we will use these very frequently; even we have used these properties earlier also.

(Refer Slide Time: 04:48)



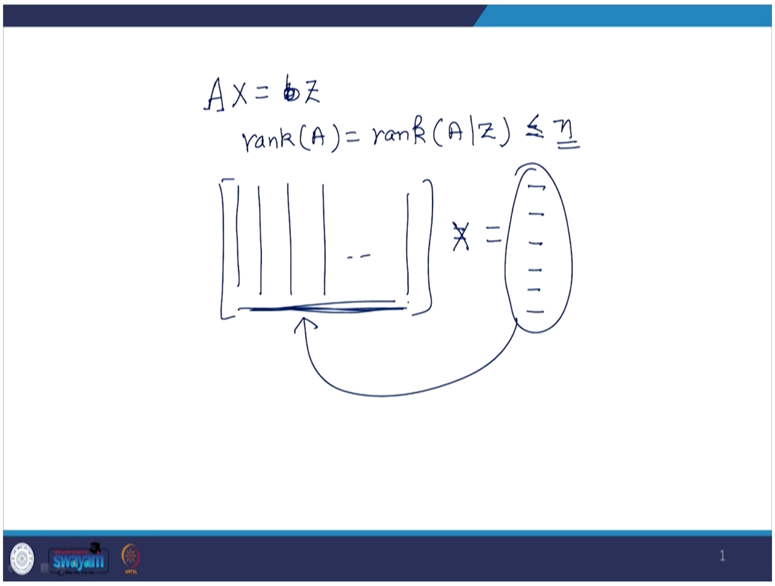
Now, matrix factorization and its relation with the rank. So, let B having vectors b 1, b 2, b r, all are m dimensional vector. So, B be a set of vectors and we are having rank of A equals to r where, A is a m by n matrix, and this B be a basis of range space of A.

So, matrix A is m by n matrix. So, it is a linear transformation from R n to R m; so obviously, range space will be a subspace of R m and that is why we are taking these b 1, b 2, b r as the m dimensional vectors ok. R linearly independent vectors.

Now, then each of the column of this matrix A can be expressed as a linear combination of B how? a 1, a 2, a n equals to b 1, b 2, b r and then columns of c 1, c so. So, now this matrix c is called a basis matrix. So, these columns are the basis and these are giving the coordinates.

So, if you want to make first column, first column will become b 1 c 11 plus b 2 c 12 plus plus up to b r c r 1. And that will be our a 1. So, in that way we will be having all the columns of A as the linear combination of B.

(Refer Slide Time: 06:13)



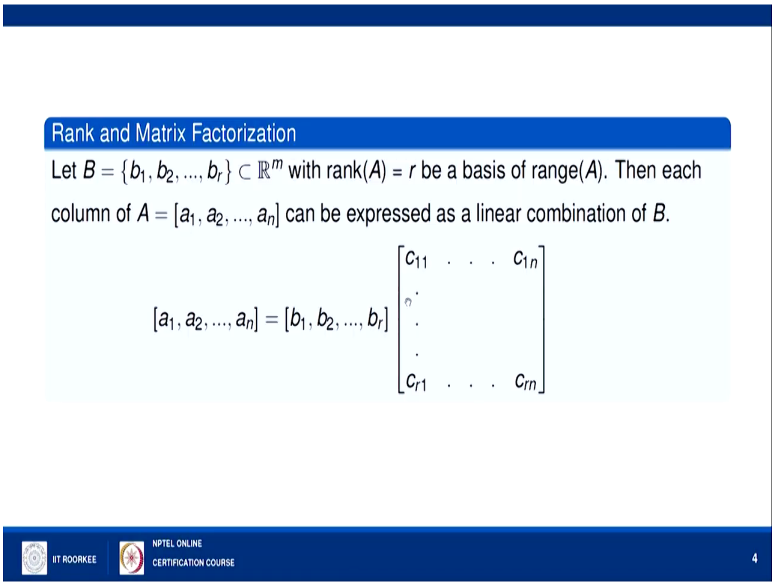
And that you can see very easily. Because, we are having if you consider the linear system Ax equals to b, or Ax equals to let us not write Ax equals to z. So, z is right hand side vector, this system will be having a solution when rank of A equals to rank of augmented matrix A z.

And when it will be having unique solution when this is equals to the number of unknowns, that is x, otherwise it will be having infinite number of solution. So, if rank of A is equals to R which is less than n, so let us say it is less than equals to n in that way so what will happen? If you can write the columns of A, and then you are having x and here you are having this z.

So, if you can write the z as the linear combinations of columns of matrix A, then the system will be having a solution. And what will be that solution? Solution will be the coordinates of that linear combination.

