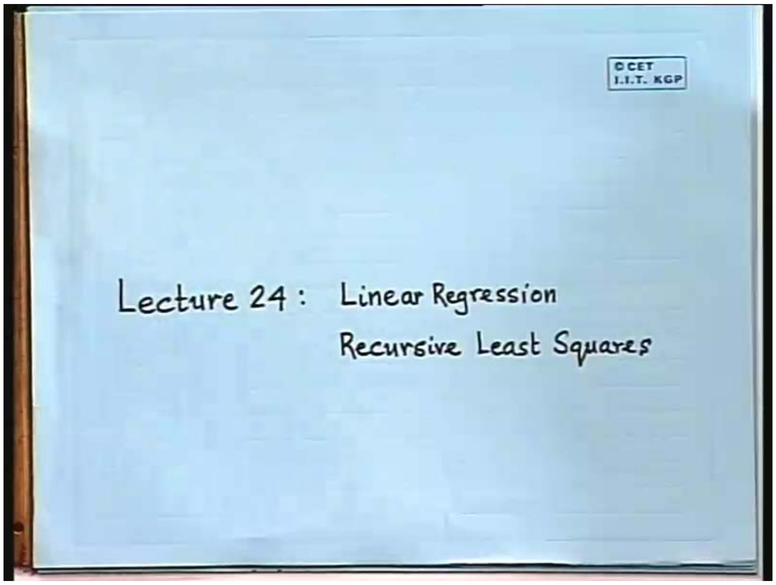
Estimation of Signals and Systems

Prof. Mukhopadhyay
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

Lecture - 24
Linear Regression Recursive Least Squares

(Refer Slide Time: 00:53)

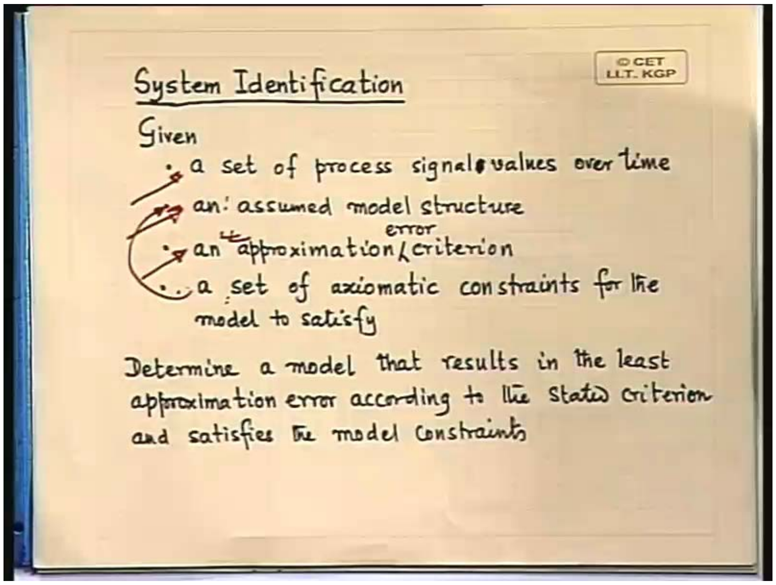


© CET
I.I.T. KGP

Lecture 24 : Linear Regression
Recursive Least Squares

We will discuss, what is known as a linear regression and a the the the the simplest method for solving that problem. So just to recapitulate once, we loosely defined system identification as a that is if you are given a set of process signal values; we are talking about process signals because we are generally concerned with identification of I mean, calculation of models for dynamic systems.

(Refer Slide Time: 01:16)



© CET I.I.T. KGP

System Identification

Given

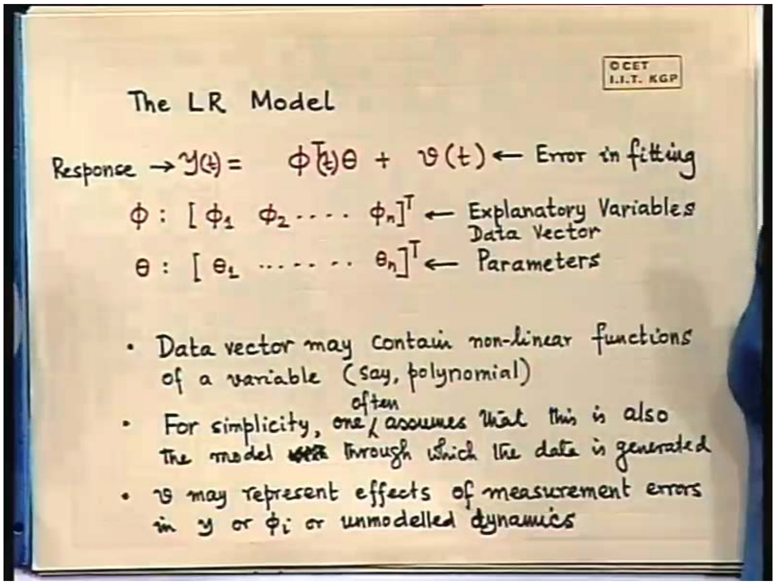
- a set of process signal values over time
- an assumed model structure
- an approximation error criterion
- a set of axiomatic constraints for the model to satisfy

Determine a model that results in the least approximation error according to the stated criterion and satisfies the model constraints

So we there will have the based on their input and output signals. So given a set of process signal values over time, some model structure we have to assume and some obviously some error criterion we have to assume. And also we could have you know about the system we will have certain a priori knowledge's, we have some experiences with the systems. So we know something already, even if you do not know the detail model set we know certain things.

So we should obviously trying try to use whatever knowledge we have while trying to look for model. So there we may have a set of axiomatic constraints for the model to satisfy, wether the system is stable or or there is D C gain is positive, some such facts we may have. So given these things, we have to determine the model that results in the least approximation error; according to this criterion that is the problem, okay. So before talking about you know we are we are we are concerned with dynamic models, systems, typically linear dynamic models; you can say systems which are describable by transfer functions loosely speaking, okay. So but before that, actually this problem is much more general, and that is why we want to look at it as a linear regression.

(Refer Slide Time: 02:51)



©CET
I.I.T. KGP

The LR Model

Responce $\rightarrow y(t) = \phi^T(t)\theta + \vartheta(t) \leftarrow$ Error in fitting

$\phi : [\phi_1 \;\; \phi_2 \cdots\cdots \phi_n]^T \leftarrow$ Explanatory Variables
Data Vector

$\theta : [\theta_1 \cdots\cdots\cdots \theta_n]^T \leftarrow$ Parameters

- Data vector may contain non-linear functions of a variable (say, polynomial)
- For simplicity, one often assumes that this is also the model through which the data is generated
- ϑ may represent effects of measurement errors in y or ϕ_i or unmodelled dynamics

You know remember that, one of the first system, one of the first systematic identification exercises which we have done probably all of us have done in our school is; when we try to do some experiment in the physics lab or in the chemistry lab, and we got some points and we try to fit a line through these points, you know that was our systematic identification exercise to be précised. Because you are given a set of data points which you got from experiments and you you have an you you know model structure which is given by some physical law; may be ohms law and you are you want to fit a best feet line, best feet in what sense? Best feet, in the least square sense, so you have a model structure, you have a you have a best feet criterion.

