Introduction To Adaptive Signal Processing

Prof. Mrityunjoy Chakraborty
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture No # 39

Derivation of the RLS transversal adaptive filter

So, we will be deriving this recursively squares algorithm. So, we have already defined and seen these things actually x n can be partitioned like this. This was where sorry ok. There was one partition another is d n as we know it was d n minus 1 then d n. So, this part you can write as d vector with n replaced by n minus 1 and then current component d n.

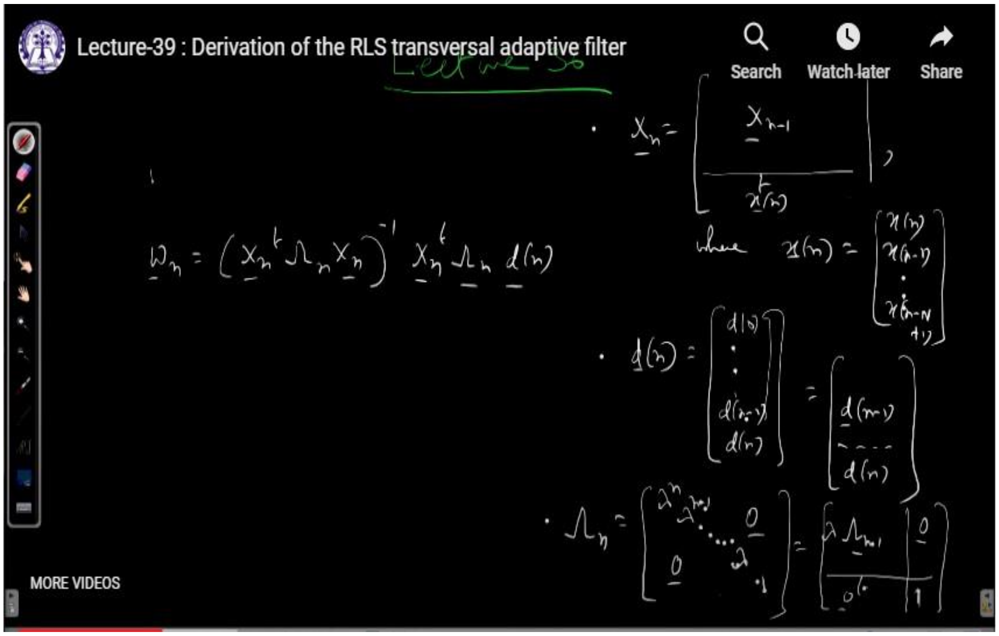
$$\underline{X}_n = \begin{bmatrix} \underline{X}_{n-1} \\ \underline{x}^t(n) \end{bmatrix} \text{ where } \underline{x}(n) = \begin{bmatrix} x(n) \\ x(n-1) \\ \vdots \\ x(n-N+1) \end{bmatrix}$$

$$\underline{d}(n) = \begin{bmatrix} d(0) \\ \vdots \\ d(n-1) \\ d(n) \end{bmatrix} = \begin{bmatrix} \underline{d}(n-1) \\ \underline{d}(n) \end{bmatrix}$$

Another one lambda n it was lambda to the power n this we can write as this is 0 as though this is 0 vector its transpose this much this is 0 column vector this is this 1 ok. So, I am basically partitioning last row is this of which I have all 0s here and just a 1 and last column all 0s here this 1. So, this upper half lambda lambda square lambda cube to the power lambda n. So, lambda you can take common.

So, it will be 1 lambda lambda square this will be lambda to the power n minus 2 because 1 lambda has been taken out this will be lambda to the power n minus 1. So, that will be this matrix at n minus 1 this is important. What is this at n minus 1? It will be 1 lambda

lambda square dot dot lambda to the power n minus 2 lambda to the power n minus 1 because this index is n minus 1 that is what I have. If I take out this 1 here all of them have lambda common if I take lambda out then this will be 1 then lambda lambda square this will be lambda to the power n minus 2 lambda to the power n minus 1 ok. So, this sub matrix will be this lambda n minus 1 sub matrix 1 lambda has been taken out as common.



