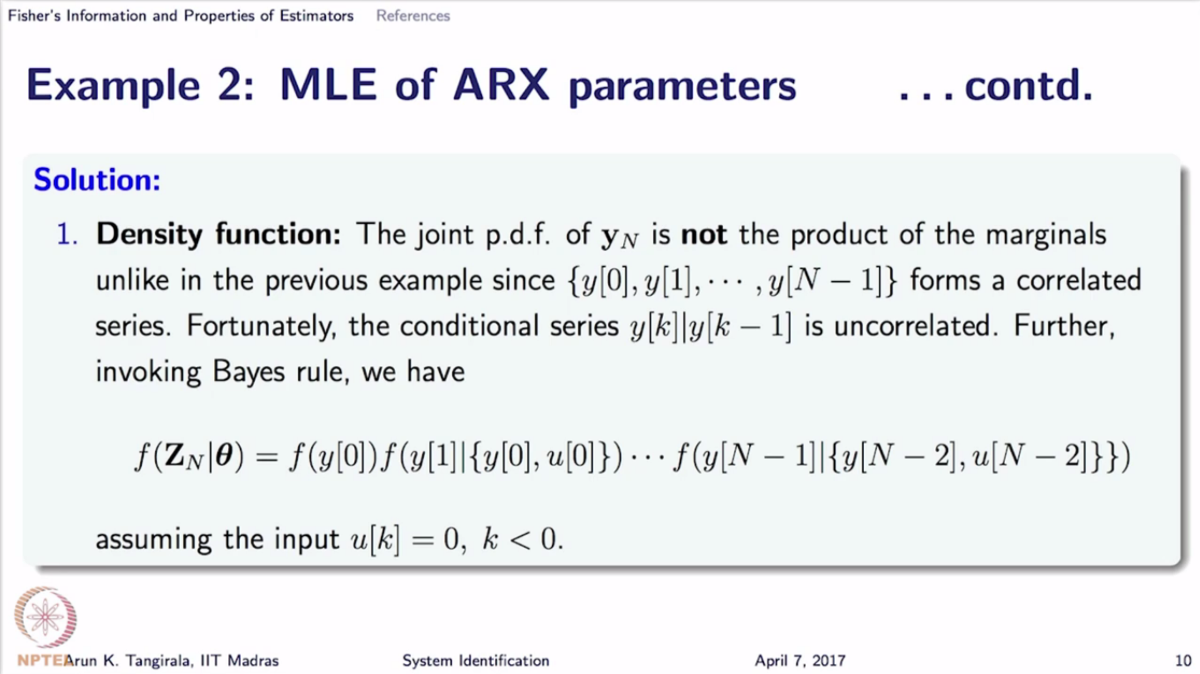
CH5230: System Identification

Fisher's information and properties of estimators

Part 15

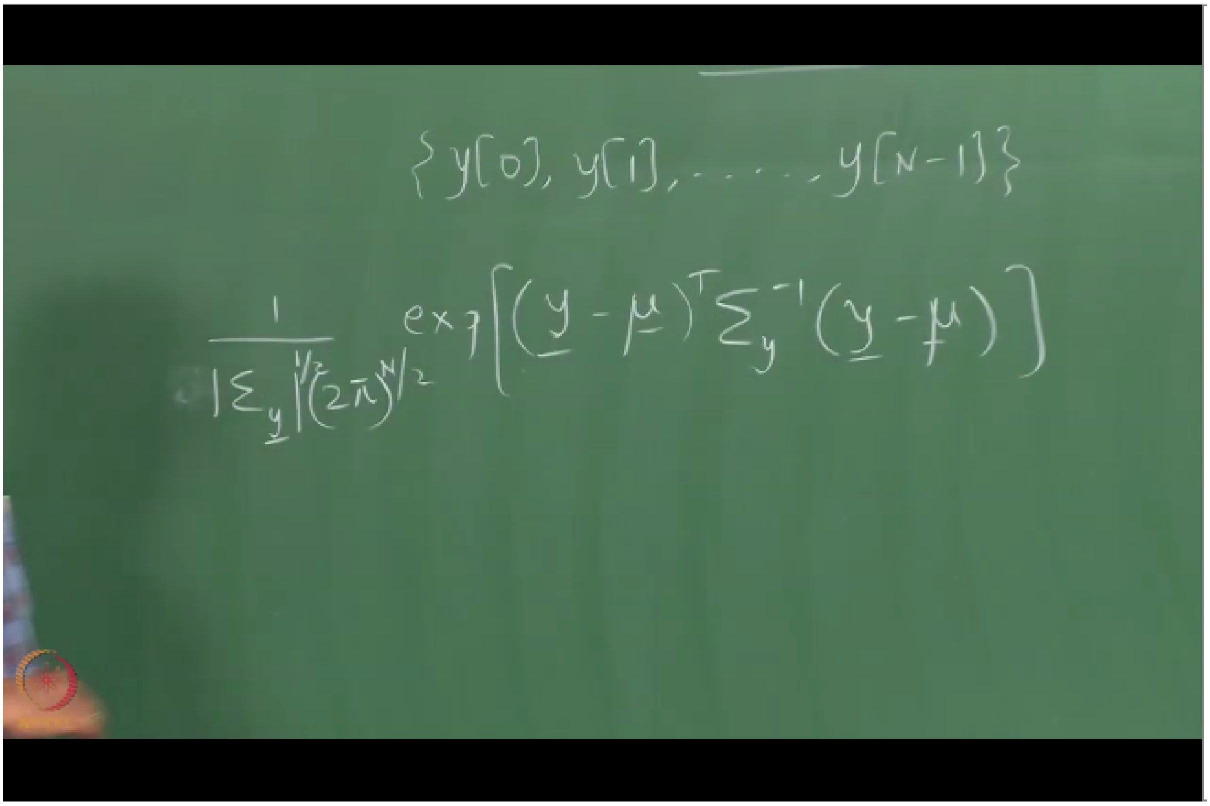
Let's get going with the example. So, we are looking at how to set up the likelihood, for estimating the parameters of an ARX model. And as I had explained yesterday, the trick is to observe that although the original series, the given series is correlated, the unconditional series, sorry, the conditional series is uncorrelated.

(Refer Slide Time:0:42)



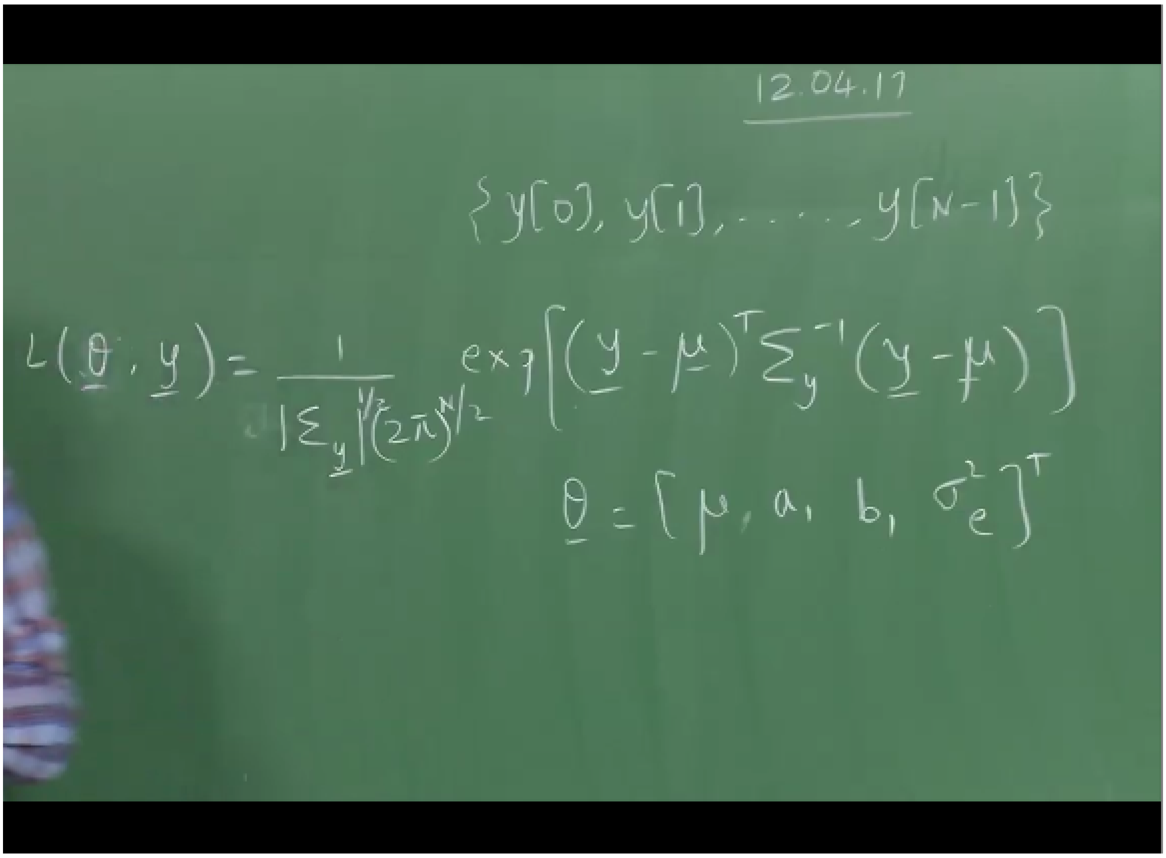
So the given series is this one here, of course, I'm not mentioning the input, the input is always assumed to be given in System Identification. This is correlated and setting up the joint PDF of this is not so easy. So in general, if you're given N observations and if you know they are jointly Gaussian distributed, there is a joint Gaussian distribution that you would set up, which is what in the literature they would call as Exact-likelihood, that is, if I'm given a vector of observations like this, then-- and suppose this vector of observations follow a joint Gaussian distribution, then you have this, right, so, if you-- so, this is the joint Gaussian distribution.

