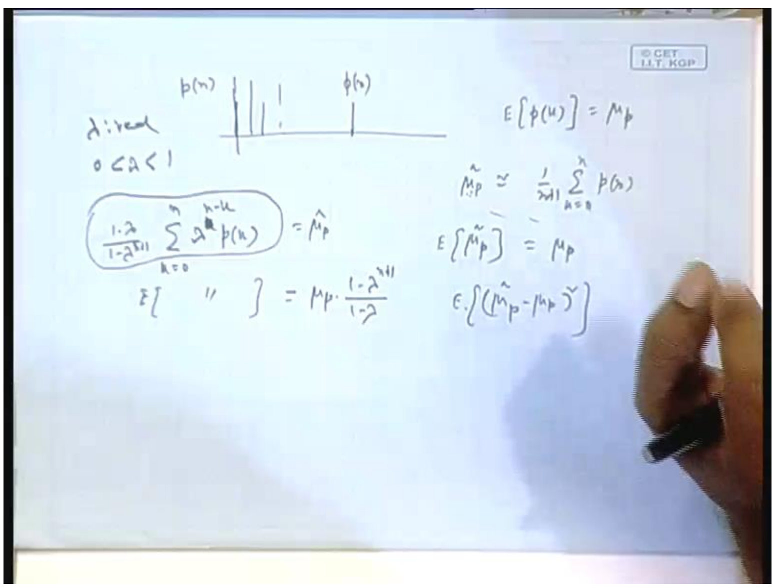
Adaptive Signal Processing
Prof. M. Chakraborty
Department of Electrical & Electronic Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture - 30
RLS Adaptive Lattice

So, this is good music**.** I discussed before these basic notions of RLS, because I think I have spent quite some time in the explaining, I mean what is our approach in RLS; I mean what is advantage, elimination, all those things we have discussed. So that I am not going into… Yesterday I introduced one factor called forgetting factor by which contribution in a inner product from the distant past can be ignored or can be suppressed, and mostly contributions from the recent samples will be taken into account. So, in that context I have to say something.

(Refer Slide Time: 01:49)



Suppose you consider some sequence; some sequence say p n: p 0, p 1, dot, dot, dot, dot up to say pn, And it is a random process, ergodic and all that. Then at each p n, p k, any p k for any k say some constant mu p; stationary. Unless stationary, it cannot be ergodic. Suppose this is given and in that case one way to estimate this mu p is, only in that case, I told you that if you assume the process to be ergodic, then n symbol average and time average turn out to be same if I assume to be this process to be ergodic, not only stationary, ergodic.

Then one way to find out mu p is to take averages; I mean to go on adding and then dividing. That is suppose mu p cap will be what? 1 by n. It will be better, I mean we normally say that it will be better if you take more samples and average, but mathematically actually what it means? That is what I want to do. We all know that if you take instead of hundred samples if you take say thousand samples and average, you will get a better estimate. But actually, mathematically what happens? That is a qualitative statement.

First, if you just take this quantity, suppose you are observing these every time and you are finding out the sum; then this is also random. Look at it mathematically. If I carry out a sum like this, without carrying (Refer Time: 03:38) you can just carry out a sum like this; this is stationary process. This quantity, this observation every time will be random, but the mean of that will be mean of this each E pn is mu p, mu p times n or may be it should be n plus 1; sorry because they are n plus 1 samples; 0 to n, n plus1; so, mu p into n plus 1 divided by n plus 1. So this is exactly mu p.

See, if you take this sum, I mean take the samples say n plus 1 number of samples, average, you get a quantity which is random. We make a quality statement, qualitative statement that if you take more and more samples, if n is very large, this is a good estimate of mu p. But that is what we have been told in some context earlier that you should take lot of samples for getting good (Refer Time: 48:37) average, but mathematically what does it imply actually? That is what I am trying comment.

Here, what will be the number of samples? If you take this quantity is random quantity, this mean is mu p; about that there is no doubt. Question is when variants of this error in this, its mean is mu p, but every time you measure it, you run this summation, you will get something.

So, the mu p cap and mu p they are not same; there is always some error. That error variants, I am telling you, you take it as exercise; you replace mu p cap by this formula and mu p we will do; square it up; you will see this entire quantity is inversely proportional to n. I am not deriving it and that is why this variance will go down; it might be inversely proportional to n or n square; that I do not do it, but I mean inversely proportional where the n or n square will too change.