So, this is what this we will be using and what we will be going for we know W this is what we have to calculate from Wn minus 1 recursively not by using the formula, but Wn minus 1 also likewise given by something like this all right. So, from Wn minus 1 we must generate the Wn recursively by simple update equation ok. Up till n minus 1 I had only Xn minus 1 data matrix Dn minus 1 data vector and this was lambda n minus 1 matrix ok. At nth clock new data has come X of n using that I build up this Xn and add a row. Similarly new Dn minus 1 has come so add this extra element at the bottom and this matrix enlarges to this ok.

$$\underline{w}_n = \left(\underline{X}_n^t \underline{\Lambda}_n \underline{X}_n\right)^{-1} \underline{X}_n^t \underline{\Lambda}_n \underline{d}(n)$$

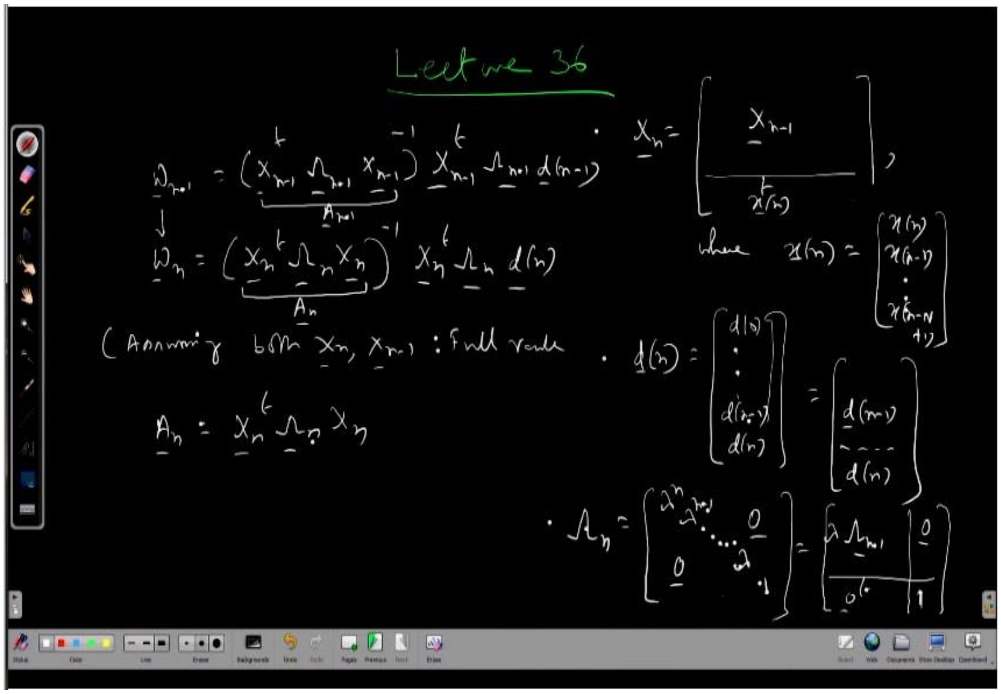
$$\underline{w}_{n-1} = \left(\underline{X}_{n-1}^t \underline{\Lambda}_{n-1} \underline{X}_{n-1}\right)^{-1} \underline{X}_{n-1}^t \underline{\Lambda}_{n-1} \underline{d}(n-1)$$

Using this additional information Wn is to be calculated, but not by this formula, but directly from Wn minus 1 because there are too many things common between Wn and Wn minus 1 that is these matrices Xn to Xn minus 1 ok this much is common Xn minus 1 is common Dn and Dn minus 1 Dn minus 1 part is common and likewise this part we have to do. And of course, we are assuming both Xn Xn minus 1 full rank if there is invertible full rank so that this is invertible ok. Let me call this product An so this will be An minus 1. So, I am assuming An and An minus 1 both are invertible An is ok this is my An, An minus 1 is just replace n by n minus 1 n by n minus 1 n minus 1 and both cases inverse is coming. So, I am assuming that they are all invertible that is small n time index is at least equal to capital N or more ok not before that.

$$\underline{A}_n = \underline{X}_n^t \underline{\Lambda}_n \underline{X}_n$$

$$\underline{A}_{n-1} = \underline{X}_{n-1}^t \underline{\Lambda}_{n-1} \underline{X}_{n-1}$$

What happens before that that we see later I have to take care of that that we will see later. So, this is my An. So, a actually Wn is An inverse say this part.