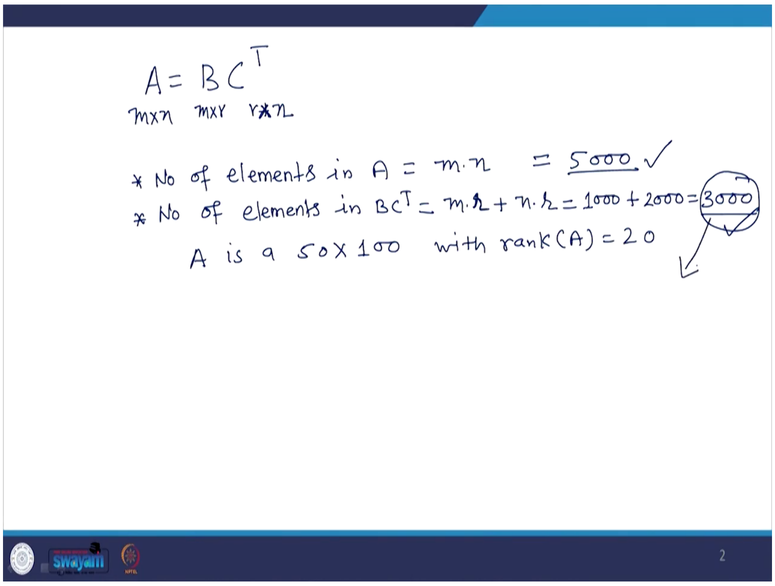
So, if you can write it uniquely; then, you will be having the unique solution. And when you can write uniquely when it is a full rank matrix. If you cannot write it uniquely means you can write in different way then you will be having more than one solution that is the infinite number of solutions; and when you will be having that when, this is not a full rank matrix.

(Refer Slide Time: 08:22)



So, in that way we can visualize this concept. So, we are saying that if A is a m by n matrix, rank of A equals to r then we can have a factorization of a as A equals to BC transpose, means we are writing the columns of A as the linear combination of product of the basis of the range space of A and then a matrix C; and here B is m by R matrix because, you are having R vectors each one of m dimension, and C is a n by R matrix. Then this particular product will be well defined.

(Refer Slide Time: 09:05)



So, what I am saying? I am saying A equals to B C transpose, where is A is m by n, B is m by r and C transpose is r by sorry r by n.

So, now what we are saying? If you see carefully, number of elements in A equals to m times n, number of elements in B into c transpose, so in B you will be having m into r elements plus in C transpose you will be having n into r elements.

Now, consider a situation where, where A is a 50 by 100 matrix with rank of A equals to let us say 20. So, how many elements you have to store here. For writing this matrix means you have to save how many elements?

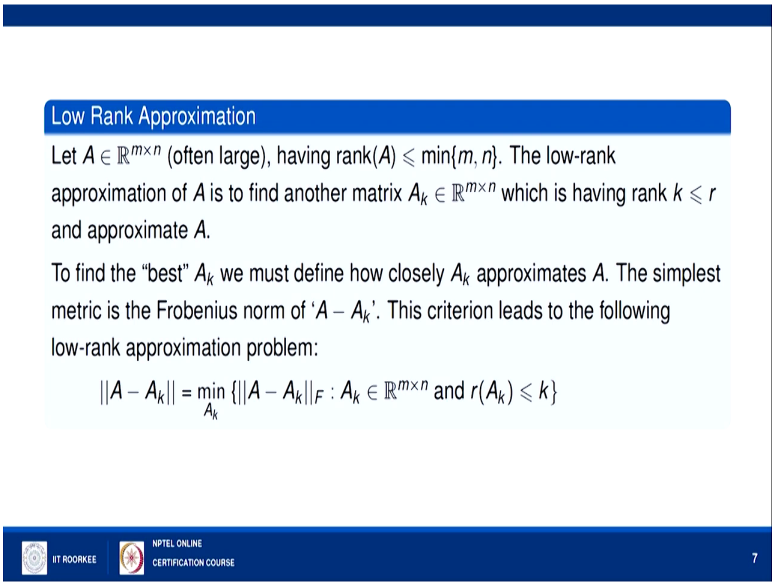
You have to store 50 into 100 that is 5000 elements; however, in this case how many elements you require to save? 50 into 20 that is 1000 plus 1000 100 into 20 that is 2000, so only 3000.

So, now if you see this motivator, this is the motivation of low rank approximation. So, in original case you have to save 5000 elements, but in this case you have to save only 3000 elements, and you are saving the same thing that is the matrix A.

