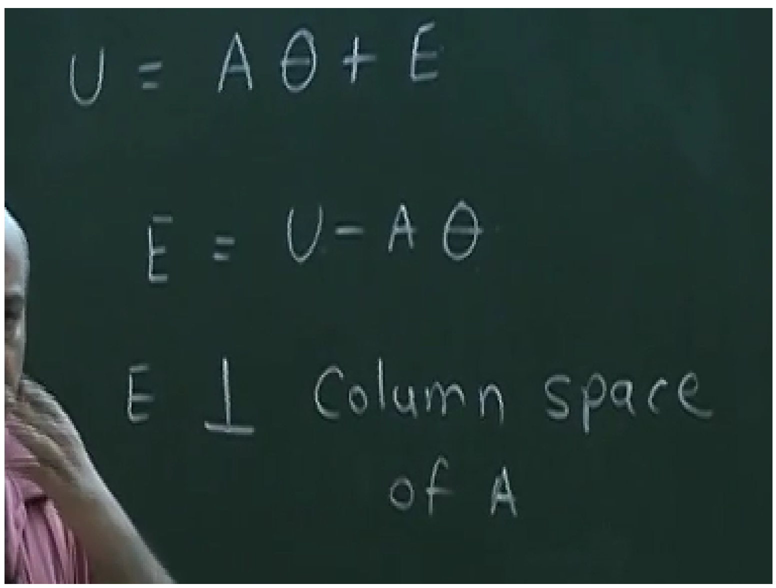
Advanced Numerical Analysis
Prof. Sachin Patwardhan
Department of Chemical Engineering
Indian Institute of Technology- Bombay

Lecture - 20
Geometric Interpretation of the Least Square Solution (Contd.) and Projection Theorem in a Hilbert Spaces

Okay so just a quick review of what we have done last time. We derived the formula for least square estimation only through geometric arguments. So, we said that the error between the predicted and the known are measured by this. Predicted by the model and the known values of the dependent variable that error should be perpendicular to the subspace spanned by columns of matrix A.

(Refer Slide Time: 01:02)



So, we had this model which was U=A theta + E, E is the error vector, so we just made geometric arguments something that you know from your school is this error this error vector U-A theta this should be perpendicular. This should be orthogonal to the column space of so E should be in the least square estimation, E is perpendicular to the column, what is column space of A? span of columns of A column space is nothing but span of columns of A.

So, this error should be orthogonal to this just using the simple argument that is argument of projection we arrived at the basic formula that was theta least square=A Transpose A Inverse A

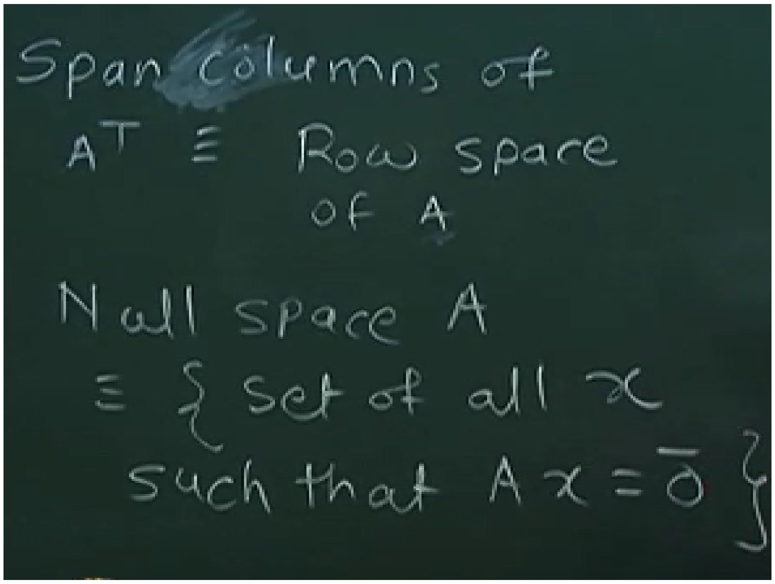
Transpose U. So, we could arrive at this fundamental result which earlier we arrived through algebra through geometry. Okay so today what I want to do is extend this beyond finite dimensional spaces also give you some more insights.

Geometric insights in the finite dimensional spaces next class after this we will start applying it for different purposes so again let us again stick to the interpretation part of it as I said there is a third way of interpreting all these things and it is through statistics. Unfortunately, we do not have time to get into that, but I am going to just upload my notes about statistical interpretation. So, those of you who are interested should go and look at the notes but that is extra reading.

And this is a post-graduate course, so you should do extra reading it is not that just to end the things here. What we actually managed to cover in this course is the tip of the iceberg. So, we derived this formula okay and I want to generalize this to some extent to any Hilbert space approximation in any Hilbert space. Before that let us look at some geometric insights. Now I just now defined column space of A.