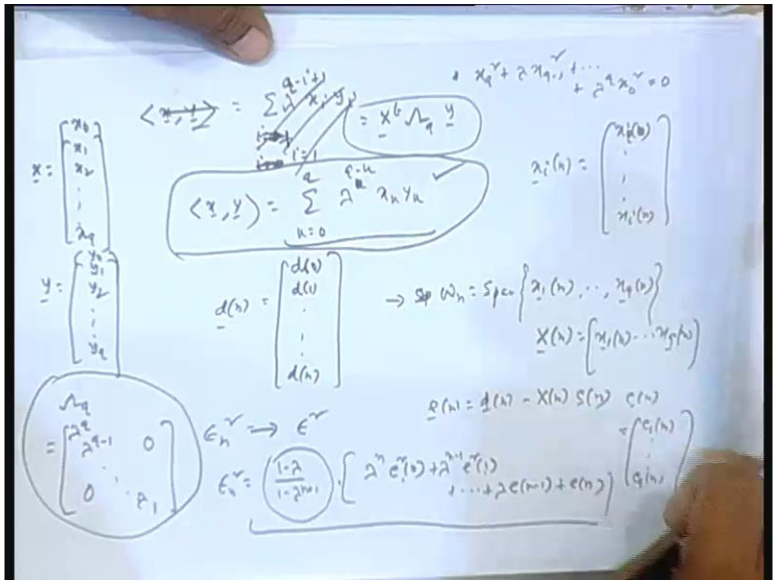
It is a very simple exercise. mu p cap to be replaced by this and take this error to square it up. You do it simple and you then apply e n and all that thing. We will see entire thing has got I think 1 by n. I mean it is inversely proportional to n; that is why if you take larger and larger sample, this error variance will come down. That is what it is meant physically. When you make that statement that you should take more and more samples, basically mean your estimation error variant will go down to 0 as index to infinity.

Now, suppose instead of doing this kind of averaging, instead of doing this kind of averaging, I am introducing a factor lambda; lambda real and I am doing an average; I mean I am doing things like this. lambda to the power k p k sorry lambda to the power n minus k p k. If you understand at k equal to 0, I have a lambda to the power n; lambda is less than 1. So, naturally that is maximally suppressed at k equal to 2, then k equal to n, and then the current index lambda to the power 0.

Suppose, I find out this quantity (Refer Time: 07:06). This quantity is also random. Every time you measure this, you have given a sequence is just running with sum; instead of this ordinary sum like this, you are finding out a weighted sum. This is also random. So, if you take the mean of this quantity, what it will be? Can you tell me? e of this will be what? e of p k is mu p; mu p into gp series; 1 lambda, lambda square, dot dot dot up to lambda to the power n. So, that means mu p times 1 minus… So, therefore if I have a factor here, 1 minus lambda by 1 minus lambda to the power n plus 1, if I evaluate this quantity, its mean will be mu p; its mean will be mu p and then you take find out this quantity, you call this mu p prime. Again, mu p prime and mu p square and all that thing mu p. Again, here you would see that it is going to be so simple as here, but again that variants will go down as an interesting thing.

So, that means we can use this kind of averaging also. Then, if you do further analysis, you will see that here, you take, I mean, to get into the some tolerance band you need more samples here than this. That is a separate matter because you are suppressing some samples which you do not or not doing there. Those further analysis can be done though it is now of no consequence here. You have to just hear me out here. But in the extent case of very large gain or n tending to infinity, you can here also find n go down to 0. This you can verify also. It will be a clumsy kind of mathematics. Of course, you have to replace v p prime by v c r then but someone can try. So, mean will always be mu p and variance will go down to 0, but it may not be as fast as here, but it will go down to zero and it is not very slow also.

(Refer Slide Time: 09:29)



Now consider that vector space treatment. I modify the inner product between two vectors where x say x q. Did I put subscript there yesterday? Yes, subscript. All are real and no conjugate. This to this is q, i equal to 1 just because instead of 0, I started 1; i equal to 1; at i equal to 1, i should have lambda to the power q; this thing say it start at x 0 mu minus k…(Refer Time: 11:08).

So, if they are taken from a time series, from the random sequence, you know this will be every time you evaluate the inner product this is the random quantity but its mean will be the actual correlation. Do you understand? The correlation between x i and any x sample or any y sample because I am assuming jointly stationery and anybody can all that. This also satisfies all the basic notions of the things of inner product linearity in x; then if c x, c will come out; that too immediately you can see and x you take will be what? This type mod y, mod x, then y mod; this everything is real; say x square is already positive.

If you have a quantity like 1, sorry x q square plus lambda x q minus 1 square plus dot dot plus lambda to the power q x 0 square, if suppose this is 0 normal square is 0, this kind of lambda is given be positive. So, absolutely every time it has to be zero and therefore lambda being not zero, each of the samples will be zero; so, that also it satisfied. We will be using this where? That earlier I was taking this thing d n, just for remembering, it was from d 0, d 1, dot dot dot, d n. This was projecting on what? Span of there is w, I think, I give the span of x 1 n, dot dot x q n; remember where any x in was x i 0 dot dot x i n.

I was projecting on this then. Projecting means I find a linear combination of this and I found matrix X n. With these as columns, I use I have to find a linear is combiner coefficient so that this matrix times that coefficient vector, I mean, subtracted from this, that is the error, non square of that error earlier with that previous inner product definition when lambda was not projected. What do we do there?