So, in that way you can save the memory as well as the computational cost because, if you want to operate something on this matrix A, you want to process this matrix you have to process all the 5000 elements means, if it is an image digital image you will be having 5000 pixels in A, but in case of if you are writing it in this way B into C transpose then, you have to play with only 3000 pixels.

So, this is a motivation for low rank approximation ok. Given a matrix or a tensor you find out another matrix or tensor which is having the lower rank to the original one and the same time it is quite close to the original one.

(Refer Slide Time: 12:19)



So, based on this motivation let us define the low rank approximation. So, let A be a m by n matrix and it is large because, if matrices are very large then only we talk about low-rank approximation. Having rank of A which is less than equals to minimum of m or n.

The low-rank approximation of A is to find another matrix A k of the same size which is having rank k less than equals to R ok. Why size will be same, but again if rank is lower that is the rank is k which is the rank of the original matrix A.

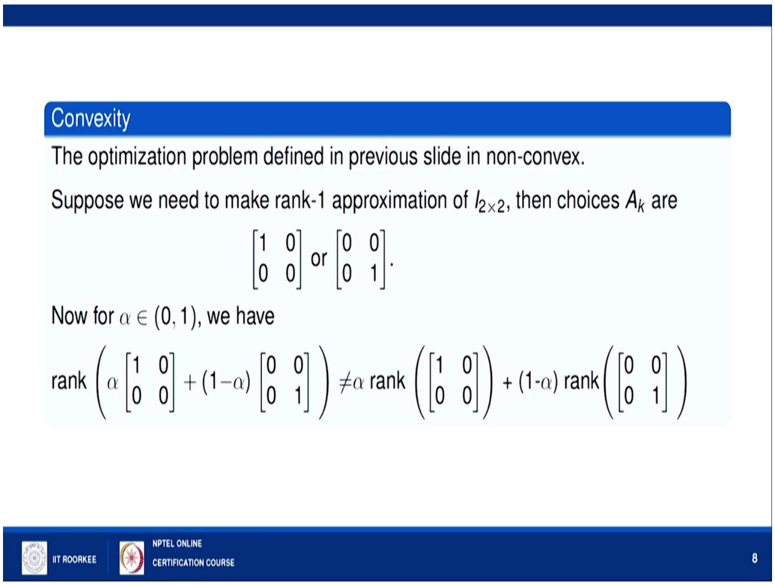
Then I can save this matrix easily with a lower number of entries by using the matrix factorization, which we have seen in the earlier slide. So, which is having rank k which is less than equals to R and approximate A.

Now, to find the best A k we must define how closely A k approximate A. The simplest metric is the Frobenius norm of the difference of these 2 matrices that is your original matrix and the matrix which you are approximating. This criteria leads to the following low-rank approximation problem.

So, Frobenius norm of A minus A k, you have to find out a matrix A k such that the minimum value of the Frobenius norm overall A k and rank of A k is less than equals to k. So, this is an optimization problem now. We have to check whether it is a convex optimization problem or non-convex optimization problem.

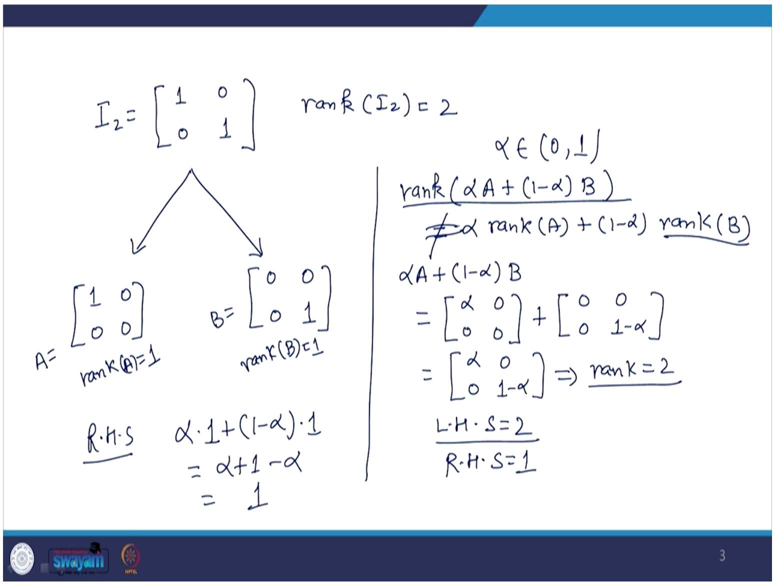
Because, if it is convex optimization problem we can solve this problem quite easily. So, now let us check unfortunately, this problem is not a convex optimization problem.

(Refer Slide Time: 14:23)



So, it is a non-convex. How? You check with the help of this small example. Suppose, we need to make rank one approximation of an identity matrix.