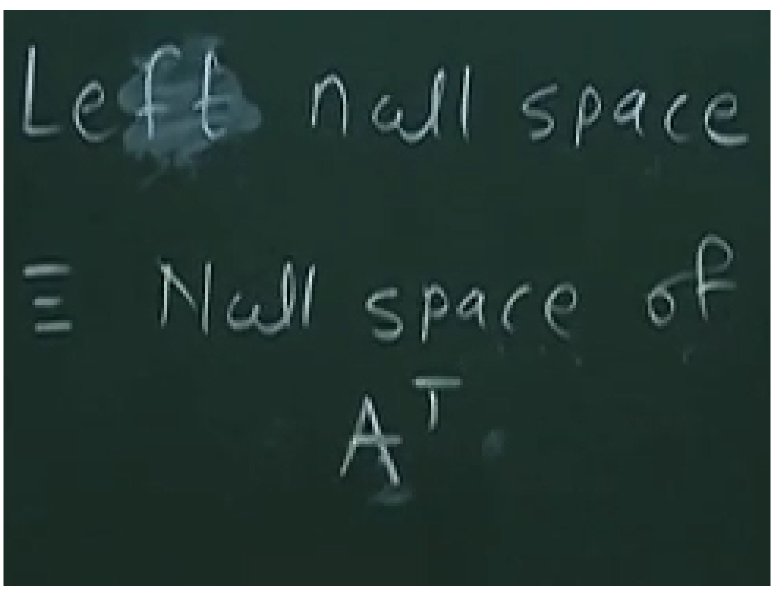
Probably in some other course you also have looked about other spaces that are associated with the matrix, what are the other spaces that are associated with the matrix null space, what about row space, what is the row space span of rows span of rows okay or span of columns of A Transpose. Okay so column space of an A Transpose is nothing but the row space. So, we just need to know a little bit about these 4 spaces.

(Refer Slide Time: 04:11)



So, given the matrix A okay span of columns of A is called as column space of A then span of row space of A row space of A is nothing but span of A Transpose or set of all possible linear combinations of rows okay, what is null space of Ax=0. So, mull space of A this corresponds to say set of all x such that Ax=0 vector. So, set of all x such that Ax=0 vector this is called as null space of A. And there is one more space associated with matrix that is called as left null space okay so left null space.

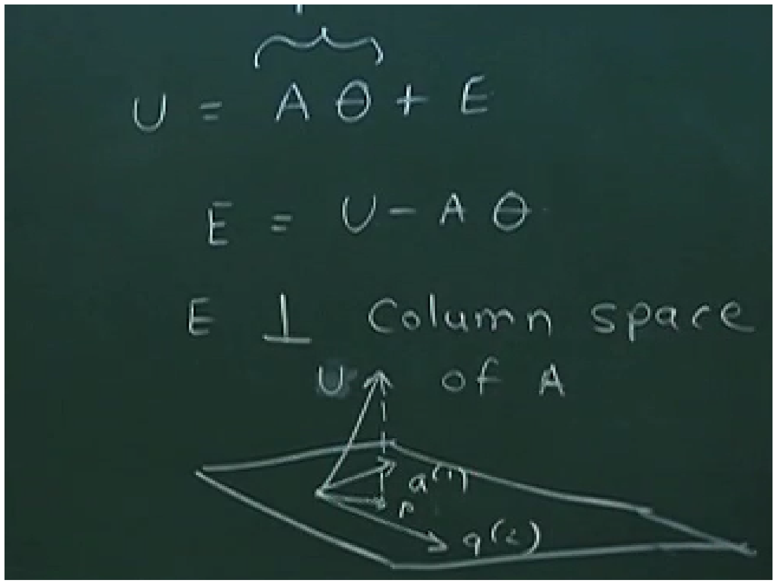
(Refer Slide Time: 06:08)



Left null space is nothing but null space of A Transpose, null space of A Transpose null space of A Transpose is called as left null space of A. Okay there are 4 fundamental subspace associated with a matrix okay one of them is the column space column space is all possible linear combinations of columns of A, row space span of row vectors null space set of all x such that Ax=0 and left null space that is set of all vectors y such that A Transpose y is=0.

Okay so this is there are 4 fundamental subspaces and what you are going to see or what you will realize here when I go to this okay we split a vector u this vector u was split into 2 components.

(Refer Slide Time: 07:31)



One component was the projection was the projection p. So, we are able to split a vector by this approach geometrically what we are doing we are splitting the vector into 2 components, what are the 2 componnets,1 is in the column space this is in the column space. Because A theta is nothing but linear combination of columns of A Right A theta is linear combination of columns of A.

So, this vector has to belong to column space of A. Okay if I just draw just draw this picture again here is what if you have a 2 dimensional if you have a matrix with 3 cross 2 matrix with 2 linearly independent columns. Let us say this is the first column a1 and this is a2 and this is my u vector this diagram I had drawn earlier. This is my u vector, so this is the projection this is the p vector this is the projection vector which is lying in the column space of a.

Okay then there is an orthogonal component, what do you mean by orthogonal component orthogonal component lies in the null space of this matrix. The orthogonal component lies in the null space does it lie in the null space or does it lie in the left null space just think about it where does it lie. Where does the orthogonal component lie, does it lie in the null space or left null space.

So, we are able to split a vector into 2 components in the column space and orthogonal to the column space orthogonal to the column space. Okay will a vector belonging to null space be orthogonal to the column space what is what is the what is the what is a property of vector that is orthogonal to the any vector that is orthogonal to the column space **"Professor-student conversion starts"** (()) (10:13).