We took this vector from the linear combination of this by a set of coefficient c 1 n multiplying x 1 n c 2 n multiplying x 2 n dot dot dot c 2 n summation of that subtracted from this, that is the error; I took that error; non square means first n k of the error vector is squared up; second n k error is squared up; third n k of the error vector is squared up; like that then added; it was simple sum of square. That is what we are minimizing which is equivalent to orthogonal projections of d n on that. There is minimizing of that note, but that time I told that, if that if i at any index, that n naught as you go from 0 to n n naught, if the statistics changes, then until or unless you forget the previous contributions to the statistic changes means r inverse p optimal filter expressions changes; so optimal linear combiner thing changes. So, if you can forget terms before n naught and take lot of but after n naught then absolutely you will get the new set of optimal coefficients. But during the transition phase the problem is you are taking more data from the past, combining only a few from the new: n naught, n naught plus 1, may be n naught plus 2. So, you get set of optimal combiner coefficients which are in between, which are neither the new one nor the old one.

So, that phase is the phase of uncertainty. That phase I have to minimize. As I go a slight forward from n naught to n naught plus 1, n naught plus 1 to n naught plus 2, then n naught plus dot dot dot, very soon contribution from the past before n naught soon start disappearing and only data from time index at n naught time index equal be n naught or afterwards should be dominant.

That is why introduce the lambda factor. If you now use this inner product here, I minimize the non-square of the error with respect to this inner products, then obviously if there is an n naught index here after n naught, after a very little while you will be for forgetting the contribution here. It will be taking contribution from the data starting at n naught and afterwards. So, that business will be gone, but one thing is there.

At any point, suppose, n naught has not taken place, suppose we are now considering a situation where statistics system is changing and all. So, earlier I give you this thing, epsilon n square. I said that epsilon n square I am minimizing. Why I am minimizing epsilon n square because epsilon n square was actually approaching epsilon n square for large n. That is what I say. What are the basics of that? That is what I discussed in the beginning today.

What was the epsilon square earlier? It was sum of the squares; sum of the squares divided by say an average for that averaging factor, numbers of terms. That for large number of such terms would approach epsilon square. I told you in the beginning today. When you are doing averaging if the sum of square previously lambda was 1.You are finding the error vector e n as d n minus X n cn; cn we know is c 1 n, c 1 or what c 1 n dot dot c q n. This error vector earlier what we did with lambda 1 that is the previous inner product. What we did? We took the non-square. Non square means every element of this error vector squared, added; that will be a, that was we will we are minimizing with respect to this.

So, that sum of square you can even divide it by the number of terms. That will not change the optimal thing because number of terms and c, they are independent; they are not related. But when you take the sum of square and divide by that number of terms n plus 1, then according to previous discussion today, that will be a good estimate of these provided n is large. Mean of that will always be actual epsilon square, but variance will go down.

So, whatever you get, that will approach this; that we discussed in one way. Question is what happens with respect to the new inner product? Is the new inner product, it is not simply sum of squares of the error; weighted sum of squares. It will be like lambda to the power n e square 0, lambda to the power n minus 1 e square 1, dot dot dot dot lambda en minus 1 plus e n. This we want to minimize with respect to this coefficients because e n vector from that e 0, e 1 up to e n comes and those terms depend on the optimal coefficients which is the coordinating function of the filter coefficients; this you minimize.

In fact, you can also introduce that factor 1 minus lambda by 1 minus lambda to the power of n plus 1 here time this because whether you minimize the entire things or minimize this, you will give the same coefficients c n, because this is this is this has no c n here. You will be differentiating these with respect to each filter coefficient, combiner coefficient. So, whether you have this factor here or not, it does not matter.

Suppose I bring this factor. In that case, this one like earlier I did not bring the factor. Lambda was 1 and I simply put a n plus 1 factor below and I said this is a good average. If n is large according to previous discussion that would approach epsilon square; fair and enough, fair enough.

I have made this slightly different. So, I have to recheck things here. Now, sum of square has become a weighted sum of squares like this and instead of n plus 1 in the denominator, I bring this quantity (Refer Time: 19:33). Still if I minimize this with respect to the filter coefficients, I will get the same ones because this factor does not change my optimization. dependants on the filter coefficients is in this: e 0, e 1, en; not here, but as I go for minimizing, I am minimizing the entire quantity which is from the coefficients. And what is this quantity? This we saw in the beginning today that mean of this quantity is again this is epsilon square.

Mean of this quantity is again epsilon square and this also for large n, I told you, variance of this quantity also will go to 0. Variance of these minus, if you call it epsilon n square, epsilon n square minus actual epsilon square that variance of difference between the variance of that, that will also after 0, according to the discussion in the morning today,

So that means jolly well I can as well optimize these and the coefficients I get, they will also approach the optimal filter because this is also approaching epsilon square. Epsilon square when minimized with respect to the combiner coefficient gives you that r inverse p; that is the optimal ones.