So, An so what is Wn is An inverse Dn right. Now what is An? An was Xn transpose.

$$\underline{w}_n = \underline{A}_n^{-1}\underline{X}_n^t\underline{\Lambda}_n\underline{d}(n)$$

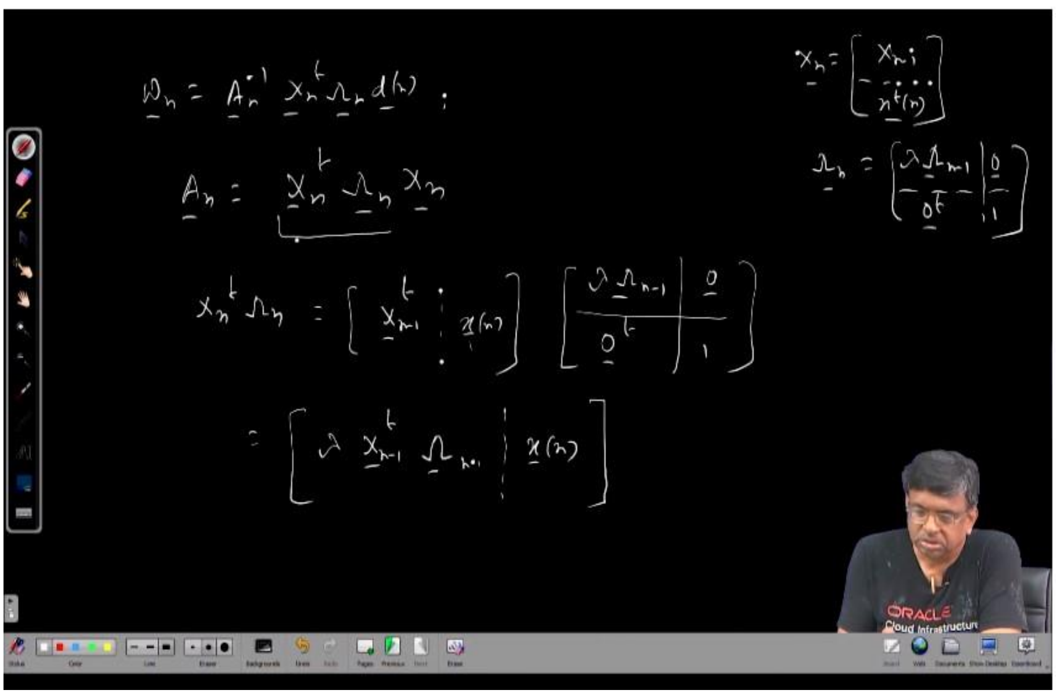
$$\underline{A}_n = \underline{X}_n^t\underline{\Lambda}_n\underline{X}_n$$

So, let us first do this part Xn you remember was Xn minus 1 it was partitioned like this and this was it was done in the previous page. So, no need to explain again. So, if I carry out this part this into this. So, Xn transpose this is Xn Xn transpose means first row becomes first column second row becomes second column like that and this last row becomes last column. So, it will be first row of Xn minus 1 is first column here first column here second row of Xn minus 1 second column here dot dot dot ok.

And last row will be last column say X transpose n is a row vector its transpose you know this will Xn this will come here alright X transpose n is this row. So, when you make a column of it you have to take a transpose of that that is what Xn. So, this is Xn transpose and delta n I am reproducing here again and we had just seen that block wise partitioning of matrices and their multiplication. So, this will be this matrix times. So, first I consider this part.