(Refer Slide Time: 14:39)



So, what you are having? You are having identity matrix I 2, which is 1 0 and 0 1. So, rank of I 2 is 2. Suppose, you have to make a rank one approximation of this matrix. So, you are having 2 choices, either you will go like this or you will go like this ok.

So, let us say I am saying this A and this I am saying B. So, rank of A equals to 1 and rank of B again 1. Now, let us check the convexity of this. So, now what I will do? I will find out for some alpha belongs to 0 1 rank of alpha times A.

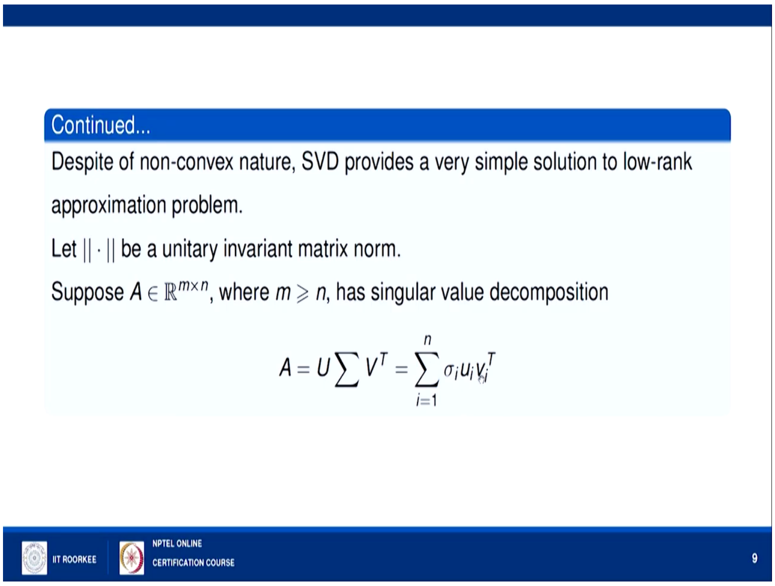
So, what I am having? Because, here my function is rank; rank of alpha times A plus 1 minus alpha times B, and for convexity it should be equals to alpha times rank of A plus 1 minus alpha times rank of B.

So, let us see whether this equality holds or not? So, first see alpha A plus 1 minus alpha times B. So, alpha A will become alpha 0 0 0 plus 0 0 0 1 minus alpha and here alpha belongs to 0 1. So, this will become alpha 0 0 1 minus alpha, and rank of this is 2; because it is having 2 non-zero rows and both are linearly independent.

So, this is the rank of this one is 2. So, left hand side is 2, now what is right hand side? So, alpha rank of h 1; so alpha times 1 plus 1 minus alpha times rank of B. So, I am calculating right hand side here. So, again the rank of B equals to 1. So, it is alpha plus 1 minus alpha it is 1. So, RHS is 1.

So, here this is not equal. And if this is not equal hence it is a non-convex problem. Now, it is very difficult to find out the global minima of non-convex optimization problem. You will see in coming lectures when we will talk about gradient calculus. So, what is the solution? So, solution come in form of singular value decomposition.

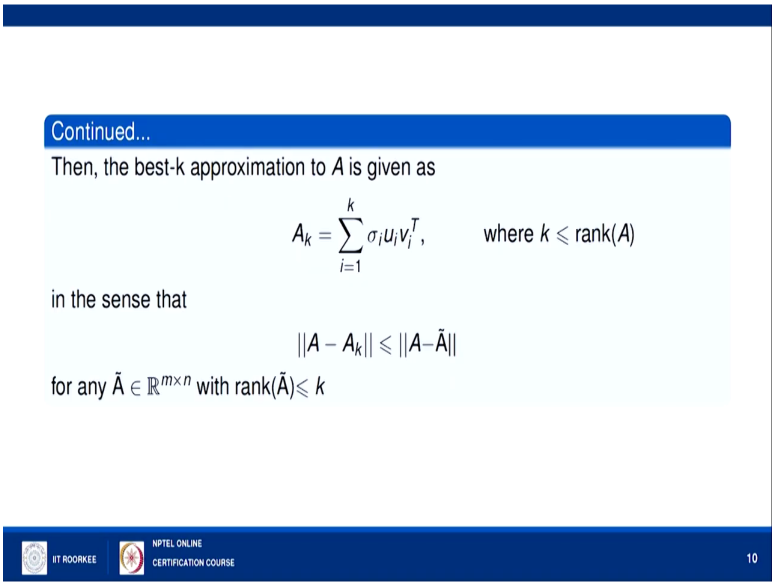
(Refer Slide Time: 18:11)



So, despite of non-convex nature, singular value decomposition provides a very simple solution to low-rank approximation problem that is the optimization problem, which we have posed in this slide this one. How? Let this particular thing be a unitary invariant norm.