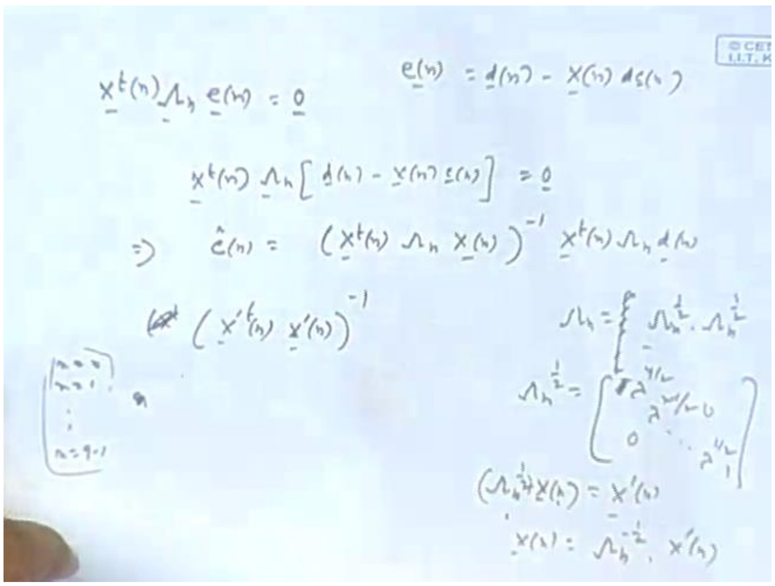
So, earlier I took simple epsilon square, no weighted sum. Now, I am taking this, but even here also I should get back the same optimal filter coefficients because after all these also will tend to epsilon square for large n. So, instead of that simple this thing, you know, sum of squares, I will be taking this weighted sum of squares. That is why I will be using this inner product. Fine because I will get in any case the same optimal filter coefficients that is what I wanted to say so far.

So, that means this quantity this quantity you can see you can write as X transpose a matrix delta q y, where delta q is a diagonal matrix. This, here you focus on this, this and this (Refer Slide Time: 21:59), these three. x k y k and also x transpose; this is the x vector, y vector. Can you see it easily? This is it.

So, this is what and this matrix depends on q because of this factor q. If you have that x row vector and y column vector, multiply you will get this. You can easily see. x 0 into y 0 into lambda q, x 1 into y 1 into lambda 2 to the power of q minus 1, dot, dot, dot and all added, fine.

So, this is how I will be writing. So, that means now what will be the optimal coefficients as I minimize with respect to minimize this sum of weighted sum of squares with respect to the new inner product? How to find that? I will follow the same procedure that is I will take the error. I am doing the orthogonal with respect to this inner product only. I am carrying out orthogonal projection; that means projection error is orthogonal to this space w n. So, I will find the error; inner product I will compute of that I mean with that error I mean between the error and instead of these function and instead of these vectors and that will be 0.

(Refer Slide Time: 23:27)



I will approach, use the same thing that is e n. e n was I am rewriting. This is much of, with much of whatever I discussed so far is a, repetition actually.

So, e n I have to take the inner product of inner product between whom? e n vector and each column here. What is the mean here? How to carry out the inner product? As per the new definition, I have one delta n; it is called what? I will sorry lambda n. Sorry. Thank you very much. lambda n, I mean, delta is you know big delta is really delta this is bridge here; lambda n and any vector from here; that will put a row form. X transpose this y. Earlier this was not present; it was simply row column.

So, that means first column here is X 1 n. So, that will be X 1 n transpose. That will be 0. Next column is X 2 n. X 2 n transpose with this 0 because of orthogonality inner product between e n and each of the columns is 0. So, as well I can put like this. After X n matrix first column X 1 n; after transposition that becomes first row. x 1n column vector becomes X 1 transpose n. First column gives rise to first row; second column gives rise to second row after transposition. So, first this will become here 0. 0 vector of length q because there are q terms here; q column vectors. Say absolutely same treatment but en is this; en is this; so you replace e n by this expression (Refer Time: 25:20). This gives rise to what? C cap put as what? More generalized form than earlier.

And once again invertibilty of this, do you understand? I am what I am doing is earlier I did not have this lambda n. I got up to this. I gave all explanation, everything. Now, today, with this in the new context also I have to carry out the same treatment. What is the guarantee that this matrix will be invertible? For that, you see one thing. Earlier I said X transpose into X is positive not negative in definite. That will be valid here also. How? See the trick: delta this lambda n, I can write as lambda n to the power half, decimal matrix.

I will define what this matrix is. Again, lambda n to the power is called squared root matrix. Half is nothing but since all lambdas are real, square lambda to the power earlier it was n; now it is n by 2, lambda to the power n minus 1 by 2, dot dot dot lambda 2 the power 1 by 2 then 1.

If you define this matrix, if you multiply these two, you will get original lambda. Agreed? And then, lambda n half into X n; you call it X prime n. So, obviously this is X prime n sorry and this is also similar to lambda. These are symmetric matrices; diagonal matrix symmetric matrices.

So, you can write like this x prime n x prime transpose n.

First X prime n X prime n means this part here and this is transpose; X transpose and transpose of this but transpose itself that comes and these two multiplied you get lambda n. This is a positive definite. A non-negative definite matrix also called positive definite. So, which has always Eigen values real and non-negative, but for it to be invertible Eigen values should be strictly non zero. That means it has to be positive definite. This is why you did it last time also.

So, what is the requirement for that? X prime n should be such a matrix that no linear when the column should be linearly independent. That is possible only if you are taking n that is number of row is at least equal to number of columns or more; otherwise it is not possible. This is a repetition of all that we do. So, number of that is valid here also; that is valid here also.

That is number of rows, do you understand? You remember [FL]? This X prime n, what are the rows? n equals 1 to 1 dau n equal to 1 1 dau dot dot dot n equal to, I mean, I have got how many columns? q columns.