(Refer Slide Time 1:56)



So, if you assume that this N observations follow a joint Gaussian distribution, which they do if the driving forces Gaussian white noise, then this is how the likelihood would be of-- Generally, likelihood is a function of theta. But to be able to work with this likelihood, you need to-- so you look at the parameters here, right. If you have to identify the parameters here, strictly speaking, the parameters are mu and then the model parameters, a1, b1 and then sigma square, transpose. These are the unknowns.

(Refer Slide Time 2:48)

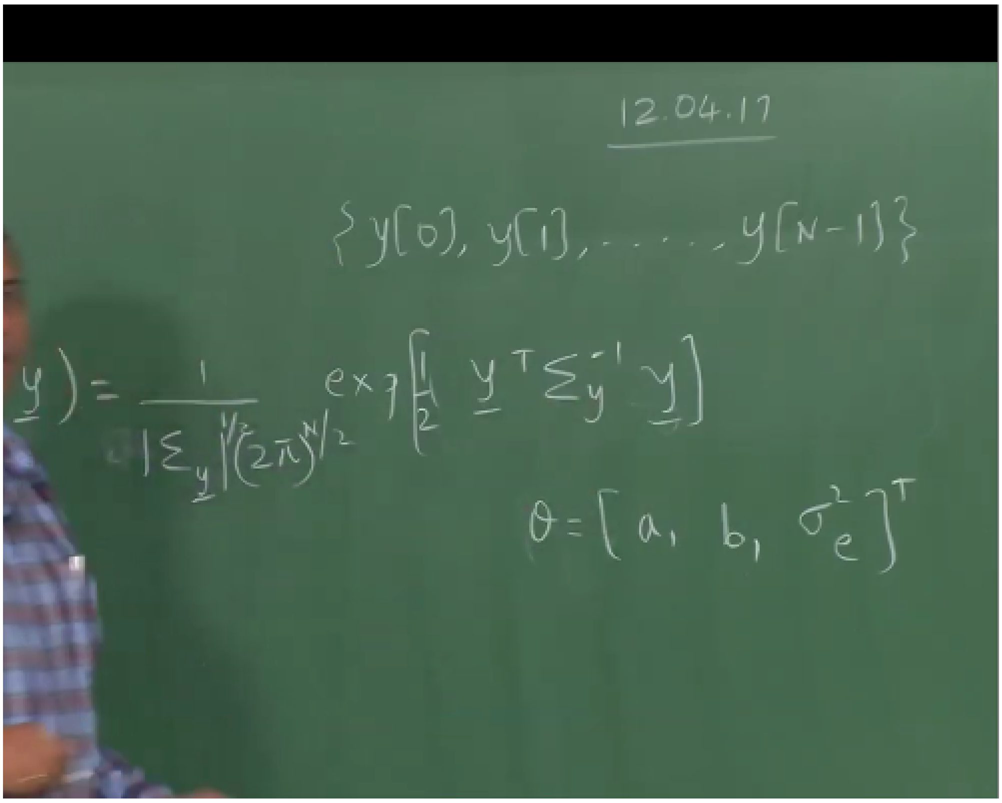


As far as this ARX model is concerned these are the four parameters of interest. Mean of y, in this case, mean would work out to be zero and then you have a and b coming from the model, and then sigma square e, which is the variance of the driving force, which is a white noise. Now, in this likelihood, of course, what, if you were to work with this likelihood, mu, of course, appears explicitly, but a, b and sigma square e will make their presence in this covariance matrix.

This sigma y, what is the size of this sigma y? There is a half-- I'm sorry. What is the size of this sigma y? N cross N. So you need to be now, be able to express your sigma y in terms of a, b even in mu maybe, possibly, mu itself maybe function of a and b and mu e, but we know that if the input is of zero mean, and if Gaussian white noise ek is of zero mean, then you can expect y also to be of zero mean.

So you can throw away mu, that's okay. At least you can straight away dismiss mu and say well, mu is zero mean. But the remaining a, b and sigma square e, will make their presence in sigma. So let me erase mean from the scene here. So I can say that this is simply, we can omit mu from the list of parameters. And now you will have to write sigma y in terms of theta.

(Refer Slide Time 4:50)



Can we do that? Can we write the noise covariance matrix in terms of a, b and sigma square e? What does sigma y consist of? How does it look like?