And, suppose A is a m by n matrix where, m is greater than n we are taking this case it is true for another case also where, m is less than equals to n, has singular value decomposition h, A equals to U sigma V transpose; this I can write also i equals to 1 to n sigma i u i v i transpose.

(Refer Slide Time: 18:55)

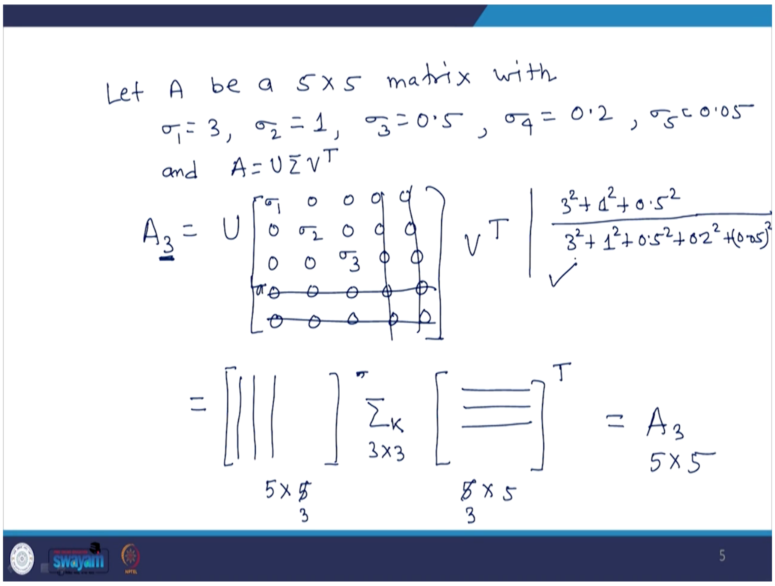


Now, then the best-k approximation to A is given as A k equals to i equals to 1 to k sigma i u i v i transpose, where k is less than equals to rank of A. So, please see here in this for the complete A, I am having this sum from 1 to n. So, it will be having a decomposition over, means sum over all singular values.

However, this when I am talking about the approximation I am having taking only k singular values and what k singular values? The largest k singular values after k singular values, I am making rest of the singular values as 0.

And, in the sense that, the norm Frobenius norm A minus A k or any unitary invariant norm less than equals to A minus A tilde, for any A tilde which is coming from the vector space of the matrix is real vector space of the real matrices of size m by n with rank tilde less than equals to k.

(Refer Slide Time: 20:15)



So, what I want to say? Let us see by some simple examples. So, let A be a 5 by 5 matrix with singular values as let us say sigma 1 equals to 3, sigma 2 equals to 1, sigma 3 equals to 0.5, sigma 4 equals to 0.2 and sigma 5 equals to let us say 0.05 ok.

So, what is rank of A? Here rank of A is 5 because; all 5 singular values are non-zero. Now, if I want to find out A 3 which is a low rank approximation of a with rank 3; then what it will become, and one more thing A equals to u sigma v transpose.

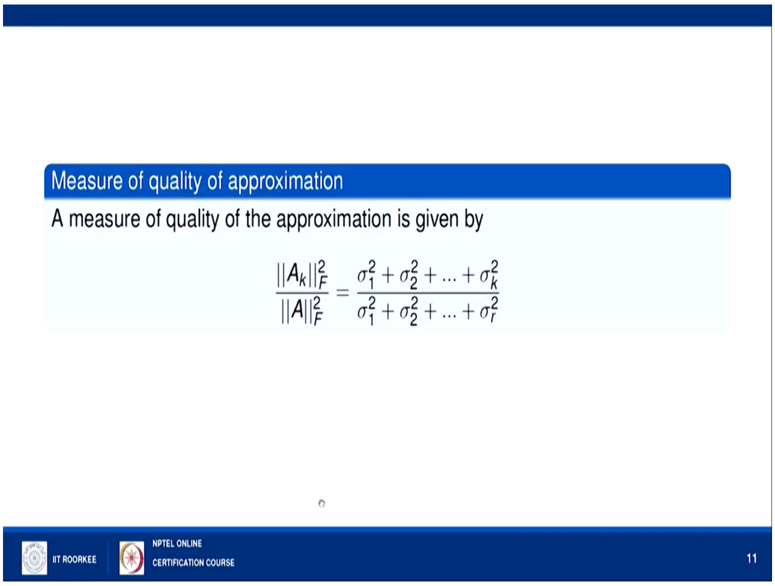
So, now what will be A 3? A 3 will be U and then what sigma U will be take sigma 1 0 0 0 0 0 sigma 2 0 0 0 0 0 sigma 3 0 0 and then two 0 rows. Because if I want a matrix three rank approximation of A, then it will be having rank 3 and it will which is having only 3 nonzero singular values into V transpose.