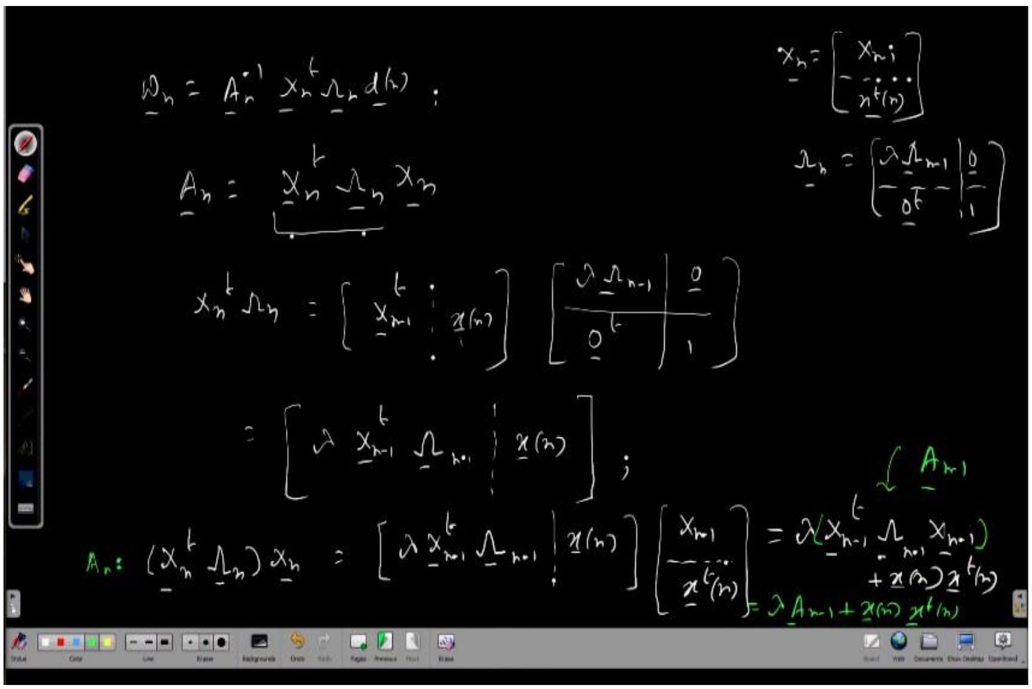
So, this matrix times this plus this matrix which is the vector times this. So, this will be this matrix term this means lambda is a scalar. So, lambda I can take in the front then this matrix times this plus Xn into 0 Xn is a column vector 0 transpose is a row vector. So, column every element this column into a 0 will give a 0. So, it will be a all 0 matrix Xn is a column vector 0 transpose is a all 0 vector row vector when you multiply you get a 0 matrix with all elements 0 very simple.

So, that when added with this contributes nothing. So, I am not adding that and then this matrix times 0 column vector which is 0 plus this column vector times 1 which is Xn. So, it will be Xn. So, this much then I have this product times Xn. So, this times this times Xn and Xn is equal to 0.



So, this is the product times Xn. So, I use this partition again. So, that means, this sub matrix times Xn minus 1 plus this sub matrix which is column vector times X transpose n. So, it will be lambda plus Xn minus 1 and what is this part? By definition this entire thing is a n minus 1, a n when Xn transpose lambda n Xn a n minus 1 Xn minus 1 transpose lambda n minus 1 Xn minus 1. So, this actually lambda times a n minus 1 plus this extra component.

This is very interesting this shows how a n we started with a n how a n comes from a n minus 1. This is your a n, a n comes from a n minus 1. This lambda times n minus 1 plus an extra term. What is the extra term? Current input vector into X transpose a very simple thing, but I am actually interested in inverse of a n. So, I will have to take inverse of a n that is inverse of right-hand side and here I will be using that matrix inversion lemma.