So, sigma y looks like this. Sigma square y along the-- assume stationary. Sigma square y along the diagonals and then you have, the auto covariance. Now, obviously, the variance and the auto covariance, all of them are functions of a1, b1 and sigma square e, right? So, given this model you should be able to, given yk equals minus a1, yk minus 1 plus b1 uk minus 1 plus sigma square e, you should be able to derive sigma square y and the auto co variances in terms of a, b and sigma square e.

That should be easy, right? I mean, you just have to write down the equations with involving expectations, but ultimately you will realize that this sigma y is a function of theta. So this sigma y that we're talking of is a function of theta. So it is not that we cannot set up the likelihood because the observations are correlated. You should not be under that impression. Except that, this likelihood now is a complicated function of theta. And remember on top of it, you're dealing with the inverse of sigma y.

So the inverse of sigma-- computing the inverse of sigma y is going to be difficult. That mean symbolically, I'm not saying numerically, and remember subsequently, we take the derivative of this likelihood with respect to theta. So which means that I will have to take the derivative of that likelihood function that I've written on the board with respect to theta, which is not going to be a very nice looking or very friendly expression.

Imagine taking the derivative of this likelihood, of course, we will take a log-likelihood, that's okay. This should not be big L, should be small L. You can take the log-likelihood, that's fine. But then remember, you're going to have a determinant of sigma y. This is also a function of theta, and here you have function of theta. Exponential will vanish one you take the logarithm, but the resulting function is going to be quite complicated for us to be able to take the derivative.

And that is why we have taken other route. The reason we went through this discussion just now is to clear any misconceptions that you may have that it is not possible to set up the likelihood for correlated processes. It is possible. This is how you set it up. For any joint Gaussian process, this is your likelihood, always. The moment you know that the observations fall out of a joint Gaussian process, and if it is zero mean, straight away you can write this likelihood. You don't need to ask anyone.