And you have data and sometimes you may you may also like to incorporate a priori knowledge, in the sense that you will like to ensure that the line passes through the origin let us say. So so from such things you also like to incorporate; so that was our one of the first system identification exercises in a in a general sense which which all of us have done. So in that we actually what in that that is not a case, where we are having a time sequence of signals of from from process model, but the but the method is absolutely similar. So first of all today we would like to see this, see how to do this model fitting, not necessarily for a for a for a for a sequence of signals y k and u k, but from but for arbitrary measurements; I mean we have some observations and we have a response and we would like to fit a model, those responses could be so many things, I mean the the the index may not be always the time index, right.

So latter on we will we will we will specialize this method for our case, where we we want to identify a transfer function, we will we will show that many of the transfer function identification problems actually finally boil down to a linear regression, right. So first let us let us try to see some results on linear linear regression which is much more general, which can be used to relate so many things I mean economic economic phenomenon, whatever the physical experiments, using a linear model, right. So in in linear regression we typically use a model like this, where we have some some responses we have and we we want to know, for example and we have we have some responses and we have some what are known as sometimes known as explanatory variables.

So we want to know, how this response depends on these explanatory variables; in this case I have written them using a using a vector notation, but actually it it will be like you know phi 1, theta 1 plus phi 2, theta two plus phi three, theta three, so this phi 1, phi 1 phi 1, phi 2, phi 3 are generally called explanatory variables things which explain how this Y has come. And this theta 1 theta up to theta n, are the parameters. So we would like to find out suitable values of theta such that these variables can can explain this Y, that is the that is the problem, these variables could be anything, okay. And obviously we will not be able to, given given a set of data there is no guarantee especially because we have we have assumed a linear structure; there is no guarantee that we will able to fit it exactly, right. Never we when we when we fitted a best feet line, always our points were on both sides.

So so there is going to be some error. So that error I am simply saying that, if I fit using a parameter theta, correspondingly I get an error. Now remember here I like to mention one very fundamental fact, which you should remember whenever you are dealing with an identification problem; that this model is simply a postulate that is we are looking for an explanation between variables phi 1, phi 2, phi n and Y, using this model structure, for for for whatever it is. These models may have been generated, how these with how these data has been generated that is a totally separate thing. For example, you can this data might have been generated actually when when when when it was finally came, it might have come due to due to many other reasons; some some of the explanatory variables may not even be included here, that is possible or their their loss may not be so simple linear case.

So we are not when we are when when we are trying to estimate using this process, there are there are two things; one thing is the data generation mechanism, for example suppose I have input and output data generated from a fourth order system suppose, so I stimulated the data using a fourth order one. Now I can say that no no no I want to fit the first order model system, let us see what happens. So I can it is not necessary that, that if I if I if I if I am estimating a first order model, data data has come from first order model, data that might have come from non-linear process.

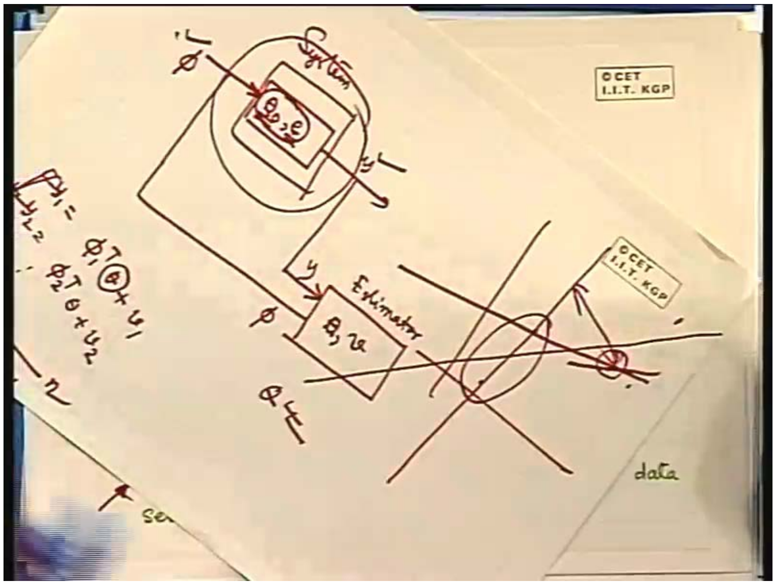
So so distinctly there is a data generation mechanism, that is that is that is underlined system which is given rise to these data; that is one entity and and the model that we want to estimate is another entity and these two a distinctly different entities, they did not necessarily be the same. So here we are we are we are not saying anything about how the data is generated, we are simply saying that we have some Y's, we have some phi's and we would like to fit a model to it. Whatever, however, I mean as accurately as possible. Whatever we will not given to model, we will be calculating it, that is all that we say, right.

So so and in fact there may be there are several and the second thing that I want to mention is that; it is not necessary that, these functions when I say a linear regression I mean that, it is linear in parameters. Here you may have for example; a very typical case will be that y is equal to say a 1 plus a 2 x square plus a 3 x cube, this is the model which is non-linear in x, but it is linear in v parameters a one, a two, a three. So what we mean is that, we we we mean linear in parameters that is why it is call linear, it is not linear it may not be linear in the signal elements, when we say linear systems it is linear in the signal elements; in this case we are talking about linear in parameters, okay, signal elements may very well be non-linear, does not matter so this must be remembered.

Now sometimes you know for I mean, especially if we if we want to get analytical results; we will have to make assumptions about how the data was generated, without that if if if the data generation mechanism is left unspecified, you cannot get any results about the quality of estimates, especially. So you will have to make assumption very often, you have to make assumptions about how what what model generate the data.

So for that at least one thing is clear, that if if your method is good then if the data is generated from first order model; and if you are trying to estimate a first order model then you should get a good estimate, at least that much your model should be able to do. So to to be able to verify that, very often we will make assumptions, that the about this system data generating model and very often we we assume that the data generating model is also this, this or this form. Only thing is that in that case we assume that there is there exists, some parameter theta naught which has given rise to this data. And you see, what let me explain this because this is this will come again and again; that there is some phi's I am going in, okay, so y is coming out, this is your system.

(Refer Slide Time: 11:46)



So so how y is coming out, so there is a system parameter theta naught and there are some disturbances e, this disturbance are parameters are properties of a system. Now this phi and this y you are feeding to your estimator and you are and so now this is your estimator, this is your system. So system use some theta naught and some e to to generate this y from this phi, these are unknown to us. Our objective is to find is to is to find some value theta, using our estimator; so we will feed y and phi and for this value of theta will get some value of v, so this theta naught and this theta are going to be distinct, this e and this v are distinct, ideally speaking. And now at least we will we will once we do that, then we will naturally like to see whether theta naught, whether theta will be close to theta naught.