So, again I go here you remember matrix inversion lemma was this for your sake I am writing once again assuming a to be invertible and this also invertible. So, it was this times this is what we have derived and now we have got with the properties of the previous page a n equal to lambda n minus 1 plus Xn X transpose n all right. That is I have a n equal to lambda times a n minus 1 plus again let us see lambda times a n minus 1 plus Xn X transpose n. So, this is what we have derived. So, this is what we have derived, but I am interested in a n inverse I am assuming both a n and a n minus 1 they are invertible.

$$\underline{A}_n = \lambda \underline{A}_{n-1} + \underline{x}(n)\underline{x}^t(n)$$

This is invertible that the total is invertible that is what I am assuming all right. So, a n inverse then will be if I take lambda out I take lambda common out plus lambda inverse of lambda times lambda n minus 1 plus Xn X lambda inverse lambda is not 0 it is between 0 to 1. So, lambda inverse no problem. So, this is like your a this one alpha is lambda inverse a is Xn and you have to invert. So, when you apply a n inverse lambda also becomes lambda inverse lambda inverse because lambda also is a matrix 1 by 1 matrix lambda inverse which then you write a scalar lambda inverse and then put anywhere as in the beginning and the end and inverse of this apply the lemma here inverse minus lambda inverse is alpha by 1 plus Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn Xn.

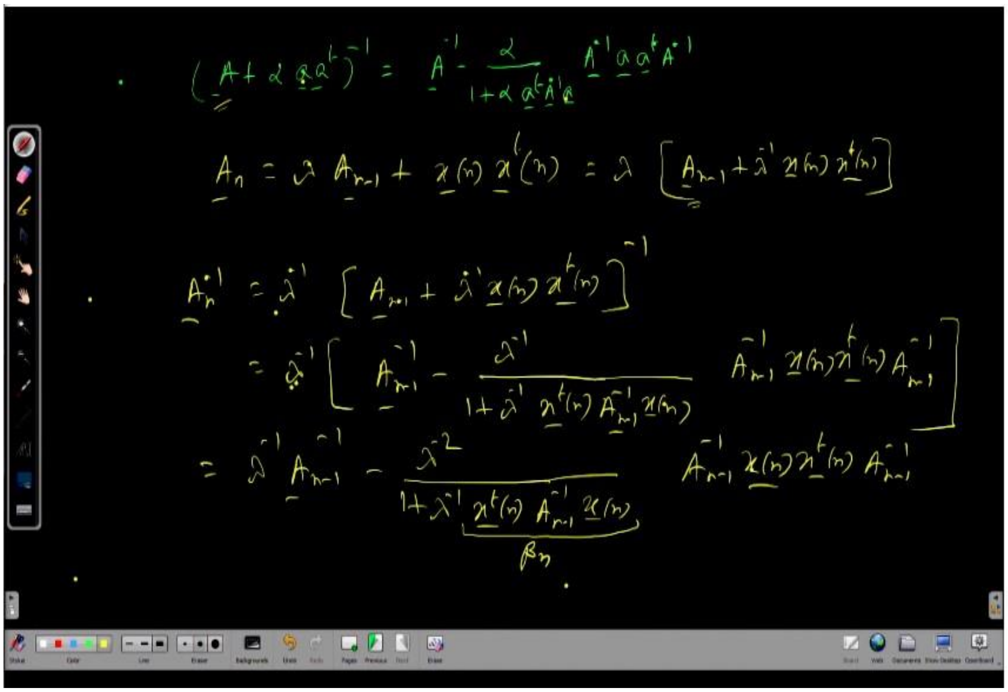
$$\underline{A}_n = \lambda[A_{n-1} + \lambda^{-1}\underline{x}(n)\underline{x}^t(n)]$$

So, this is a transpose again lambda inverse a is Xn. So, a transpose that is X transpose n, a is a n minus 1 inverse a inverse a is a n minus 1 here and again small a small a is Xn this much into this. So, this is a inverse a is Xn Xn Xn Xn Xn Xn. So, this is a n minus 1 here inverse Xn is a and now bring lambda inverse inside because lambda inverse lambda to the power minus 2. So, let me take this to be this remember this matrix is a column vector.

$$= \lambda^{-1}\left[\underline{A}_{n-1}^{-1} - \frac{\lambda^{-1}}{1 + \lambda^{-1}\underline{x}^t(n)\underline{A}_{n-1}^{-1}\underline{x}(n)} A_{n-1}^{-1}\underline{x}(n)\underline{x}^t(n)\underline{A}_{n-1}^{-1}\right]$$