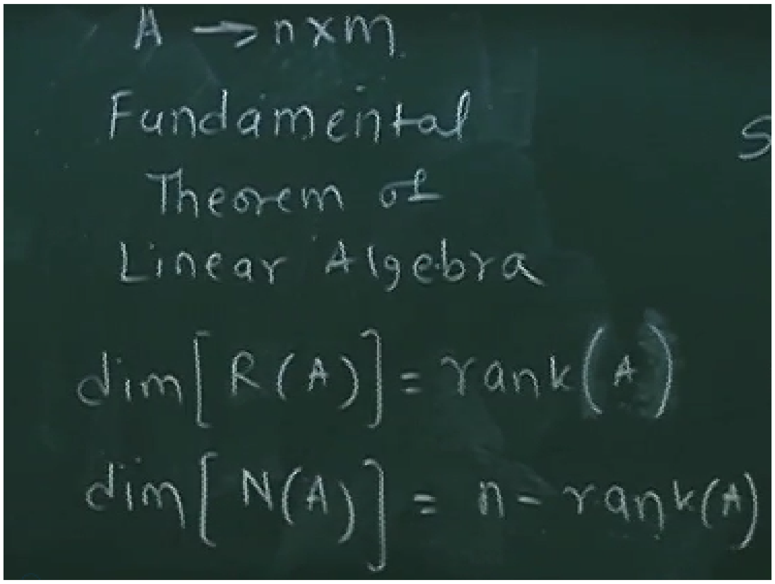
Yeah A times that vector okay A times that vector will give me 0 vector okay of course E is not in the null space, E is in the left null space just think about it E is in the left null space E is in the left null space and p is in the column space p is in the column space okay **"Professor-student conversion ends".** Okay let us there is a fundamental theorem of linear algebra okay I am just going to mention it here without getting into the proof you can refer to Strangs book.

You can refer to a linear algebra book for this. Now the column space of A is often referred to as R of A. The column space is often referred as R of A and the null space is often referred as N of A, N stands for null space A stands for null space of A. So, there is 1 fundamental theorem of linear algebra which says that I think you know this you know this result.

I am just going to say I state it again number of linearly independent columns okay is=number of linearly independent rows. In any matrix number of linearly independent columns is always= number of linearly independent rows which is=rank okay which is=rank. So, this is the algebraic result which you know which is also geometric result. Because linearly independent columns okay it relates rank which is an algebraic quantity with a geometric property.

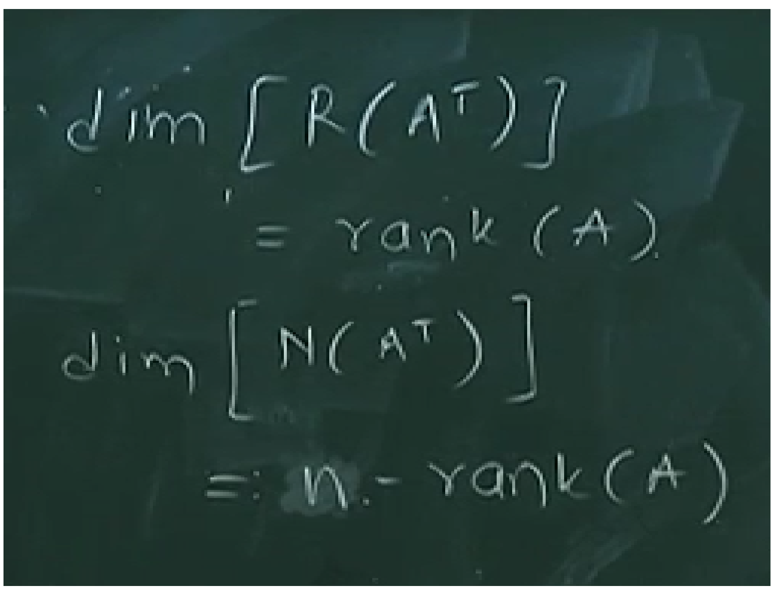
That linear independence okay so what is the dimension of the column space dimension of column space is always equal to rank.

(Refer Slide Time: 12:55)



Okay so the fundamental theorem of linear algebra dimension of row space of A dimension of column space of A dimension of column space of A= rank of A. Okay I am looking at a matrix A which is n cross m. I have a n cross m matrix so dimension of column space of A=rank of A, dimension of null space=n-rank of A.

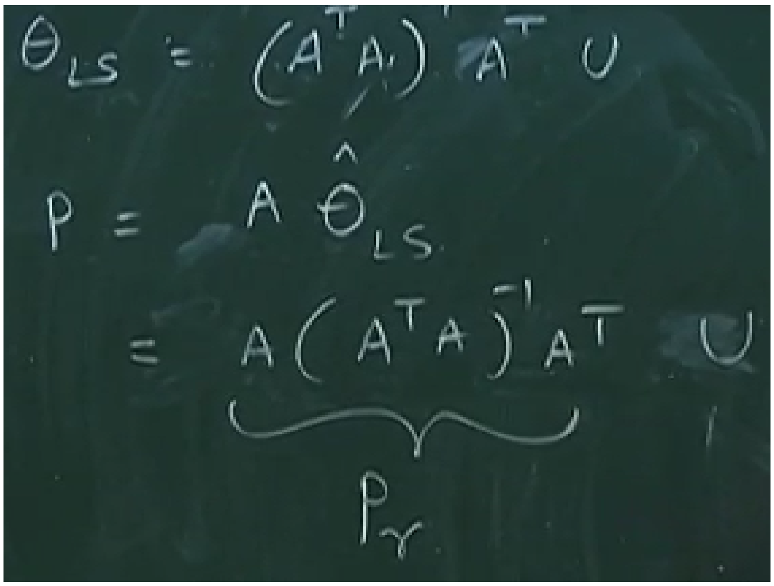
(Refer Slide Time: 13:41)



Then dimension of row space of A Transpose sorry column space of A Transpose that is nothing but row space=rank of A and dimension of null space of A Transpose **"Professor-student conversion starts"** m-rank a number of columns is m yeah yeah if I put n cross m it should be and n- alright and then the should be and n- agreed it should be n-ranking **"Professor-Student conversion ends".**

Okay this is just the background this is well known result I am just checking it again okay now from this background I am going to prove some nice algebraic properties of projection that we have looked at. So okay now actually what we have achieved is a way of splitting a vector into 2components see we found out a least square estimate right we found out a least square estimate and how did we get the least square estimate how did we get the least square estimate.