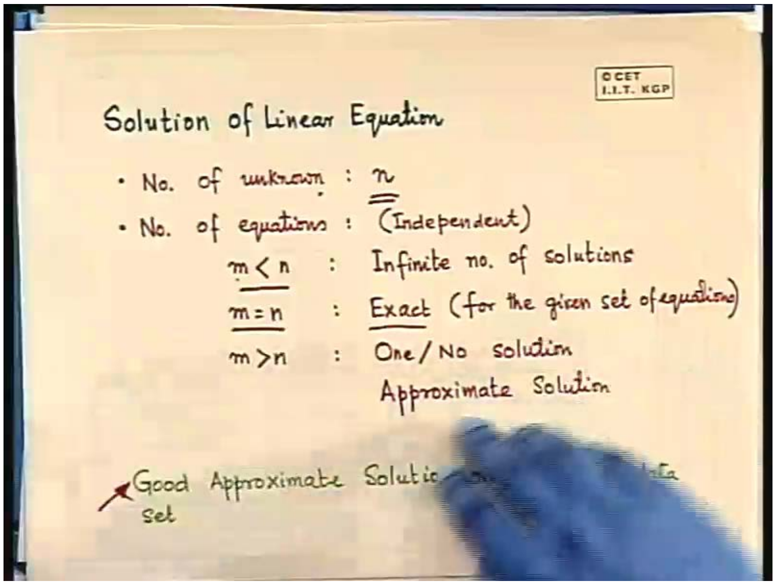
So so such properties will look for, so very often we will assume such model structures for the system also; but we must remember that the that the exactly the disturbance that is that the system used for generating the data and the disturbance, that you got from your model miss match are different things, right. So these things should be clear, right in the beginning because sometimes you know they they they get they get mixed up; because they are same forms of models, you do not know which one you using, are you using the true system model which is unknown or you are using the estimated model you must be able to distinguish between them.

When we may represent effects of measurement errors in y or phi and it it will affect; I mean I mean various kinds of thing like un-modelled dynamics or your or your estimation inaccuracy, so many things will come. So having said this, let us try to find out first of all you know, what you would what you would ideally like to do; this v is simply an error in fitting, that is something which we could not fit but but ideally the best would have been if I could have a theta which would exactly satisfy y is equal to phi term plus theta, that would give me zero zero error fitting and the question is can I now do that? I can certainly do that in this equation, but but in general the point is that I am not go to fit it for one point; I have to I have ten, twelve, fifteen, twenty points and at the feet all of them, right that is the problem.

So actually I have a set of such equations. So actually my set of equations are y 1 is equal to some phi 1 transpose theta plus v 1; y 2 is equal to phi 1, phi 2 transpose theta plus v 2, in in this way I have say n number of data points plus v n and I want to find this single parameter vector theta, such that all these equations are satisfied in the best possible manner, that is my problem.

Because I have a I have a number of experiments, when you did in a physics lab you had a number of points and you want to fit one line which will it will approximately describes all the points, right. So now when can you do that? Firstly can you solve it exactly? It is it is it is a very simple thing. It is a it is our linear simultaneous equation solving problem, that you have a you have to solve a set of linear equations, for a for for those unknowns, so so when can you solve it?

(Refer Slide Time: 15:36)



©CET
I.I.T. KGP

Solution of Linear Equation

- No. of unknown : $\underline{\underline{n}}$
- No. of equations : (Independent)

$\underline{m < n}$: Infinite no. of solutions

$\underline{m = n}$: $\underline{\text{Exact}}$ (for the given set of equations)

$m > n$: One / No solution

Approximate Solution

Good Approximate Solutio [illegible] ta set

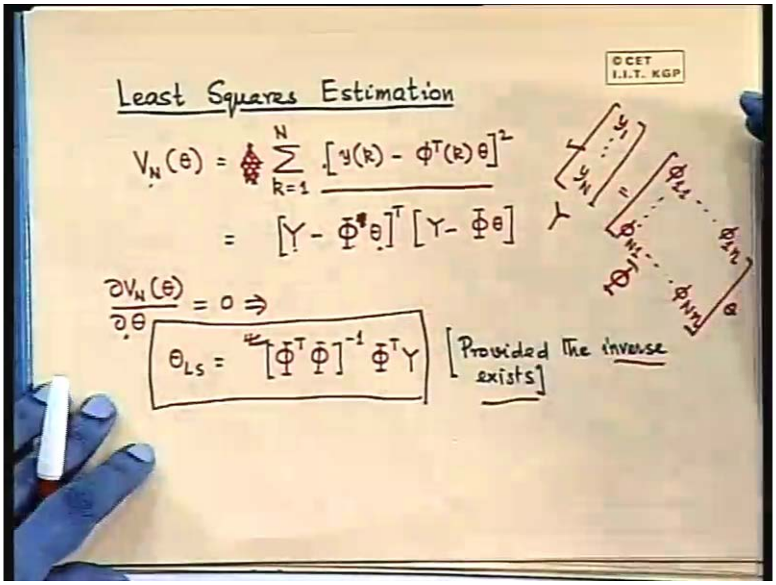
If you if you if you if your number of unknowns is is n, that is the dimension of theta. So if you have the number if you have number of equations less than the number of unknowns, then you have infinite number of solutions, right. If you if you have number of equations is equal to number of unknowns then you have exact solution; when provided that the given set of equations are independent, if they if they dependent you do not get it.

Secondly, that this so obviously now why you could not do these? Suppose you have n, n equations rather suppose, you have capital N number of equations and you have and this dimension is n. So you take the first n equation, invert, solve the equations, so you will get one value of theta. Now the point is that, if if you took any other different set of equations; suppose you took these n, you can again solve, you will get a different thing, you will get different theta than than what you got in the first time. So if you try to solve exactly; then there there will be no single theta which means satisfy all the equations, because your number of equations are more than than number of unknowns.

So you so when the this is this is the most difficult situation, when the when the number of equation is more than number of unknowns. So in that case what what you like to do is that, you like to get a good approximate solution for all the points rather than getting an exact solution. And for example; typically speaking if you have points like this, you could do these two points and then find out one line which will be quite quite erroneous with respect to these points or you could take these two points, these two points and then find another line.

So in general you will find, that is if you do that you will get large very large error with respect to the other points which you are not included in your fitting. So so what you must do is that, you must find a line which does not satisfy any of the equations exactly but satisfies all of them approximately, right. So you must find an approximate solution for the whole dataset as such, so that is what we are going to do.

(Refer Slide Time: 18:16)



©CET I.I.T. KGP

Least Squares Estimation

$$V_N(\theta) = \sum_{k=1}^{N} [y(k) - \phi^T(k)\theta]^2$$

$$= [Y - \Phi\theta]^T [Y - \Phi\theta]$$

$$\frac{\partial V_N(\theta)}{\partial \theta} = 0 \Rightarrow$$

$$\boxed{\theta_{LS} = [\Phi^T \Phi]^{-1} \Phi^T Y}$$ [Provided The inverse exists]

$Y = [y_1 \dots y_N]$
$\Phi = [\phi_{11} \dots \phi_{1n};\ \phi_{N1} \dots \phi_{Nn}]$
θ

So so whether is usual and simple, that is first of all I define and some of errors. So if I choose that parameter theta, then what was what could have been my error in estimating for each one of those points? And then make a some of squares and I say that, this is my performance criterion, I want to minimise it. So find theta which minimizes this, that is the problem.