Now, what I want to say if you are having this matrix U which is a, if A is 5 by 5 matrix, so this will be a 5 by 5 matrix, then I have written this sigma. So, now ignore all these because, these all are 0. So, now this sigma k will become 3 by 3 matrix; and then V transpose which is again 5 by 5.

Now, how you will define this product? So, for defining this product this matrix must be 5 by 3 and this will be 3 by 5. So, 5 by 3 means you take first 3 columns of U and first 3 rows from here.

And then, make this product. It will ultimately give you a 5 by 5 matrix, and which will be having rank 3 as compared to rank 5 of the original matrix A, and this will be the best approximation of a with rank 3. Why I am saying it will be the best approximation? Let us see an example on it.

(Refer Slide Time: 23:39)

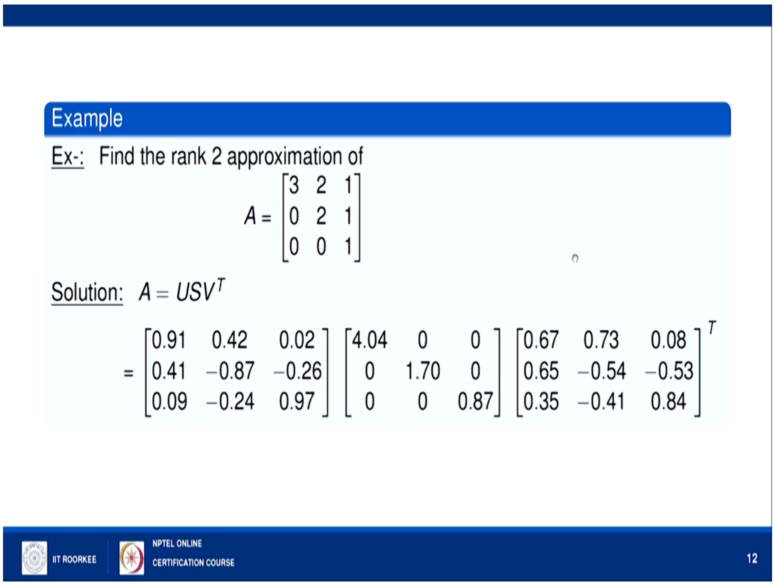


And one more thing here, the measure of quality of the approximation is given by that if you are going from rank R to rank k, where k is less than equals to R then sum of the squares of the first k singular values divided by sum of the squares of the R singular values.

So, for example, if you see here, approximation will be 3 square plus 1 square plus 0.5 square that is the first 3 upon sum of all. So, 3 square plus 1 square plus 0.5 square plus 0.2 square plus 0.05 square.

And you see if you ignore the a singular values those are quite close to 0 or very small then, quality of approximation will not be affect much ok. So, in that way this is a measure or this is a metric for finding the quality of low-rank approximation.

(Refer Slide Time: 24:54)

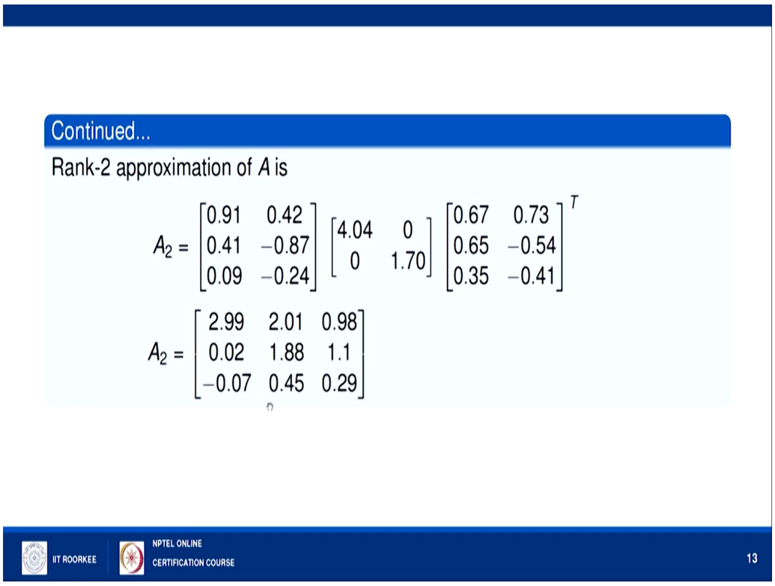


So, find the rank 2 approximation of this matrix. So, if you see it is a upper triangular matrix and the rank of this matrix is 3 because, you are having 3 non-zero linearly independent row. Now, find out the singular value decomposition of this matrix. So, singular value decomposition of this matrix is given by this is your matrix U, this is your matrix S, and this is your matrix V.