(Refer Slide Time: 15:14)



So, theta least square this was A Transpose A inverse A Transpose U correct this is how we got the least square estimate. Okay so what is the projection see we are talking of 2 things right 1 is this projection onto the column space or there is a vector error which is perpendicular to the column space okay so so how do I project given a matrix A how do I project it on to its column space okay so the projection p is given by A times theta hat LS right.

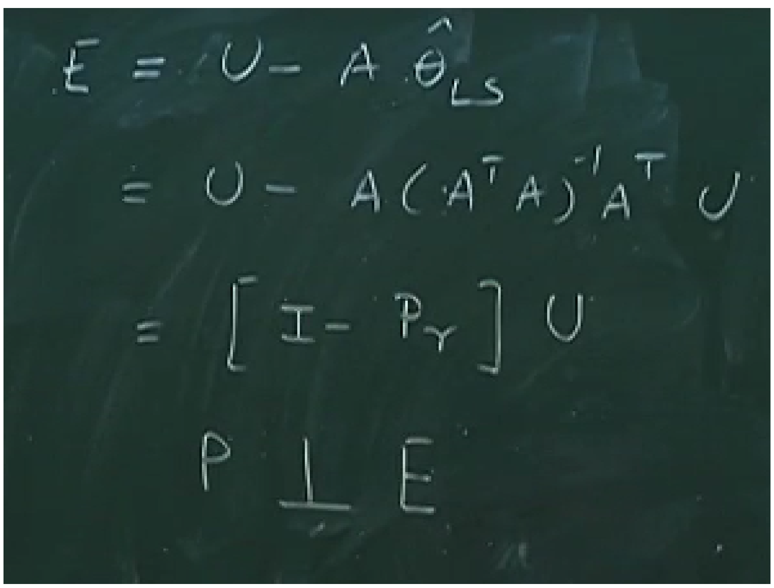
The projection the found out the theta which projects now projection vector itself what is the projection vector A times theta. Okay so this is nothing but A Transpose A Inverse A Transpose U correct. But A Transpose A Inverse A Transpose U okay if i call this matrix as projection Pr matrix, let me call this matrix as Pr okay where does p where does this vector p lie the vector P lies in the column space of A.

So, given a matrix A how do I project any vector onto its column space. I take matrix A i find out this i find out this matrix this is called projection matrix this is called projection matrix this projection matrix projects any vector U onto the column space of A. Okay see because look at this see theta here see theta happens to be this. But what is theta theta being a least square estimate is some number.

So, A times theta is nothing but linear combination of columns of A. So P has to lie in the column space of A. Okay so actually what we have done is we have found out a way given a matrix A which is non-square which is actually tall matrix which is it has more rows than more than the columns for such a tall matrix we know how to project any vector U onto the column space okay this is the component along the column space.

What is the component which is orthogonal to the column space?

(Refer Slide Time: 18:08)

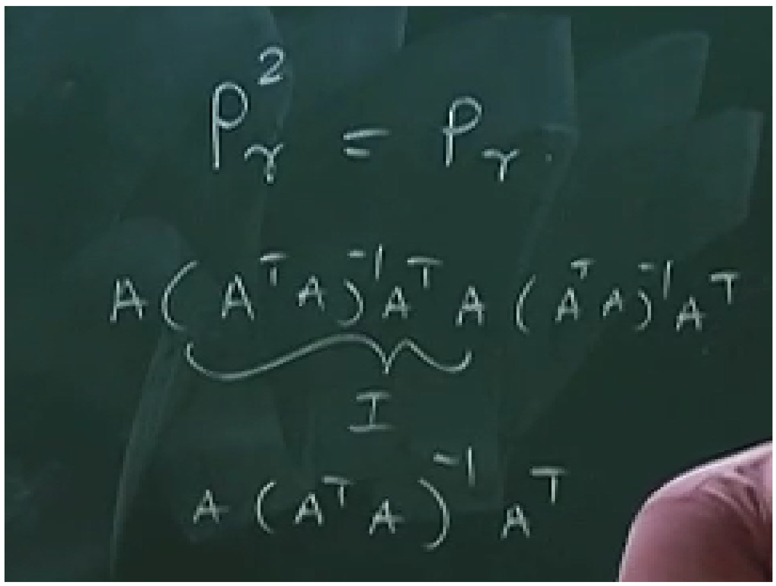


So that is E so how do you get E, E is U-A theta least square okay that is U-A A Transpose A Inverse A Transpose U which is I-projection matrix, what is this? I-projection matrix everyone with me on this I-projection matrix. Okay you can check that you can check that P is orthogonal to E take inner product of P and E you will get you will get 0 these 2 are orthogonal components. So, now given a vector U given a vector U given a vector.