Now this can very easily you can you can understand that that that this can be written like this; where this y is nothing but so I written in vector matrix rotations, so basically all these equations if you stack, so y will be equal to y 1 to y n this is your Y, capital Y is equal to your your matrix. This thing is this is your phi 1 1 to phi 1 n, small n. And this is phi n 1 to phi n, n, this is your capital phi matrix, this is your capital Y matrix and theta is theta. So this all the set of equations you can write like this, y is equal to phi theta that is your matrix equation.

So if you want to find out this square, it will be Y minus phi theta transpose, Y minus phi theta that will give you, that will give you V N theta. See this Y is the column vector; so if you make it transpose it will become a row vector, this will become a column vector multiplication will give a will give a scalar which is this sum, you can verify it very easily. So our usual way, that we have to we have differentiated with respect to theta and that is equal to zero, that will immediately give you if you try it, it is very well known nothing great about it, we have we have also try it in in other context before. So it will immediately give you that this is the least square solution to this problem, provided the inverse exists; provided these inverse exists otherwise you cannot obtain it obviously. So so this is your least square solution to this problem, if this inverse exists, otherwise there is no solution. So now what will do..

Student < sir this phi is the liniment of the view model>

[Conversation between Student and Professor – Not audible ((00:20:52 min))]

phi are the phi are the explanatory variables; you have some variables you have measured and you want to, use using these variables you want to explain another variable,

Student <is it other model?>

[Conversation between Student and Professor – Not audible ((00:21:02 min))]

Other model yes. Now signals are common to models and systems; signals signals are generated from system but we do not know how they are generated, but we would like to postulate some form of the model and then we like to find out the the best parameters in within that form, which explains this signal is the best.

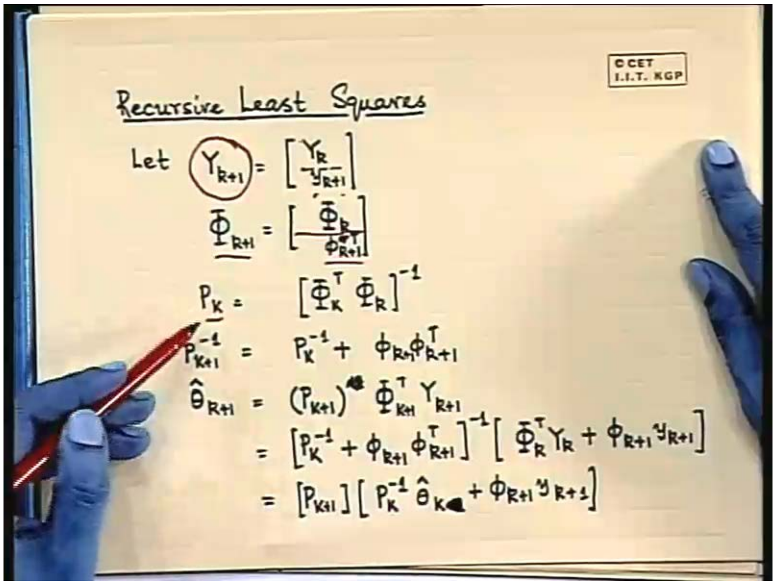
[Conversation between Student and Professor – Not audible ((00:21:24 min))]

theta I said the parameters, it could be registered inductance value, if you have a circuit.

Student < model parameters?>

Yes, model parameters, right.

(Refer Slide Time: 21:34)



©CET
I.I.T. KGP

Recursive Least Squares

Let $Y_{k+1} = \begin{bmatrix} Y_k \\ \hline y_{k+1} \end{bmatrix}$

$\underline{\Phi_{k+1}} = \begin{bmatrix} \Phi_k \\ \hline \underline{\phi_{k+1}^T} \end{bmatrix}$

$P_k = \left[\Phi_k^T \Phi_k\right]^{-1}$

$P_{k+1}^{-1} = P_k^{-1} + \phi_{k+1}\phi_{k+1}^T$

$\hat{\theta}_{k+1} = (P_{k+1}) \Phi_{k+1}^T Y_{k+1}$

$= \left[P_k^{-1} + \phi_{k+1}\phi_{k+1}^T\right]^{-1}\left[\Phi_k^T Y_k + \phi_{k+1} y_{k+1}\right]$

$= \left[P_{k+1}\right]\left[P_k^{-1}\hat{\theta}_k + \phi_{k+1} y_{k+1}\right]$

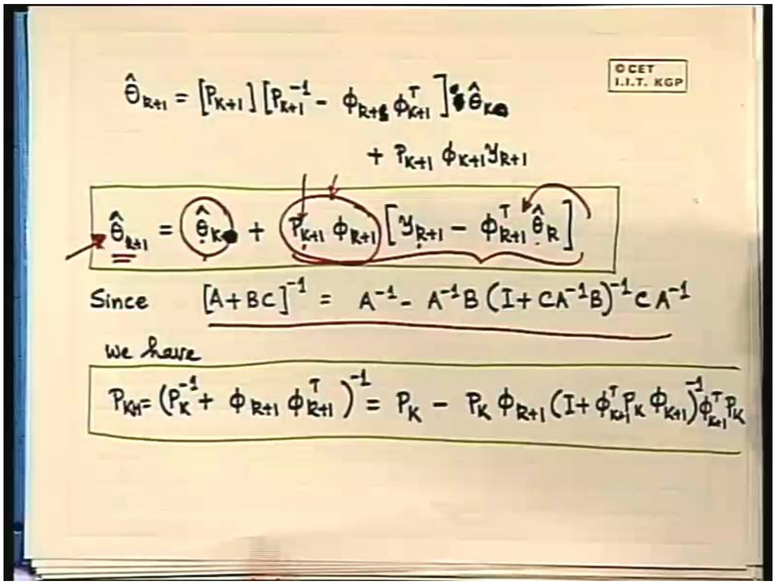
So this is my least square solution. Now obviously, these these concepts by it knew known that in various situations; we would like to I mean every time I get a new data, if I have to if I have to solve these matrix again, remember that this matrix this side grows with N, N is the number of data points, it is it is actually a very large matrix. So so so I mean every time you get a new data new data point, if you have to again recompute this configuration then solve it, it is a it is a big wastage of computation, so we would like to recursive it, that is very simple.

So it turns out that, now you see you see that we I have just written that, if you have one more data point then how does capital Y look in terms of small y? So another element, so it is like this previous Y k and then another element added at the end, this vector will become one more element longer. Similarly this phi matrix will become the old matrix plus another row at the bottom correct, theta is same.

So if you define the P K is is, now now now we have to find out; basically you have to find out that, if we know the solution if you know if you have already computed this solution for K K sets of data, how I can use that solution and then make little update so that, I get the same solution for K plus one data without really re-computing this equation, that is what I am trying to do. So I am defining, these are this this is just you know, simple I mean mathematical manipulations. So I can show that theta had K plus 1 will will turn out to be define this inverse as P K, whole inverse whole solution is actual solution that I want to compute is this inverse this. So this inverse for K elements I have defined as P.

So obviously what is my theta K plus 1, P K P K plus 1, phi transpose K plus 1, Y k plus 1. This is P K, this P. So P phi transpose Y. So now I am just simply writing it in terms of the previous ones and the new data points that I have got, I have to express that that is a lot of that is just a manipulation. So if I manipulate, this is the simplest possible derivation I got, there are more complex derivations than this but it turns out that; this is your update, so if you already had computed the least square solution for K data points, then the then a new solution for in that includes K plus one data points, will will be given like this.