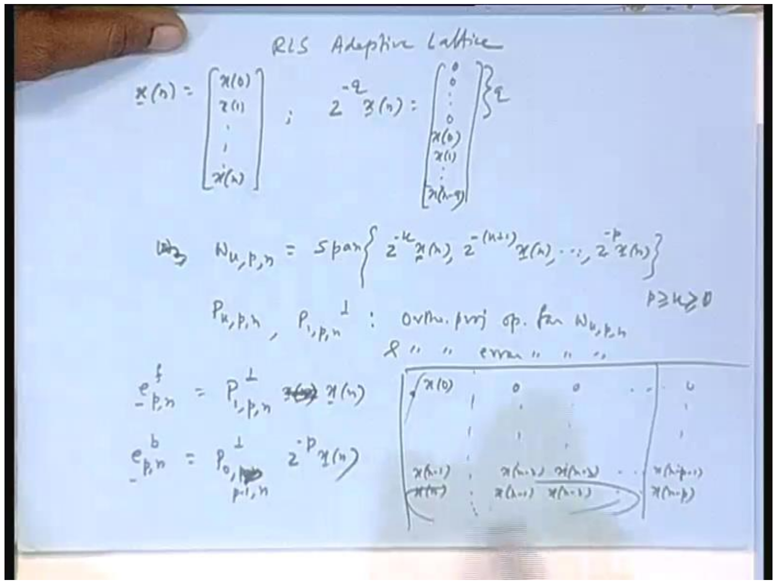
So, n equal to q minus 1. These rows, how many rows? q. So, n should be at least q minus 1 or higher. If n is less than q minus 1, obviously I have got more columns than rows. So, some column will be but that means the dimension is n; dimension of the vector space is n; number of vectors, number of columns is more than n. So, at least one column should be expressible as linear coefficients of the others.

If I have got n number of linearly independent vectors, there are more than n because q is greater than n. So, q minus n extra vectors; each of them should be expressible as linear coefficient of the rest. So, immediately then you remember that positive definition goes. I can find out the linear some vector here a, which will give rise to this into equals to 0, but a is non-zero that will corresponds to Eigen vector of that Eigen value 0; all this I did last time. So, again this is the condition that first n should be at least q minus 1 or greater and there should be no linear relation between the columns of X prime n otherwise. But this X prime n, I have to talk in terms of original X n, X prime n and original X n. I am saying that the coefficients of the columns of X prime n should be linearly independent. There should be no linear relation better than it.

In terms of X n, what does it mean? Columns of X prime n and columns of X X n, how are they related? Do you change the rows; that will not change your linear dependent thing. If suppose columns of X prime n are linearly independent, please do it yourself and then show that columns of X n also are linearly independent. You can do it very easily you know this way.

Suppose this is lambda into the power half into X n equal to this. This is the square matrix. So, that means X n is into X prime n and then from X prime n you can infer about it. You do it as an exercise; take it as an exercise. Let me do it this way. So, then in conclusion, we can do all those things; we just removed the new inner product. Now I march to this topic of today that is RLS Adaptive Lattice.

(Refer Slide Time: 32:08)



RLS adaptive lattice: I am assuming that my data sequence X n which is a random sequence is starting at n equal to 0.

The data before n equal to zero will assume to be zero. This called pre windowed case, pre windowed, this is a space particular case of this lattice filter. There is other extensions where you are not assuming data to be 0 before n equal to 0, but suppose I am assuming here because that is always the case, because switch on somewhere at some point of time with 0 initial conditions.

So, you define the vector X n as x 0, x 1, dot dot dot x n. So far so good. Then define this; I am giving this Z to the power say minus q. For any integer q x n, Z was delayed; it will be delayed like the sequence, original sequence is delayed q times; that means first q locations will be 0s. Pre windowed case any data before n equal to 0 to the left of origin is 0. So, those 0s are coming in. So, how many zeros? q 0s. q 0s and then x 0, x 1, dot dot dot… (Refer Time: 33:42).

W say k p n span of, you will understand why I did that lattice, stochastic lattice in that projection frame work using vector space, the treatment will be over lapping. You know in this frame work will be same in some places, most of this I would say.

I am assuming of course p greater than equal to k; k greater than 1' greater than equal to 1. So, at least one term is there; z to the power minus 1 or maybe here you can 0 minus in this definition. This is W k p n; this is a span. P k p n, remember in the stochastic lattice case I did not introduce this; I did not have this n in the subscript because I told you using stationarity all whatever I do n will disappear. So, you can jolly well and whatever you do get real out of index n. If you replace n by n you will get the same result because of stationarity. I was not using n here. Here it is not;

everywhere I have to bring this, keep this n here; there is no e operator, no stationarity business, nothing. Please understand there will be no e operator anywhere. It will be exact computational linear algebra.

You know very well, ortho projection op for an orthogonal projection error operator for this.

Then define e p. Earlier I had this this vector e p f n. First define and then will see what is meaning actually. e p f n as sorry x p 1 p x; that is p 1 p.