We know how to split this vector into 2 components, 1 in the column space other orthogonal to the column space. Okay this matrix A*A Transpose A Inverse A Transpose is called as a projection matrix there is something very nice about the projection matrix, is projection matrix symmetric is it symmetric matrix? If you take Transpose of this matrix what will you get? I am not going to do it on the board just think about it.

There is one very nice property of this projection matrix I want you to check it out and I want you to understand what it is meaning this is a very funny matrix it is square of itself.

(Refer Slide Time: 20:03)



Yes, this matrix has a very nice property Pr square=Pr can you check this just multiply Pr*Pr you will get back Pr mind you this is not an identity matrix this is not an identity matrix. And you have to explain me geometrically what is meaning Pr*Pr=Pr first of all are you able to get this is is his coming out just, multiply what is A*A Transpose A inverse A Transpose A A Transpose A Inverse A Transpose Right A Transpose A inverse A Transpose A this will become identity.

You will get A A Transpose A inverse A Transpose this is not a matrix which is this is not a identity matrix, yet it has this very nice property it is square of itself, what is the geometric meaning of this? **“Professor-Student Conversion starts”** Yeah (()) (21:34) yes non-matrix vector u plane same vector precisely very very wise “**Professor-Student Conversion ends**. So, if you project the projection again you will get the same vector right.

See here this my U vector is split into 2 components projection and orthogonal to the projection, Now this projection if I project again onto the plane what should I get the same vector because there is no the best approximation of this vector inside this plane is this vector itself okay this vector itself. So the geometric interpretation of projection square okay you can just go on you can just say projection cube is also=very very nice matrix.

Okay vice versa any matrix that has this property is called as projection matrix that is square of itself is the projection matrix. Okay so this this is a this is a nice property of projection matrix you can on multiplying it with itself and you will get by the same matrix what about Transpose is a Transpose of this matrix is symmetric matrix okay here Transpose it is a symmetric matrix okay it is a symmetric matrix it is q square of itself.

Okay what will be projection matrix if your U is perpendicular just without computing you should tell me what the projection matrix should be if U is perpendicular projection matrix should be o because the best approximation-best approximation of U which is perpendicular to this plane is a 0 vector. Okay projections least square approximations associate this geometric ideas.

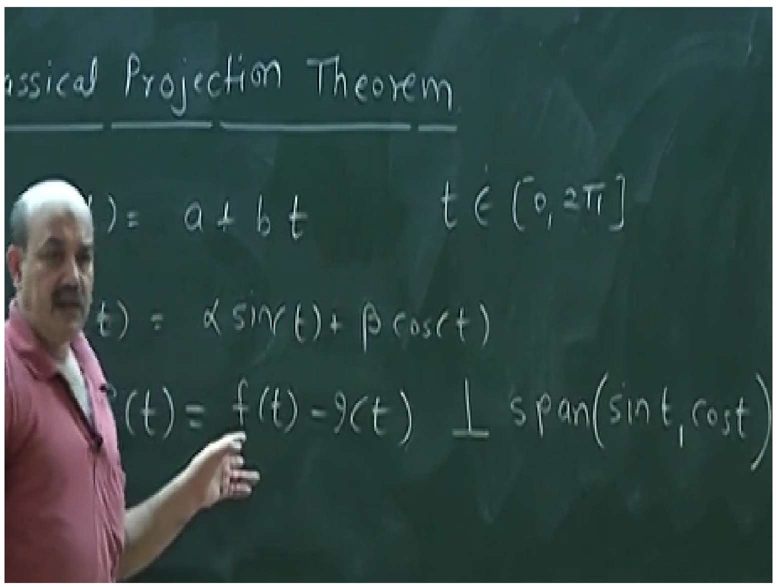
Then you will remember it better than just what happens if U is already inside we talked about it if U is already inside you it will get back the same vector same same matrix. So, it is projection matrix is and what happens, but it is square if A matrix is square matrix then what is the projection matrix. If A is a square invertible matrix If A is a square invertible what will be the projection matrix? A is a square invertible.

So, what is A Transpose A inverse it is A*A inverse which is I*A Transpose inverse*A Transpose inverse which is I identity matrix. So, if A is full rank projection matrix will turn out to be identity matrix. **"Professor-Student conversation starts"** There will be no error There will be no error you are already in the full space yeah **"Professor-Student conversation ends"** okay so I am going to generalize this to any Hilbert space this is a very nice result.

And then we need to you know extract its full potential. We should not just restrict our self to n dimensions till now I have been looking at only n dimensions why not go ahead and look at you know any Hilbert space. Any Hilbert any infinite dimensional space in fact what you will see is that soon what will pop out is Fourier series not just the Fourier series but generalized Fourier series.

Okay let us look at this very quickly because this is something which probably Fourier series wants you from the second year of engineering and then this will probably explain you the basis for how how it works why Fourier series why we look at orthogonal functions and so on. Okay now what I want to do is to generalize this result * any Hilbert space.

(Refer Slide Time: 26:17)



So, what we are going to look at now is see basically when you when you will study all these methods numerical methods for solving pressure differential equations boundary value problems algebraic equations whatever. I want you to remember all these geometric insights all this you know it is not it is not just the procedure. It is this foundation which help to modify to tweak to twist for a particular you know application.

Because if you want to if you want to really concoct a solution for a given problem you cannot many times just rely on the standard tools. You have to come up with some way of you know concocting a new solution procedure that is where you should understand all these fundamentals. Okay you should know where it is arising from, how whole thing was derived and then you can just derive another approach if you if you need to so that is critical.