(Refer Slide Time: 24:19)



© CET
I.I.T. KGP

$$\hat{\theta}_{K+1} = [P_{K+1}][P_{K+1}^{-1} - \phi_{K+1}\phi_{K+1}^{T}]\hat{\theta}_{K} + P_{K+1}\phi_{K+1}y_{K+1}$$

$$\boxed{\hat{\theta}_{K+1} = \hat{\theta}_{K} + P_{K+1}\phi_{K+1}[y_{K+1} - \phi_{K+1}^{T}\hat{\theta}_{K}]}$$

Since $[A+BC]^{-1} = A^{-1} - A^{-1}B(I+CA^{-1}B)^{-1}CA^{-1}$

we have

$$\boxed{P_{K+1} = (P_{K}^{-1} + \phi_{K+1}\phi_{K+1}^{T})^{-1} = P_{K} - P_{K}\phi_{K+1}(I+\phi_{K+1}^{T}P_{K}\phi_{K+1})^{-1}\phi_{K+1}^{T}P_{K}}$$

Again you see this same form, everywhere you will get the old solution plus another collection term, that collection term will have a gain and it will have an error everywhere this thing comes, all estimated will have this structure. So here is the old estimate plus a collection term; same thing is to happen in the Kalman filter, x hat K, given K is equal to X hat K given K minus one plus K K into same thing.

So here also you are having a gain vector and you are having an error, this is the error. So the question is that the old parameter, can it explain the new measurement? If it can explain the new measurement then this error will be zero. If it cannot explain, then it needs to be changed. So now this decides; so this says how much, what is the magnitude of change? So this is the scalar and this decides that the change should be made in what direction, right because this is a vector. So everywhere you get the same thing, you get the same thing in observers, you get same thing in Kalman filter, you get the same thing in parameter estimators, you get the same thing in neural network training, everywhere it is same thing.

So so this is your update equation, only thing is that you have to compute this now. So you have to update this also. So for updating this we use what is known as matrix inversion lemma; which says A plus B C, this is this is a this is an identity which holds for I mean dimensionally compatible matrices. Now you have to have to have to remember it now in not the form, but we must remember that such thing exists, do not be do not think that you have to remember it and then we sight it, anyway usually it is available.

So so then using this lemma P K plus 1 will be given like this, which again has similar forms too there are of Kalman filters. Incidentally, here I have written this I but actually this thing turns out to be a scalar, this is a number. So you can you could have you could also write, P phi phi transpose P divided by; when you have a scalar you can generally divided by, when you have matrix, you write something inverse.

So this turns out to be a scalar, why because if you take just take dimensionally; this turn out to be a scalar. So so so this is by recursive least square estimator, right. So what I have what I have done? I have first take got a set of measurements, I proposed a I proposed a model form, I simply figured it in the least square manner; and then I found a way of updating my least square solution, if I get another measurement that is so so I can do it recursively every time I get a new measurement, I can just once update this equation that, that will give you the new least square solution.

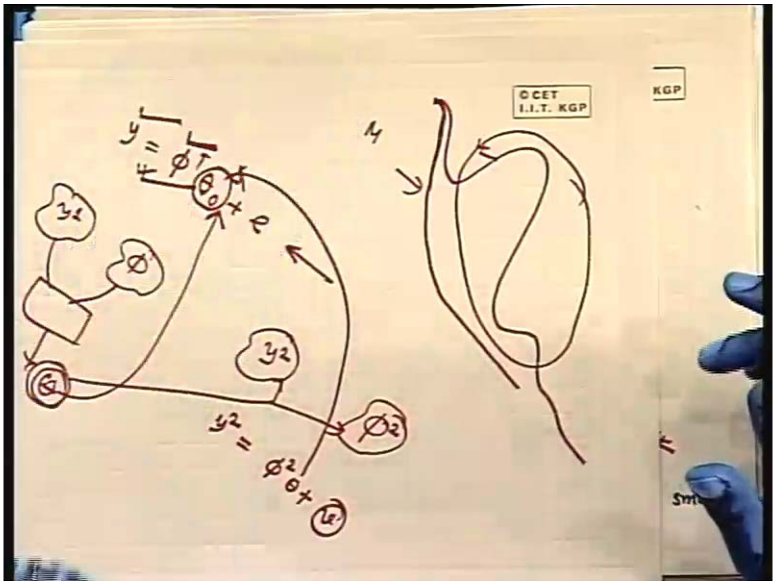
So having done that, we need to know how good this estimator is, right. And and we need to answer one very important question, definitely this will give me the least square error; whatever way the data were generated because that is a explicit thing, for those data points using you cannot get a linear estimator which will give you lower least, lower square error, that fact is true whether the data were generated by a non-linear system or not, that is true. But why are you making this estimator? You are making this estimator because once you find those thetas, you are going to use it on other data points.

See normally, why do why do you build the builder system model? We build a system model because we are going to use that model elsewhere. So typically speaking; this we must remember this another thing that we typically use some amount of data for for building a model and we and we employ methods such that, the model that we get explains that data, very well. That is that is by a by basically by virtue of our method or or for example for this case; we got some data points Y one, Y two, Y three, Y n and we found a least square estimate, so so for particularly for those set of measurements the parameter vector that I have got, naturally we will fit those data, it it is forced to fit that data, but the question is that, now can I use this parameter theta, generally?

That is can I take this system and then in future generate different data and then use this parameter to explain it, will it work? That is the main main question, otherwise can the model generalize; it has been built using a certain data set, does it work well for other data sets, if it does not work well for other data set, that model is useless. So we always look for a kind of generalisation property of models, right. This is this is this is a big thing which has to be ensured with neural networks; otherwise I mean neural network could if if you it could also happen for neural network incidentally which generally does not happen here, is that it could the the given data say it could model so nicely, that it will work badly for for other data sets, that also could happen, that is you could have over fitting not only under fitting.

So so for that matter, so so therefore with respect to these data sets, I will I will get least square error, no problem but the question, but if I want to now suppose; so now what is the question I am I am asking, when when will this model will work well for other data sets as well, when will it work well? So if you want to understand that then you have to made an assumption about this system; how the data are generated, that is how the system generates data. So if you assume that, the system generates data using a same kind of data set;