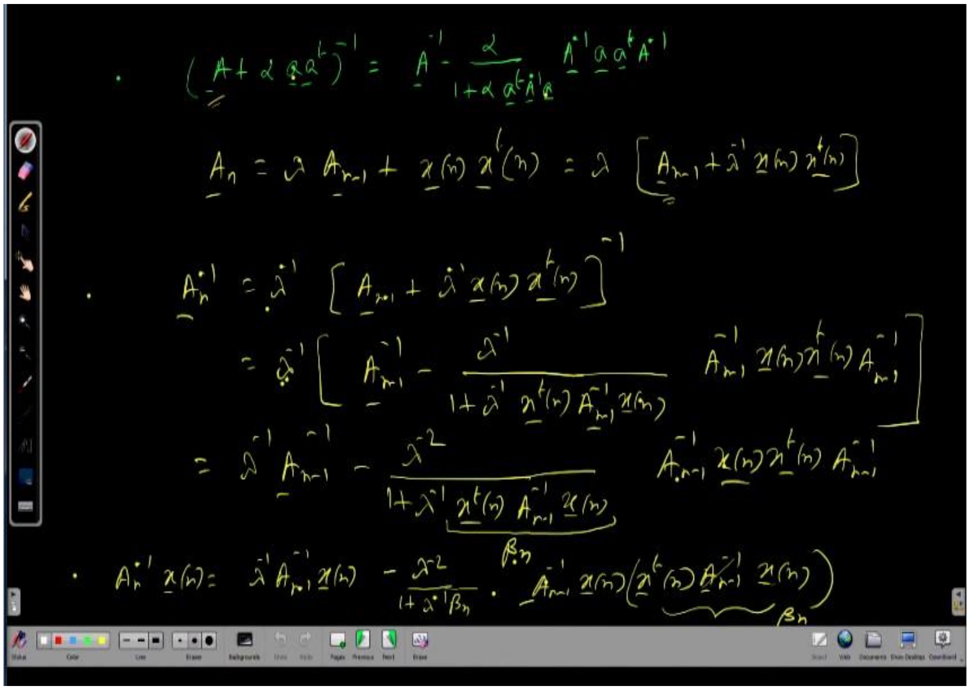
So, matrix into column vector is a column vector is a row vector row into column is scalar. So, let me give this scalar a name beta n. So, notational it will be easier all right. So, it is lambda inverse into this minus lambda to the power minus 2 by 1 plus beta n times this thing this one relation you have found out.

$$= \lambda^{-1}\underline{A}_{n-1}^{-1} - \frac{\lambda^{-2}}{1 + \lambda^{-1}\underline{x}^{t}(n)\underline{A}_{n-1}^{-1}\underline{x}(n)}\, A_{n-1}^{-1}\underline{x}(n)\underline{x}^{t}(n)A_{n-1}^{-1}$$



Then for some you know algebraic covariance if I consider this a n inverse multiplying Xn then what happens if this a n inverse works on Xn that means right hand side working on Xn that means this working on Xn minus this lambda to the power minus 2 by 1 plus beta n times this thing into after this there is a Xn Xn all right.

This was already there these four terms these four terms after that Xn, but then the advantage is I can put a bracket around this and this is nothing, but again beta n X transpose n a n minus 1 inverse Xn. So, this is again beta n. So, then life becomes simple it is lambda inverse a n minus 1 inverse Xn minus lambda to the power minus 2 divided by 1 plus lambda inverse beta n into beta n scalar you can write first and then a n minus 1 inverse Xn okay.