Okay so just like a let us take a look at what we have done till now, what we have said that I have a way of projecting a vector okay. So, we looked at a 3 dimension we looked at a vector which is outside the plane you know and then suppose this is the plane. Now I have a way of projecting this vector onto this plane and a vector component a component which is perpendicular to this plane.

Okay in general, I said why just a 2-dimensional plane if I have a vector in RN and I have a subspace in RN. Okay I have a M dimensional subspace I know how to find a component lying in the subspace and orthogonal to the subspace right we looked at M dimensions. And N dimensional vectors being projected onto M dimensional subspace and a component which is orthogonal component which is orthogonal.

So, we were able to split in final dimensional vector spaces. We were able to extend our school geometry that you know the shortest distance of a point from a subspace or from a plane is draw a perpendicular. I want to do the same thing in any Hilbert space why can I do this? In Hilbert space I have an inner product with me since I have an inner product with me you know I have orthogonality.

I can talk about projections I can just talk about you know 2 vectors being orthogonal to each other. Okay i can just talk about now you know taking a subspace in a Hilbert space or taking subspace in an inner product space any space on which inner product is defined and then in that subspace. Okay and given a vector which is outside this space I can project this vector onto the subspace right.

I can find a component of the vector along in the subspace and the component perpendicular to the subspace, So, what was the trick, the trick was the component which is error okay our approximation error is orthogonal to the subspace. Okay so I find the error vector finding its orthogonal and you know that will help me to come up with approximations. See for example just to motivate you know.

Let us say i have this function ft a+bt a and b are some known values okay and then I want to approximate this function using. I want to construct an approximation. Let us say g*t which is alpha sin t+beta cos t is this a vector it I a vector let us say this is a vector where it belongs to 0 to 2 pi, is this a vector this function is a vector this is a continuous function over 0 to 2 pi. I want to construct an approximation alpha sint+beta cos t for this vector.

So, what do I mean by that the error what is the error function what is the error vector, error vector let me call this et will be ft- gt ft-gt. Okay what is the subspace here, what is the subspace that these 2 vectors span. It is a 2-dimensional subspace right 2-dimensional subspace of set of continuous functions between 0 to 2 pi, why 2 dimensional there are 2 linearly independent vectors sin t cos t okay there are 2 linearly independent vectors sin t cos t.

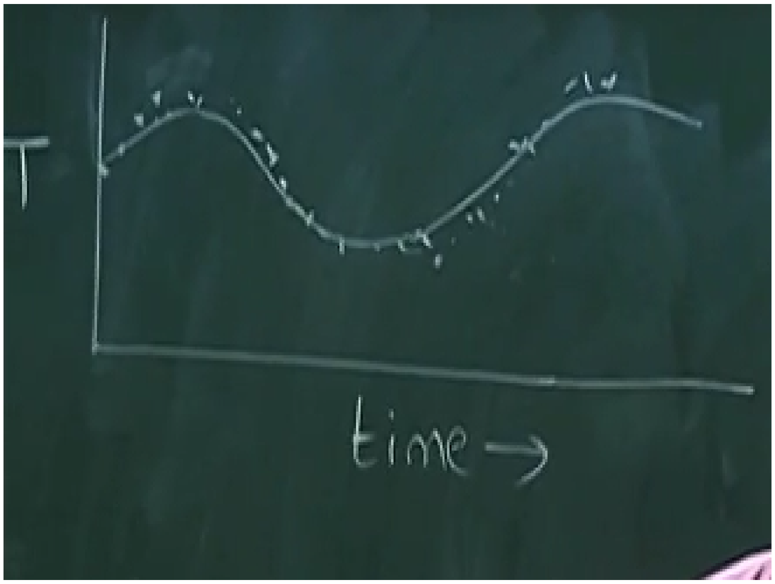
Now this we extend our ideas of projection what should happen this error function this error function should be orthogonal to subspace which subspace sin/sin t and cos t. 2-dimensional subspace sin which is spanned by all possible linear combinations of sin t and cos t. So, this should be perpendicular to span of sin t cos t this error this approximation error between the function original function and the approximation.

Okay see this a and b are known constant. Let us take as you know 1+5 t, let us take specific a a nd b are not known. We have to find out a and b least square approximation. Okay I want to get a least square approximation so i want to get at least square approximation of this. I am just taking very simple function in general you may have very complex functions which you need to approximate.

For example, you may have data which is actually coming from you know daily temperature variation which cyclic right daily temperature variation is cyclic you could approximate using sin and cos functions. You could develop very nice approximations so so i want to develop a approximation alpha and beta which is which what is the advantage of let us say that I have a data for temperature variation throughout the day okay throughout the day.

Okay what is the approx., what is the advantage of fitting see what I mean is you know i collect temperature at every 5 minutes and I plot it.

(Refer Slide Time: 33:33)



See let us say it is you know every 5 minutes I have plotted you have data for 5 minute for 24 hours how many data points ,12 per hour*24. So, many data points if I have to save this data for years. Let us imagine how much data I have to save okay as against if I am able to fit sin and cos okay I fit some sin and cos which is approximation which does not pass through every point. Okay nevertheless, it sorts of gives me a best fit least square fit.