Firstly, see for first time using what is… first forget about this projection error. Take this as a projection. What does this projection mean? On one hand they have got x n factor and what is the space here? z to the span of z to the power minus 1 x n to z to the power minus p x n.

So, last take the last element of the each column vector. Here I have got x n and here I have got x. Here how much? z to the power minus 1 x n means 1 0. So, last element will be x n minus 1,

next one will be x n minus 2, next one will be x n will be x minus p. Before that will be x n minus 1 to x n minus 2, xn minus 3, dot dot dot x n minus p minus 1, dot dot dot dot and here it will be x 0; here 0 because x minus 1 is 0, x minus 2 is 0, 0, 0 0. This is the situation.

Is this letter size coming too small? Can you see this because I get a feedback that size is possibly small? Is it okay this size letter size?

So, finally I do projections means projections of these vectors on the space span by these columns. This is one column, this is one column, this is one column, this is one column space span by this column vector. That means what? I will be linearly combining them by a set of combiner coefficients and take the sum of squares of the errors with that weight lambda. lambda you forget for the time being. For the time being, assume no lambda. Lambda equal 1. Then what is happening? You are finding out the combiner coefficients so that x n minus a linear combination of past p samples, then x n minus 1 a linear combination of past p sample, dot dot dot, we are finding out the errors, square up and summing.

If you minimize these because that if you take the error, square up and take the sum that is the non-square. If you minimize this, then only you are computing orthogonal projections, but if you minimize this, there should the combiner coefficients. Then for large n, those combiner coefficients indeed will be the optimal forward predictor coefficients and these projections will be indeed for optimal the actual forward prediction. But with those coefficients we will get a vector; with those coefficients when I combine the columns you will get a vector; vector of what? Orthogonal projections.

If n is larger and larger, if from n you go to n plus 1 or n plus 2 you will get a new sets of coefficients. By then you combine these columns, you will still get vector may be of length n plus 1 or n plus 2, but this initial part will be common but initial part common means this projections will be better projections then. This I discussed. This I discussed at length for half an hour in the very last class. I am just trying to show what does it mean here. And this will give indeed give rise to, I mean p 1 p n x n will indeed give rise to, I mean if you take this vector, projection vector, that last coefficient, latest coefficient, current coefficient, current term, latest term, current term. That will indeed turn out to be the actual prediction error; p th order forward prediction error when you have taken a lot of data. Because I am what I am doing? I am instead of I no longer have that correlation things or e operator having this exact value of the prediction error variance which we are minimizing, that time prediction error variance was the norm square of the error there; prediction error there.

What I am doing? I am finding out linear combiner, subtract from x n, get one sample error, another sample error, another. I am squaring up the error, adding, minimizing it; that is computing orthogonal projection. But minimizing that error means what? For large n that error will be approaching the actual epsilon square that is an actual prediction error variance.

Student: (Refer Time: 41:20)

How?

No, we just see system is such that it is a indeed we have got 0 before x 0; then am not introducing any error this that this is not erroneous. Your question is

Student: (Refer Time: 41:41)

I am not projecting a particular sample on those 0s. I have got a random sequence. I am taking any sample at nth index. I am finding out the linear combination of past pre values taking the error square up the error; again do the same thing at n minus 1 nth index, n minus 2 nth index, like that adding up the errors so that I get a good estimate of the actual error variance. Stochastic variance, I mean stochastically obtained a variance that I am minimizing.

So, if I take large number of samples the sum of square thing will indeed tend to be a good estimate of, will converge on the actual variance; e operated on the prediction error variance square. Therefore, the combiner coefficients will also give you the optimal ones. Therefore, the projection will be if you take the end with those optimal ones, you combine these vectors, you will get a vector, but I am only interested in the current component, latest component because that is the index where I am standing that is the current time index.

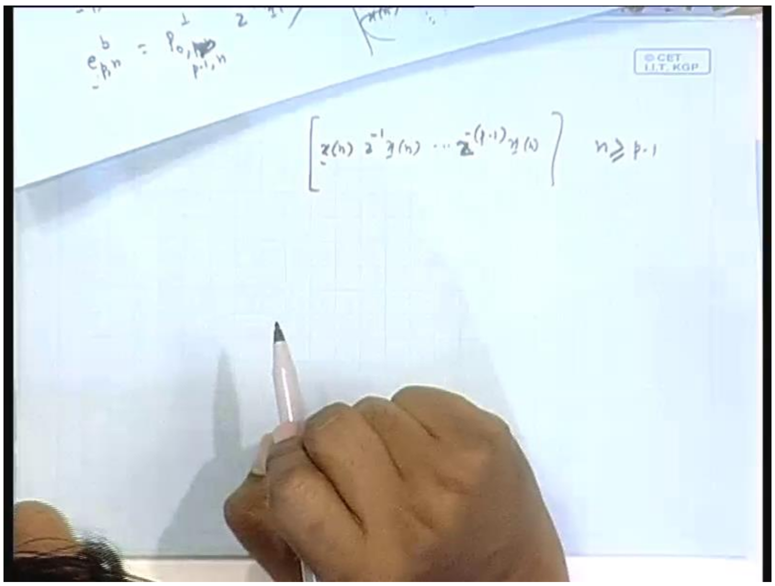
You will get previous components also which will be a better estimate of a projection because you are using we use more data than the earlier. Therefore, you get a better combiner coefficients but I am not bothered about x; I mean the index 0 or index 1 or index 2 anymore. I am already bothered about the current index. If you take out that component that is if we take, this is a vector, if we take the nth component of this vector; that is what I will be focusing on. And that if total number of samples taken is very large, then that component will indeed be I mean the actual forward prediction error obtained in the lattice earlier. That is what I mean. Understood or not? Is this approach clear?