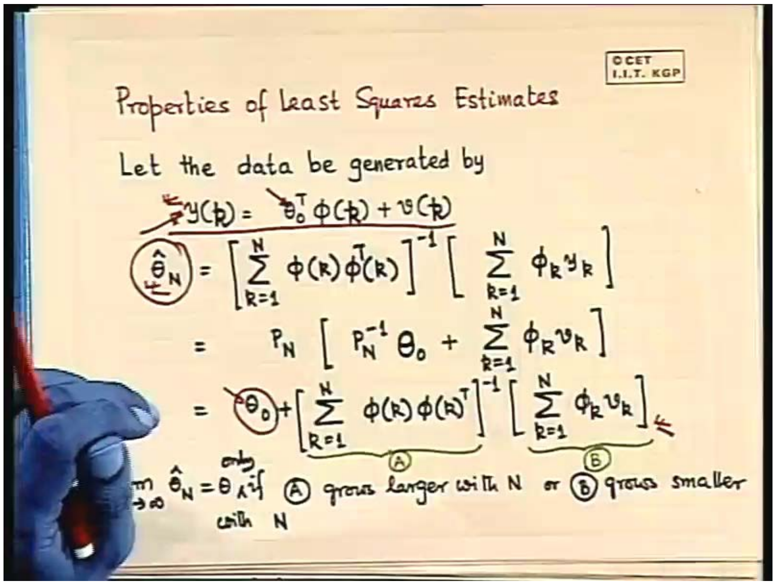
(Refer Slide Time: 30:38)



this is the way the system generates phi, given phi the system gives out y. This theta naught is unknown it is inside the system. Now you have got one set of y s and one set of phi s and you use it in your estimator and you and you got a theta. Now if you take another phi from the system and and another phi phi from the system, will this theta work for this? That is say; this is the first data set, this the second data said set then will y 2 be also given by phi 2 theta plus some error, v get this error will be small, will it work ?

Obviously it will work, if if if this theta is closed to this theta naught because the system generates data like this. So if this theta is equal to this theta naught then for all data sets it it will match. So therefore it is it is relevant to ask; that is whether this theta is close to this theta naught, if we assume that the system generates data using the same form. It is it is that is why it is it is relevant to ask for this problem to ensure that, your model will be of general usability, okay. Hence, so therefore we we look for such property. For example, so first of all I will assume that the data is generated by this model, that this is the system model.

(Refer Slide Time: 32:16)



©CET I.I.T. KGP

Properties of Least Squares Estimates

Let the data be generated by

$$y(k) = \theta_0^T \phi(k) + v(k)$$

$$\hat{\theta}_N = \left[\sum_{k=1}^{N} \phi(k)\phi^T(k)\right]^{-1}\left[\sum_{k=1}^{N} \phi_k y_k\right]$$

$$= P_N \left[P_N^{-1}\theta_0 + \sum_{k=1}^{N} \phi_k v_k\right]$$

$$= \theta_0 + \underbrace{\left[\sum_{k=1}^{N} \phi(k)\phi(k)^T\right]^{-1}}_{A}\underbrace{\left[\sum_{k=1}^{N} \phi_k v_k\right]}_{B}$$

$\lim_{N \to \infty} \hat{\theta}_N = \theta_0$ only if (A) grows larger with N or (B) grows smaller with N

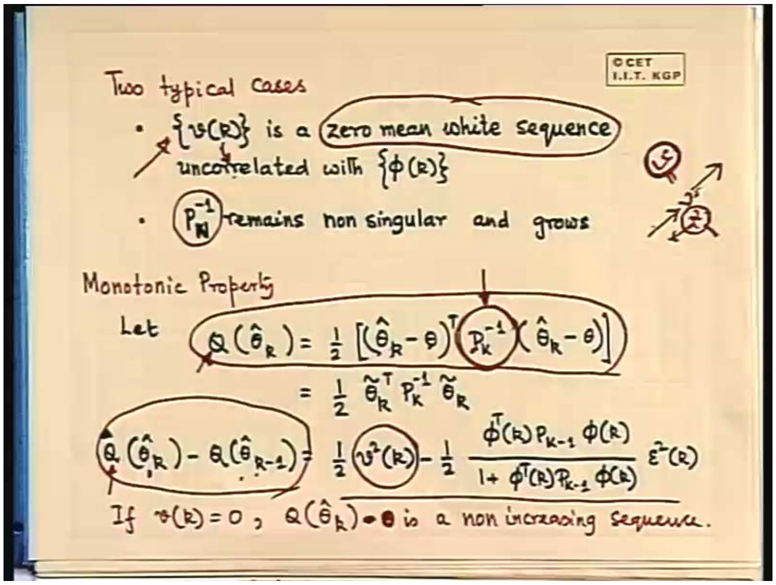
So and I have taken N such data points which have come from this system and I have estimated a parameter vector. So now I am explicitly putting hat, meaning that it is an estimate of my model, this is the true model; true model means something which this the system itself is using for generating data. So obviously will all would like to know, I mean it is therefore important to ask whether this is is closed to this but even equal to this, does it happen? So it turns out that if you do a little bit of manipulation and if you if you use this relation between y and phi; which is property of the system, that theta hat N turns out to be this theta naught plus this term.

So this will be close to the true one which the system itself we using, provided this term is one. Now when when is this term small? Either when this term is small, the second term B is small or when this bracketed term becomes larger and larger because this is inverse. It is like you know y by z, so when does y by z becomes small; either when y goes small, smaller and smaller or z goes larger and larger. So so it is important to know whether that happens in our algorithm, only under that condition even with this is the same system and model structure; you can hope to get the true parameter which the system is using, otherwise even using the same model structure you cannot, your your estimator will never given the true model.

So it does happen on the certain condition. So what are the conditions? First condition is that, these goes to zero. How how do you say that, this goes to zero, when will it goes to zero; when this V K and phi K are going to be orthogonal or uncorrelated over sum, because because this is like an expectation, right. So that is why this these are the reasons for which you will find such assumptions are made, that the that the noise which the system is which is which is going into the system is orthogonal to these data points and we will see that; in our in our dynamic system estimation sometimes that happen, sometimes that that does not happen.

So for some models forms; for some if you write the model structure transformations in a certain way that may happen, if you write it in some other ways, it may not happen. If it does not happen then you have every chance, that you will never get the true system parameters; it is possible even though your data will be fitted in the in the least square fashion, data will always be fitted in the least square fashion, about there is no doubt.

(Refer Slide Time: 35:18)



©CET
I.I.T. KGP

Two typical cases

- $\{v(k)\}$ is a zero mean white sequence uncorrelated with $\{\phi(k)\}$
- P_N^{-1} remains non singular and grows

Monotonic Property

Let $Q(\hat{\theta}_k) = \frac{1}{2}\left[(\hat{\theta}_k - \theta)^T P_k^{-1} (\hat{\theta}_k - \theta)\right]$

$$= \frac{1}{2} \tilde{\theta}_k^T P_k^{-1} \tilde{\theta}_k$$

$$Q(\hat{\theta}_k) - Q(\hat{\theta}_{k-1}) = \frac{1}{2} v^2(k) - \frac{1}{2} \frac{\phi^T(k) P_{k-1} \phi(k)}{1 + \phi^T(k) P_{k-1} \phi(k)} \varepsilon^2(k)$$

If $v(k) = 0$, $Q(\hat{\theta}_k)$ is a non increasing sequence.

So the two typical cases which will give good results; are firstly that this is the we often assume v to be zero mean white, because this zero mean white will will ensure, actually what we need is that V k is un correlated with phi, these all that we need to know. But some but in many cases, if you assume V to be zero mean white if this is this is satisfied, you will see that in the case of our transfer functions system.