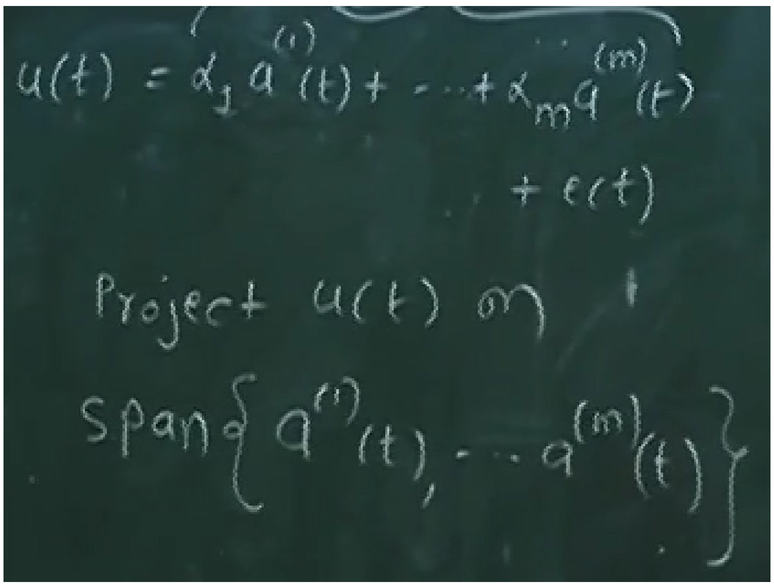
Okay for today suppose I have to save data for today for todays data okay I could find alpha and beta for todays data. I have to say only 2 numbers you must have heard about data compression right you have heard about data compression this is a data compression algorithm. Data for five minutes for temperature for the entire day can be compressed and written in terms of only 4 numbers or 2 numbers.

If you are not happy with sin t, cos t adds at 2 more you know sin2t cos2t okay and then you will be able to get an approximation of temperature versus time and instead of saving all this 12*24 values for 1 day. I could just say 2 approximation coefficients. All that I have to know these 2 approximations co efficient correspond to sin and cos. I multiply them I can get a very good approximation of what happened on that day right.

It may not be accurate to the extent; it may not be passing through every point does not matter but i can get a trend okay I can get a trend. So, this is very very important very very important okay so let us leave this aside for a while. Let us look at this problem we want to get geometric insights and we want to extend them to okay. So, now what should happen is this error should be perpendicular to the span of these 2 vectors.

Now in general in general my model could be of this form.

(Refer Slide Time: 35:58)



Okay let me write a general model, my general model is ut=this is my general model, okay I will just make 1 small change here just to be consistent with our notation, let us call this as p because this is the projection this is the projection let us call it as p okay and then here let me call this as ut. Okay now let us come back here so in general in general my ut okay is some linear combination of these known vectors a1t a2t a3t amt are known vectors.

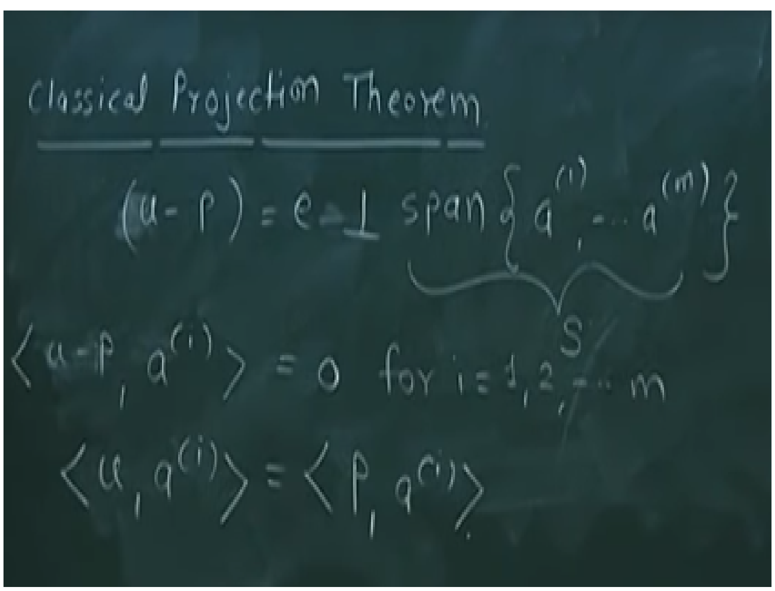
Just like sin t cos t okay you want to project this on to some known subspace okay you know you know it is basis vectors okay and linear combination of these basis vectors will define that subspace, what is all possible linear combination of this a1 to am? If they are linearly independent, it will be M dimensional subspace right it will be M dimensional subspace. Now the question is if i give u if i give you ut.

Okay can you find out alpha 1 alpha 2 which are least square. How do you find out alpha1 alpha 2 which is least square so this part this part here is the projection this part here is the projection onto onto the subspace defined by a span of a1 to am right? So, I want to find out projection project ut on. Okay now just for the sake of convenience of the notation I am going to drop this tttt I am just going to work with up a1a2 t is there.

I am just going to drop it for the sake of convenience otherwise the writing will become cluttered what I am saying here and what I told you here is not different just remember this this is a 2-dimensional example. Okay here so this is my a1 this is my a1t and this is my a2t this is my a1t and this is my a2t the subspace which is fine by linear combination of these vectors is 2 dimensional.

In general, you may have M such vectors and you are talking about distance of a point from the subspace. Same problem okay visualization will not with relation you like big different from what you know in 3 dimensions, but I am extending it to any M dimensional subspace of an infinite dimensional space. Okay so the projection theorem I am not the the statement is formal statement is given here.