So, this is my definition. By the same token, I now define sorry this is p minus 1 because this is z to the power minus p x n mean this guy (Refer Time: 43:59). This guy will be projected on what the space span by the previous fellows along with x n. This will be x n, z to the power minus x n, z to the power minus 2 x n up to the one before this space span by that. On that I will project this that will give rise to backward projection. After all what is the last entry here? x n minus p; that are not that you are not linearly combining as in terms of this fellows. p future terms standing at n minus p; taking that error again, here again, here squaring of all those errors sum, minimizing that is only you are computing this projection. But sum of that square will be what? A good estimate of the actual backward prediction error; p th order backward prediction error.

If total number of sample is very large, see you minimize that as you compute the orthogonal projection, those coefficient combiner coefficients indeed will converge on this optimal backward prediction error coefficients. Same logic what I had for forward prediction, same for backward prediction and therefore with those optimal, with those linear combiner coefficients if I combine them I get a vector. I am interested only the latest component; that component will be what? Some combiner combination of this fellows which will be a very good estimate of the actual p th order backward prediction error at index n.

This is my definition and now I have to form this. I have to find out and one thing I should tell you. You know this let us consider this. From this, I have to find out e p plus f n e p plus 1 b n. Of course, it will be very similar to what we did there in that stochastic lattice, but just before that, there are you know places where things are different and we should be very cautious. Look at this space here say in the forward prediction error p 1 to p n, for here also p 0 to p minus 1 just take one of the two is here.

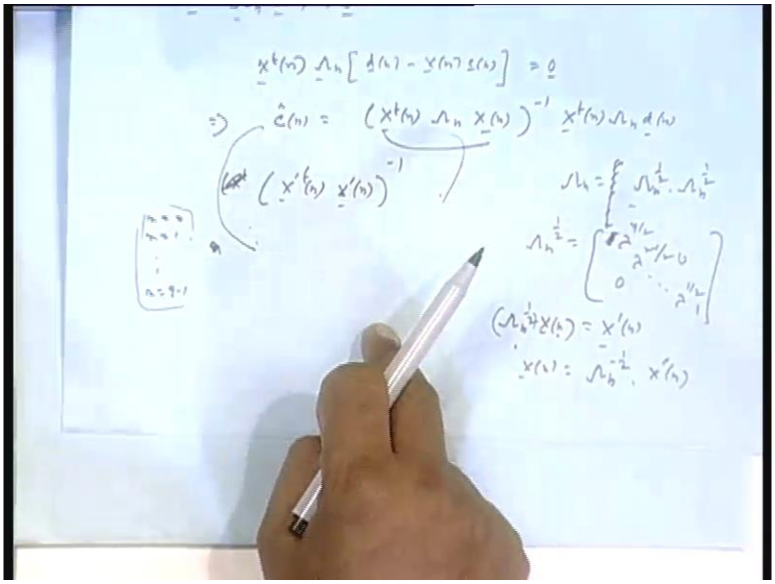
(Refer Slide Time: 46:08)



Here, what I have got? How many terms? How many terms? p terms. How many columns here? p columns. We have to assume, mind you, this column we start at x 0 and go downward. This we start at 0 and then x 0 data; I mean 0 and then data 0 0, data 0 0 0, data. All the analysis for the time being you assume that we are doing for index n greater than equal to the total number of rows. How many rows or total number of columns? How many columns are there? P; n starting at 0; so, n is greater than equal to p minus 1.

That is true here also. That is true here also because you can see easily number of column in either case is p. Assume that we are doing all these analysis for n higher than the number of columns present in the number of columns present in the respective data matrixes; otherwise, that problem of that thing will come up; dependence or independence will come up. This we assume and how that time it was showing up when you computed that exact that matrix.

(Refer Slide Time: 47:47)