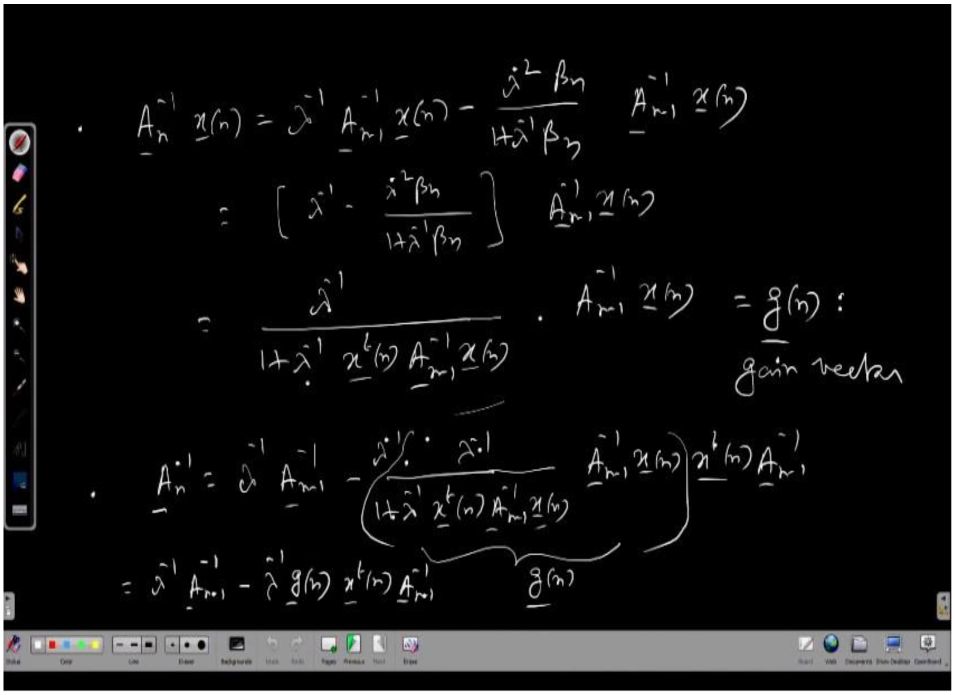


Let me write down again this means if I carry out this product I have from previous page see first I find out a n inverse formula then I am just carrying out one result if I use a n inverse a n inverse times Xn what will happen now because I know the a n inverse formula. So, I replace a n inverse by this what happens to that I will require that later that is I am working out.

So, it will be lambda inverse then a n inverse Xn minus lambda to the power minus 2 by 1 plus lambda inverse beta n times this quantity this beta n I write first this is a scalar I can write in the front or in the end does not matter. So, beta n and then this this fellow is a matrix into column vector this is common matrix into vector matrix common this is scalar this is scalar. So, I can just take lambda inverse here minus lambda inverse 2 by beta n divided by 1 plus lambda inverse beta n these are whole scalar thing multiplying this matrix times Xn and here you see 1 into lambda inverse that remains lambda inverse lambda inverse lambda to the power minus 2 beta n that cancels with this. So, what you get is lambda inverse by 1 plus lambda inverse beta n and what was beta n now you can replace beta n by this thing, this is scalar times this quantity and this entire quantity right hand side let me call it a vector Gn approximately vector matrix times a column vector that is what it is this is scalar matrix into column vector. So, vector I call it a gain vector this is one important result.

So, what I have done here is now I know Gn and I know what is this An inverse I can write this right hand side using Gn to make it look simpler. So, to do that first let me write down what was my An inverse originally obtained lambda inverse An minus 1 inverse minus this is the original form lambda to the power minus 2 like this into this. This is what we have obtained earlier now this look at this part lambda to the power minus 2 take one lambda inverse out. So, lambda inverse lambda inverse 1 plus lambda inverse x transpose An An minus 1 inverse xn same thing here and An minus 1 inverse xn An minus 1 inverse xn. So, this part if I take if I write like this into lambda inverse.

So, this entire part lambda inverse by this into this this much this much this is nothing, but my Gn look at Gn lambda inverse lambda inverse 1 plus 1 plus lambda inverse and all these things An minus 1 inverse xn this Gn. So, that means, these are the interesting result this is the lambda inverse Gn and this thing. So, not only An comes from An minus 1 now An inverse also come from An minus 1 inverse through this relation there is an extra term this is the most important thing because in that formula it is not An, An inverse comes. So, Wn has An inverse Wn minus 1 has An minus 1 inverse. So, I have to relate An inverse with An minus 1 inverse that is what I have done using these results we will now be generating Wn from Wn minus 1 that will give you a slow recursion relation that is what we are aiming for and that I will do in the next class.



Please come prepared with this in the next class. Thank you very much.