(Refer Slide Time: 39:35)



The projection theorem says that you know u-p that is=error is orthogonal the statement of the projection theorem classical projection theorem on Hilbert spaces is that error is orthogonal to the error the point which is at the least distance from the point in the subspace which is at the least distance from the a point outside can be reached by dropping a perpendicular which you know from 3 dimensions.

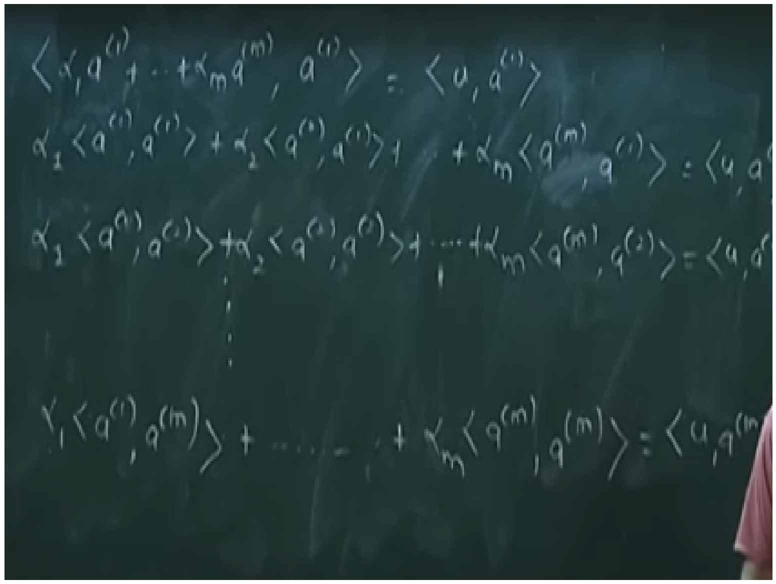
Same result holds here that error is perpendicular to the subspace okay that is the statement of the theorem formal statement is given here. I am just compressing and writing here that this error vector is orthogonal to the span. So, this is my let us call this subspace s okay it is orthogonal to the span. Now how am I going to use this to compute alpha 1 to alpha m. Okay so what I am going to do is very very simple.

Okay I am going to start i know that this is orthogonal to a1 to am. Okay which means this error is orthogonal to a1 is also orthogonal to a2 is also orthogonal to a3. So, one by one I am going to write these equations. Okay so I am going to say my first equation is that inner product see I am just generalizing thus result. I am saying inner product of u-p*ai=0 right u-p is this okay what is the meaning?

What is the meaning of this statement u-p this error vector is orthogonal for i=12 up to m. How many equations I need how many unknowns are there are m unknowns alpha 1 alpha 2 alpha 3 up to alpha m there are m unknowns. Okay I need to write m equations okay here are my m equations. I am just going to use them okay so just look at this well I will just rearrange this and write it and I will say that this means that inner product of uai=these 2 are same equations right.

I am just rewriting them I am just expanding and writing this equation. So, I am going to use this now to create. So, what is this?

(Refer Slide Time: 42:43)



Okay so this equation first equation is that you know alpha 1 a1+upto alpha m am inner product a1=u inner product a1 right right so this equation will actually be can be written as alpha 1 a1 inner product a1+alpha 2 a2 inner product a1+alpha m am inner product a1=u inner product a1 right. I am just expanding this equation I am just expanding this equation, what is my second equation?

So, second equation if I write just like this and expand what will I get? I will get alpha 1 a1 inner product a2+alpha 2 a2 inner product a2 alpha m am inner product a2=u inner product a2. Okay how many such equations I have. I have M equations okay so alpha1 a1 inner product am up to finally up to alpha m am inner product am=u am. Is everyone with me on this. How many equations you have got M equations in M unknowns.

Okay do I know these vectors a1 a2 a3 am, I know these vectors I have chosen them sin cos okay sin 2 cos 2 and so on. So, I know this vector so if I if I know these vectors these inner products can be computed whatever integral over 0 to 2pi. You can compute them okay you can compute them and then this equation can be solved this equation can be solved because you have M equations in M unknown, is it a linear equation.

It is a linear equation it is like matrix times vector=some vector. U is known to you a1 to am are known to you, so right hand side is known to you, all these coefficient or inner product are known to you alpha 1 to alpha m is not known Okay now see what is the advantage of choosing these vectors to be orthogonal suppose a1 a2 a3 am are orthogonal what will be the inner product okay which inner product will not be 0 a11 a22 a33 a44.

Yes, if you choose orthogonal vectors okay the solution is particularly very easy then you will get first equation will be alpha 1 times this=actually what we have derived is nothing but generalized Fourier expansion. I will just come to this in my next lecture very briefly, this is if you if you take a1 am to be orthogonal what do you get are nothing but Fourier co efficient. This is called as generalized Fourier expansion and then we will get Fourier coefficient.

Fourier expansion need not be only with sin cos tomorrow in some function you know you want to approximate some function over some domain set 0 to 1 using shift and general polynomials why shift general polynomials shift and general polynomials are orthogonal polynomials okay and then this approximation will be very easy. so, we will continue on this particular equation is called as normal equation and this is like cornerstone of optimizations projections.

And helps us in discretizing and so on. Okay so actually if you go back and look little bit carefully this is a lot different from A Transpose A inverse or A Transpose A*Theta= A Transpose U because inner product of columns A Transpose A, if you take columns of A inner product of the same equation written in a generic form for any Hilbert space same equation not different okay. So, let us stop here and then let us continue with its interpretations in our next lecture.