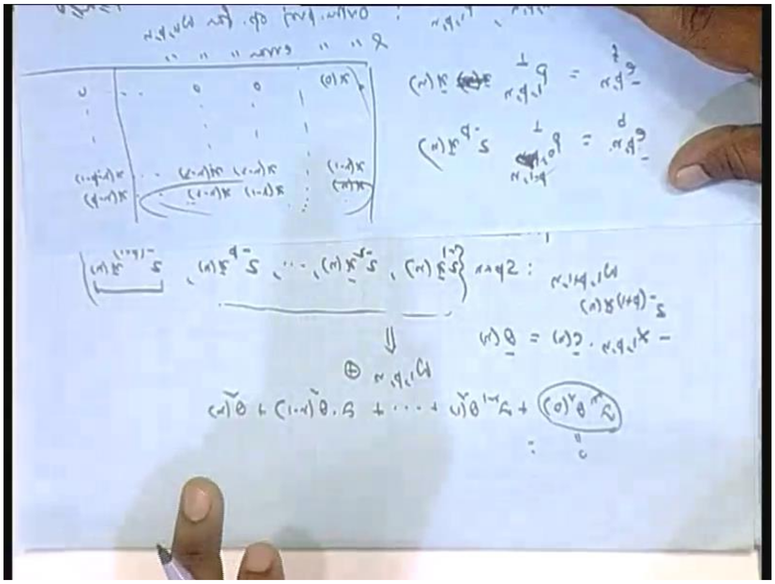


When you computed the c cap n directly by this, but will we will not be doing like this. We will be because that is a direct computation, we will be obtaining them recursively. When you are obtaining directly at that time, it was showing up clearly that you know this should be invertibly and all that, but in the recursive procedure no such matrix base computation will be done; no such inversion will come up directly. But this phenomenal will reveal in a in a different way, in a different way that that columns of X n must be linearly independent. Otherwise, you need solution. This is not I mean this solution does not exist.

In the recursive phenomenal also algorithm which will, this will come up in a different way. In a direct computation, it comes this way that is inverse does not take this, but will not be using direct computation; will be recursive and will develop a recursive algorithm.

But that phenomena we will proceed is crucial. Here also that will see how the linear independents of column will be required; that is how number of rows should be greater than equal to number of columns. That will reveal here indirectly; not the way it is comes here that inverse of matrix comes up and then inverse of that means and all that.

(Refer Slide Time: 49: 02)



So now, W k p n you know. So, suppose I wanted to W k p plus 1 n; very quickly that will be W k. This treatment is absolute similar to what we had there. W k p plus 1 n, not W k, w 1 p plus 1 n. Here I have W 1 p n W 1 p plus 1 n will be W 1 p n and direct sum is what? This guy;

This needs to be explained.

We had lets go slowly because things are I mean even though there is similarity between lattice and this every at every step you know there is some new elements. So, let us be let us not run through it. What we are doing? We are giving e p f n and e p b n. We want to find out the same quantity, same vectors for p plus 1 th order p plus 1 th order and the same index n and then I will take the last component. Taking the last component it is not a problem and same thing I have to do.

Now, so that means the space will be for finding out e p f n; e p plus 1 f n. What will be the space? Earlier it was the w 1 p; not in the w 1 p plus 1. I have there is no doubt. What is this space? The span of there is a nice thing here. Sorry, Z to the power minus 2, Z to the power minus 1. It was 1; first p, then I will write that separately. This part I will take out separately; there is a span of that will give rise to W 1 p n. About that there is no doubt; 1 p n, about that there is no doubt.

Question is - this will be direct sum of there will be direct sum of I mean this entire thing will be this space direct sum with the span of what? If you take this guy, project it orthogonally on this space, take the error, span of that there is the slight difference between e p b n and that thing.

There is slight difference. It is not coming directly; this not so nice as in the case of stochastic lattice where stationarity was used and n was dropped. Here biggest problem will come because of presence of n. That time you know n was the index n was missing and lot of I could play around you know but here n is present everywhere. This guy, please see the inner dynamics. I donot know whether I have so much time or not; this guy is still projected on this space; first fellow top entry of each of the columns is 0. Start at z to the power minus 1. When I do the orthogonal projection, what I will be doing? I will be linearly combining them and then last entry I will subtract from a last entry here; last but one entry of that combined thing subtracted from here.

I will take that errors, square of the errors; top most entry will be what? Combine value of 0s minus subtracted from 0. It will be 0 and sum of square of the error I will be weighting them; how? That is if you consider this part to be this matrix, you take X; you put the columns in a matrix form 1 p n 1 because Z to the power of minus 1 x n goes up to p and n I am putting them as a column of a matrix. This matrix is to be multiplied by a set of combiner coefficients. How many? 1 to p. So, this is a length p vector; that is the projection; this minus I am projecting this orthogonally on this. So, this is what I have to do minus this; this is that error (Refer Slide Time: 53:38) I donot know what error is that. I have not given any name to it. See, you say call it a theta n. No problem but what is theta 0? I will be taking this theta n and I will be and what is theta 0 by the way? 0 because top guy first row of this matrix is all 0, because Z inverse 1, Z inverse 2, Z inverse p; first row here first element here also 0.

So, theta 0 and now I will be what? I will be doing minimizing lambda to the power n theta 0 square plus lambda to the power theta 1 square dot dot dot dot lambda theta n minus 1 plus theta n sorry theta square nth. This quantity I will be minimizing. But this guy is 0; forget this; this equal to 0. And what is this quantity? Suppose I am not standing at index n minus 1; suppose I am standing at index n minus 1 here and I want to find out the backward prediction error vector p th order, but at index n minus 1; that is current index is not n but n minus 1; then this vector will be what? Z to the power minus p x n minus 1 what? Will it go? I think you know there is no time today. I mean this thing will come with this thing. So, I will continue from here in the next class.

You please come prepared with just this back ground because this will take at least ten minutes. You know to explain so when we meet next, fine.

Thank you.