Right now we only say that, V k is uncorrelated with have to be uncorrelated with phi K and this P N inverse should must remain non-singular and must grow in size. If it grows in size then this term will grow in grow in, will it be smaller and smaller and my estimate will come closer and closer to this true system, right. So this is now there are there there are several nice properties; you know this is a this is this kind of results are called asymptotic, in the sense that we can only show that eventually as n tends to infinity this going to happen. Now now this n tends to infinity thing is you know, it is a result which it is something it says that okay; if you do it long enough it will happen, but this long enough is how long? In in general you are going to you are going to only work with finite data, so you are very interested to know that does it happen with within hundred points, within two hundred points.

Finally, it goes to something goes to zero. Now now something can go to zero like this, something can also go to zero like this. So is it that it is this is monotonic; it will always reduce this is not monotonic sometimes it may increase, sometimes it will it may decrease. All we are saying is that, it finally go to zero. So it typically if it does like this, it is likely to take longer. So so we are also interested to know that, what what more we can say about the estimates? Can we at least say that, this performance criterion is going to come down monotonically? Then at least I have little better hope that it will probably converge little faster than if it is when non-moulded. So now we can prove several such cases; for example, you can prove that if you define a performance index like this, now why did you define it like this? What is what is the relationship of this with the original one which you are minimizing?

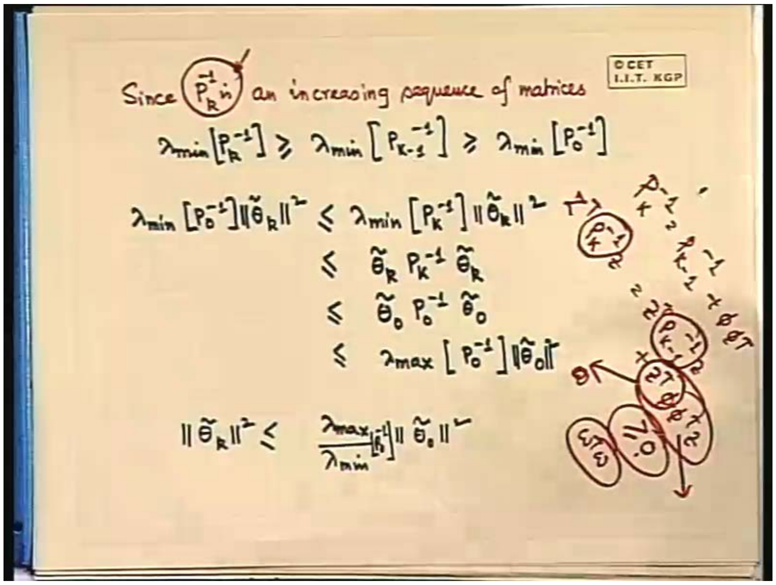
The the reason is that you have put this P K inverse. Now there is a reason, why you are put this P K inverse that will soon come to but suppose; if you define a performance index like this which is theta k minus theta, theta is the true one which is theta naught and this is your estimate. So you are you are taking a weighted sum of the error, near estimate, right and you have chosen the weight to be P K. So the weight is also changing, why did you choose such a function? For one reason that this weight incidentally goes on increasing with K; so you are gradually having larger and larger weights, this element. Now if you can show that, just imagine that suppose you have a quantity Y is equal to lambda x square; now this lambda goes continuously increases and then you can show that this Y cannot increase, it is it is bounded it will not increase ever.

So if this is increasing, this must decrease, is it not? If if this is bounded and if this is continuously increasing then this must be decreasing, right. So we are so this our x is like this, so we are going to have this this argument, if we can show that over K this Q does not increase; but but this keeps on increasing then I will be able to infer that this keeps on decreasing, right. So so it can be proved that with this thing; this the performance index that is Q at theta at k and Q at theta at k minus 1, if I take the performance this performance in index, if I evaluate at k minus one and at K, it turns out to be like this.

So at least if if this if in the system has no noise suppose; this V K is zero, there is a system generated pure data using some unknown parameter but there was no noise then this is a purely positive quantity. See phi transpose P phi is a positive quantity, phi transpose theta is a positive quantity; so there ratio is also a positive quantity, this epsilon square is also positive. Here is the minus which means that; Q theta hat K is strictly less than Q theta at K minus 1, it will be less than or equal to it cannot increase.

So so so this performance criterion this performance criterion, can strict if the system did not have a noise then the this performance criterion would have can only decrease; while this goes on increasing but for, what will it prove? It will prove that, this must always decrease; so the parameter error will strictly come down, it cannot increase so so so the convergence is is not like this, convergence is like this, right.

(Refer Slide Time: 41:22)



© CET I.I.T. KGP

Since P_k^{-1} is an increasing sequence of matrices

$$\lambda_{min}[P_k^{-1}] \geqslant \lambda_{min}[P_{k-1}^{-1}] \geqslant \lambda_{min}[P_0^{-1}]$$

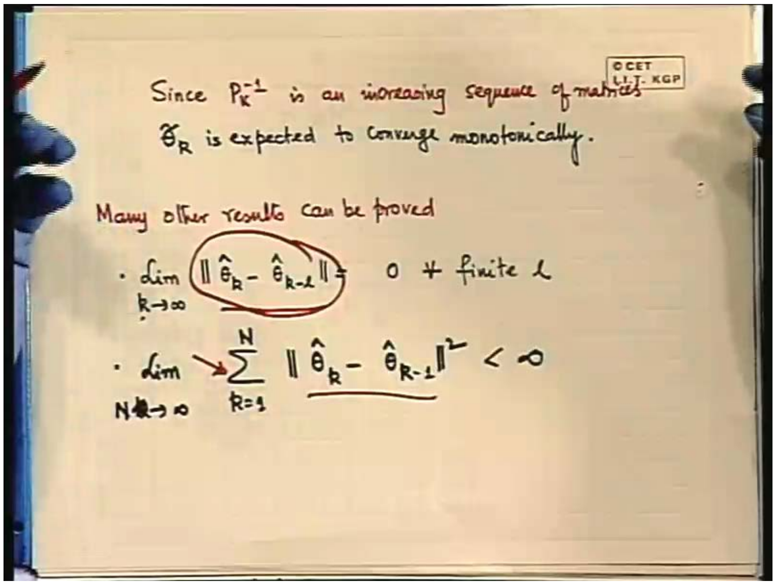
$$\begin{aligned}\lambda_{min}[P_0^{-1}]\|\tilde{\theta}_k\|^2 &\leqslant \lambda_{min}[P_k^{-1}]\|\tilde{\theta}_k\|^2 \\ &\leqslant \tilde{\theta}_k P_k^{-1} \tilde{\theta}_k \\ &\leqslant \tilde{\theta}_0 P_0^{-1} \tilde{\theta}_0 \\ &\leqslant \lambda_{max}[P_0^{-1}]\|\tilde{\theta}_0\|^2\end{aligned}$$

$$\|\tilde{\theta}_k\|^2 \leq \frac{\lambda_{max}}{\lambda_{min}}[P_0^{-1}]\|\tilde{\theta}_0\|^2$$

And why is it an why is it an increasing sequence? That is y is this a more positive definite matrix than P K minus 1 inverse? Because P K inverse is PK minus 1 inverse plus phi phi transpose. So if you take any vector Z and do z transpose P K inverse Z then it will be z transpose P K minus 1 inverse z plus z transpose phi phi transpose z.