Now, what you do? You have to find out 2 rank 2 approximation from rank 3 original 1 2. So, what you do? You make this singular value as 0. What you do? You take first 2 columns of this matrix.

So, instead of 3 by 3 matrix you take these 3 by 2 matrix, take these 2 by 2 sub matrix. So, 3 by 2 multiplied with 2 by 2 and then they first 2 rows of this matrix; then 2 by 3.

(Refer Slide Time: 26:02)

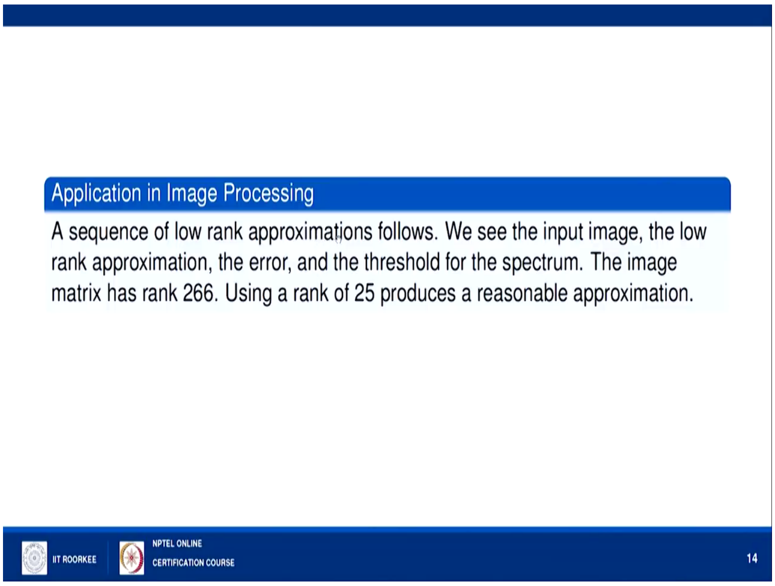


You will get these 3 by 3 matrix as I have taken here ok. And then this will be over why I am writing transpose because once you take the transpose this product will be defined these are the first 2 rows only. So, this will become the rank 2 approximation of the matrix A. So, rank of this matrix will be 2.

Now, if you compare this matrix with the earlier the original matrix A the first entry is 3 here. It is 2.99 so very close to 3 then, here it is 2; it is 2.01, so very close approximation ok.

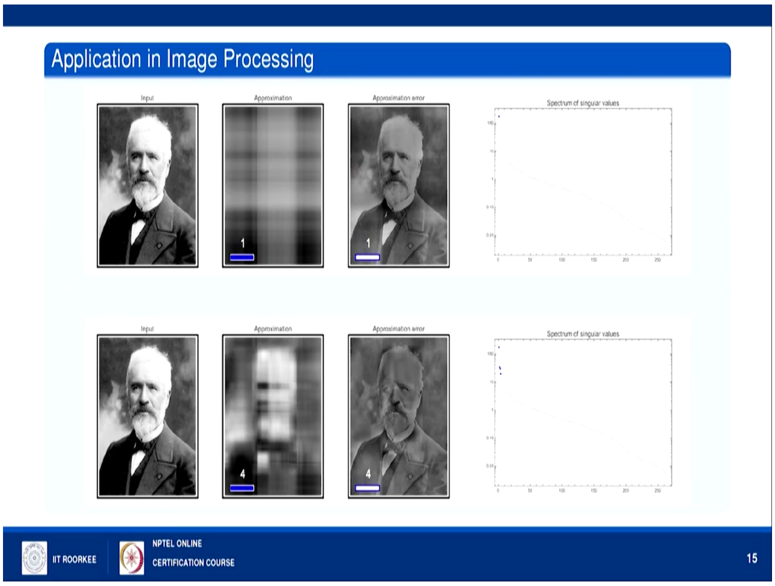
And the same time rank is reduced by 3 to 2 ok. So, this is one of the example of that for a smaller matrix but generally we do this low rank approximation for larger matrix. And where when we are dealing with learning of images and all those. So, let us see one example of digital images. So, we are having a sequence of low rank approximation.

(Refer Slide Time: 27:14)



We have a an input image, and we will find different low rank approximation, the error, and the threshold for the spectrum. The image matrix has rank 266. So, the what is the rank of the image? 266. Because image is a matrix only. So, it is having 266 nonzero singular values.