So these can can only be greater than or equal to 0, because it is a product of; if you recall Z transform are equal to w then it is like w transpose w, if it is w transpose w it cannot be less than 0, it must be greater than equal to 0. If this greater than equal to zero then which means that, this this is a more positive definite matrix than this, correct. So in that sense P K, P K inverse keeps on increasing. So therefore you you can do it much more regressively mathematically, but you can show that it will it will continuously decrease.

(Refer Slide Time: 42:37)



Since P_k^{-1} is an increasing sequence of matrices $\tilde{\theta}_k$ is expected to converge monotonically.

Many other results can be proved

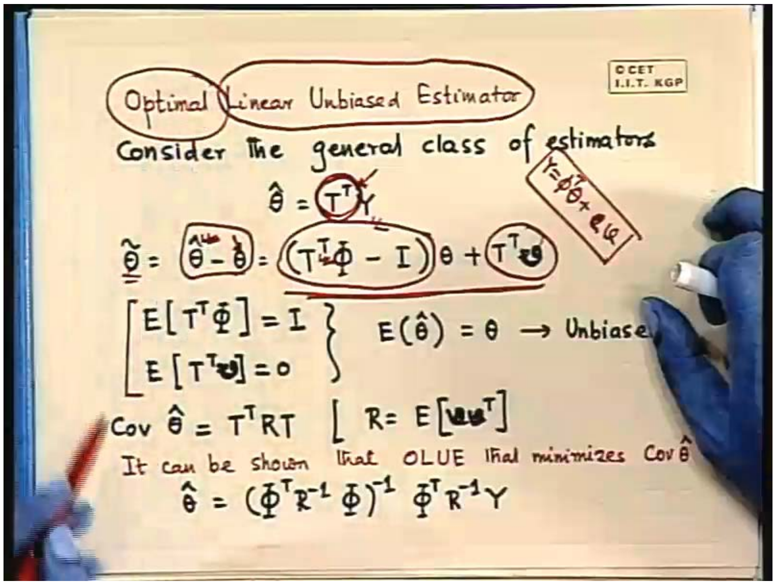
- $\lim_{k \to \infty} \underline{\| \hat{\theta}_k - \hat{\theta}_{k-l} \|} = 0 \;\forall$ finite l
- $\lim_{N \to \infty} \searrow \sum_{k=1}^{N} \| \underline{\hat{\theta}_k - \hat{\theta}_{k-1}} \|^2 < \infty$

So at least you know that, if your model structure was correct; that is if the system indeed use use some unknown theta and and generated the data, using y is equal to phi transpose theta then your least square algorithm will eventually find that theta if it is presented with enough data right. So then you can you can prove, you can prove many other results, for example you can prove that; in fact we will we will do some of them in our tutorials, that as K to the infinity, what does this may mean? Mean this equation means what? If I can prove it, it means that the parameters will finally converged to a constant value, because if k goes large enough then K plus three and k will be same, K plus phi and K will be same.

For any finite l, these these parameters are going to be same; which means the parameters will will not continuously change, they will finally arive come to a stable value. So you can prove many such results. This says this says something about, how fast it will how fast it will converge? So this this is the sum the sum itself will be less than infinite, if you want to if you want to understand these things; you have to understand properties of infinite series, so that we we do not intend to do but all I wanted to say is that, you can show that if the model structure was ideal and if there was no disturbance then the least square algorithm will converge to the true parameters, monotonically, so at least it has some good properties.

Now the question is that, this this one way but sometimes, you can ask that okay you can ask many other things, for example you can ask that; if I consider the general class of linear estimate, that is given the y's I want to generate some estimated using some arbitrary linear combination of Y.

(Refer Slide Time: 44:38)



©CET I.I.T. KGP

Optimal Linear Unbiased Estimator

Consider the general class of estimators

$\hat{\theta} = T^T Y$

$Y = \phi^T\theta + e(v)$

$\tilde{\theta} = \hat{\theta} - \theta = (T^T\Phi - I)\theta + T^T v$

$$\left[\begin{array}{l} E[T^T\Phi] = I \\ E[T^T v] = 0 \end{array}\right\} \quad E(\hat{\theta}) = \theta \rightarrow \text{Unbiase}$$

$\text{Cov}\ \hat{\theta} = T^T R T \quad [\ R = E[vv^T]]$

It can be shown that OLUE that minimizes $\text{Cov}\ \hat{\theta}$

$$\hat{\theta} = (\Phi^T R^{-1} \Phi)^{-1} \Phi^T R^{-1} Y$$

I I want to use some matrix here, I do not know what matrix. So what should be the property of these matrices, so that I will get something I will get an estimator, how can I do that? So it turns out that if you put that that is again data is generated; if data is generated as y is equal to phi transpose theta plus in this case e or v whatever say v, then if you take any any arbitrary what I am saying is that the least square estimator is just one linear estimate, there may be other linear estimator in fact there is a there is very important linear estimator, which we will study in a future class called instrumental variables. They have very nice probabilities; they give you much better estimates over least square using a very computational efficient form.

So it is not that you have to always use the standard least square standard least square estimate. So what you are trying to see is that, if we construct a a a general class of estimators which are generated from Y in some way; this this T what this T I do not know, but what should be the property of T such that if I if I generated using this T then this theta delta which is the error between the true and the estimated parameter will be low, if I ask this question. So if I if I calculate this, this will be my theta delta using this equation as the data generating equation; then it turns out that, if theta delta has to be zero then T should have this property.

Property number one, that this should be equal to identity and if if if this is equal to identity and this is zero and this should be zero, and what does it mean? That that this matrix T should be such that, it is where you well correlated with the actual data phi; so so it is not necessary that you have to use the data, you can use something else other than data but you must use use something which is very correlated with the data and very uncorrelated with it. If you can do that then whatever you put in this T matrix, will give you a estimator. In fact we will see latter especially in the context of the instrumental variable, that sometimes not using the data itself has great benefits, right.

So so this was you know then so any linear unbiased estimator, but what what do you what do I mean by unbiased? Un-biased means that, asymptotically this theta hat will be will will tend to theta, if if that happens then we call an estimator as unbiased. So and then you can you can what is an optimal unbiased estimator is that, if you go for that that of all unbiased estimator which one gives you the least co-variants, that also you can ask. So fine so we will have to close today. So before closing that all I want to say is that, so what did we do?

We first find out found out a a basic least square solution to a problem, then we found that under the assumption that the data has been generated using a model of the same type we can get some good properties from the least square estimate; that is a that is a dead estimated parameter will finally convulsed to quote unquote the true parameter which which the system used to generate, the data that we found. And lastly we found that that, it is not always necessary to go for a least square estimate; sometimes it may be may be good that you will rather than putting the directly the data, you can put slightly modified data or some other matrix only thing is that, whatever we use you must ensure these two properties.

So if you can you can ensure this property; you will get what is known as a linear unbiased estimator. So we will we will continue with this and we will see especially in in the next class, that there are several variants of the least square, this is the basic least square problem; you can toss and turn this problem in a in a little different way here and there, and you can get new new estimators which are used, which will be useful in several various various contexts. We will see that tomorrow, thank you very much.