(Refer Slide Time: 27:44)

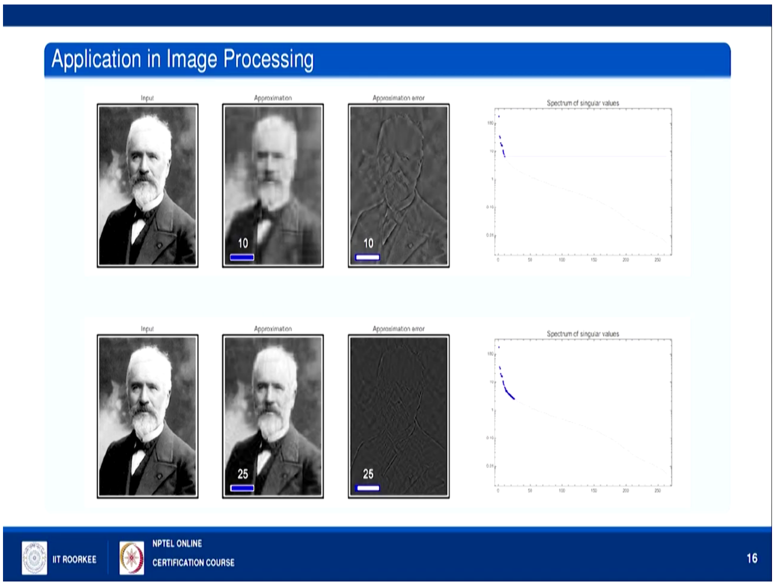


Now, see this is my original image input image which is having 266 nonzero singular values. So, if I take only sigma one that is the largest singular value then what I will be having? If it is a m by n image I am having singular value decomposition m by m into m by n into n by n.

If I take only first column of U for sigma ones and then that is the U 1 sigma 1 V 1 transpose then I got this approximation and this is the error, error means this minus this, or error means the image reconstructed from rest of the singular values; just ignoring the first one.

So, here it is not because you are having 266 rank and this is just rank one approximation if you go to rank four approximation you can observe we are having some pattern which are there in the input image. If we are taking more singular values we are having better and better approximation. This is just a spectrum, this is the first singular value and these are the first 4 singular values.

(Refer Slide Time: 29:07)

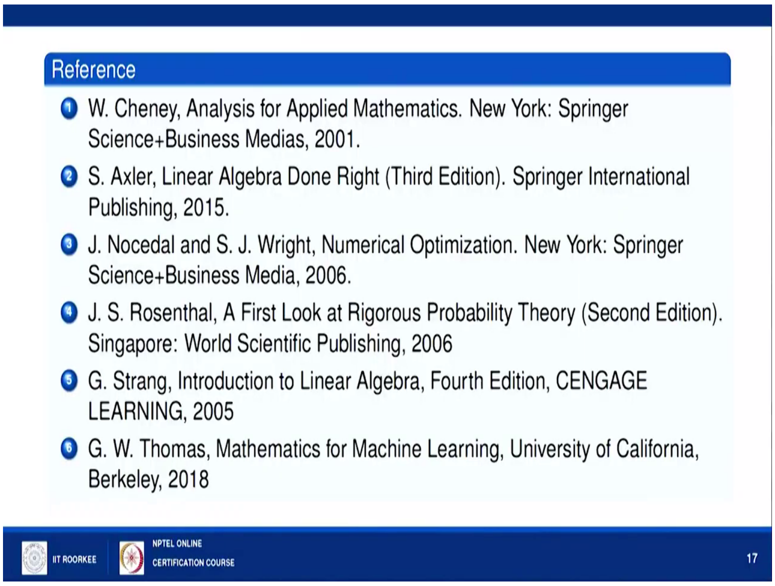


Now, this is the rank 10 approximation means, when you are taking the top 10 singular values and you are reconstructing, or approximating your image 266 rank image to 10 rank 10 image and this is the error and these are the spectrum of those 10 singular values. This is an example of if you are taking 25 top singular values in the approximation instead, of 266.

Originally, I was having 266, but here I am using only 25 so you can see, how much pixels I can ignore and if you compare these 2 image we are quite close to original one. So, what I can say that top 25 singular values give a significant approximation.

So, in that way in image processing we can use make use of this low rank approximation in image compressing where we have to save image with low storing power or using the low memory. Similarly, we can make use of this concept in machine learning means we can learn instead of the original feature space 2, the lower feature space with low rank approximation.

(Refer Slide Time: 30:39)



So, I hope you have enjoyed this lecture these are the references.

Thank you very much.