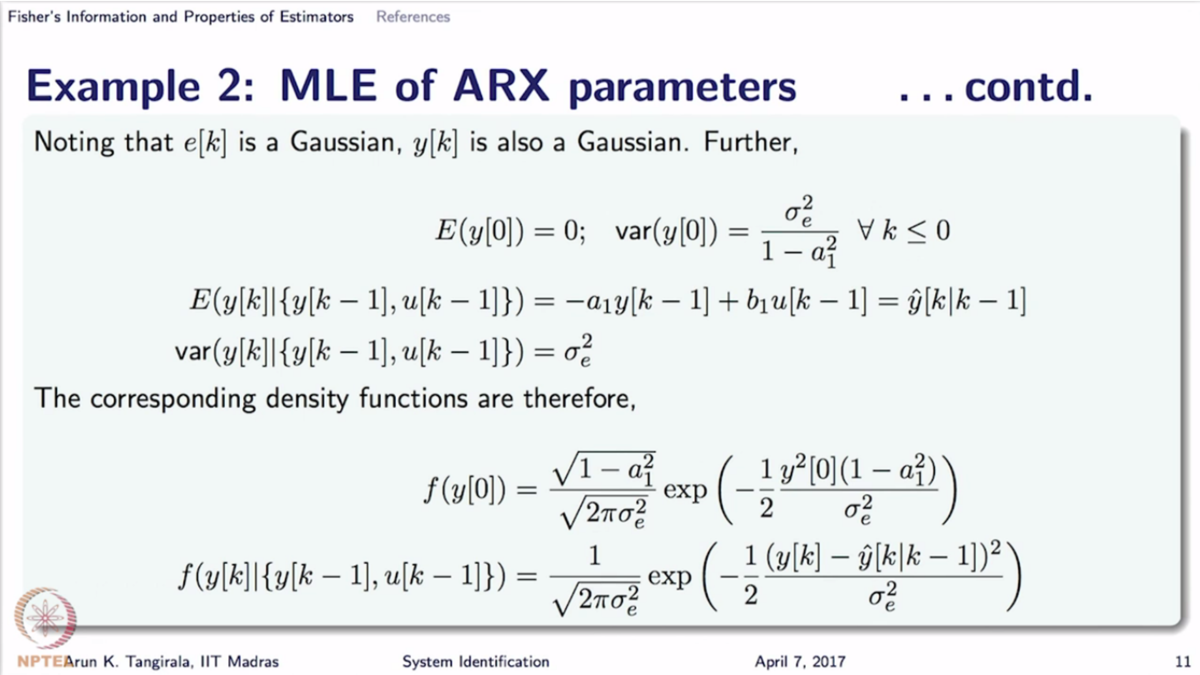
But the challenge would be, what would be the challenge? To first figure out, how the parameters theta, enter your sigma y? The ultimate parameters of interest are a, b and sigma square e here, but in some other problem, they may be something else. So the challenge is to figure out how the parameters enter sigma y and then work with the derivative of the log-likelihood and so on. That is the reason we take the other route which we went, which we discussed yesterday, where we don't write this likelihood-- this is called exact likelihood, what we are working with also is exact likelihood.

So, instead of working with f of y given theta as is, as given here, as I showed yesterday, you can break this up and write it this way. So that, now it becomes a lot easier to figure out how the parameters enter-- this is nothing but your likelihood itself. So although I've written this, this is nothing but… How the parameters enter your sigma y. So what we have observed is that under the given model assumption, the series y1 given y0, y2 given y1 up to yn minus 1 given yn minus 2, they form an uncorrelated series. But they're all anchoring on y0, right? For a given y0, a generatey1 and in turn for a given y1, a generate y2 and so on.

So y is, the randomness of y0 is also has to be respected. And f of y0 takes that into account. So what remains for us is to figure out, what is this f of y0, which is unconditional PDF and f of yk, given yk minus 1, which is a conditional PDF. And as I had mentioned yesterday, if you're dealing with an ARX, second order ARX, then the idea remains the same but the only difference is, you would condition y1 on y0 rather-- let me ask you, how would the likelihood change? Sort of misspelling it out.

If you were to be working with a second order ARX model, if you're estimating parameters of an ARX 2, 1, 1, how would this likelihood-- this factorization change?

(Refer Slide Time:11:34)

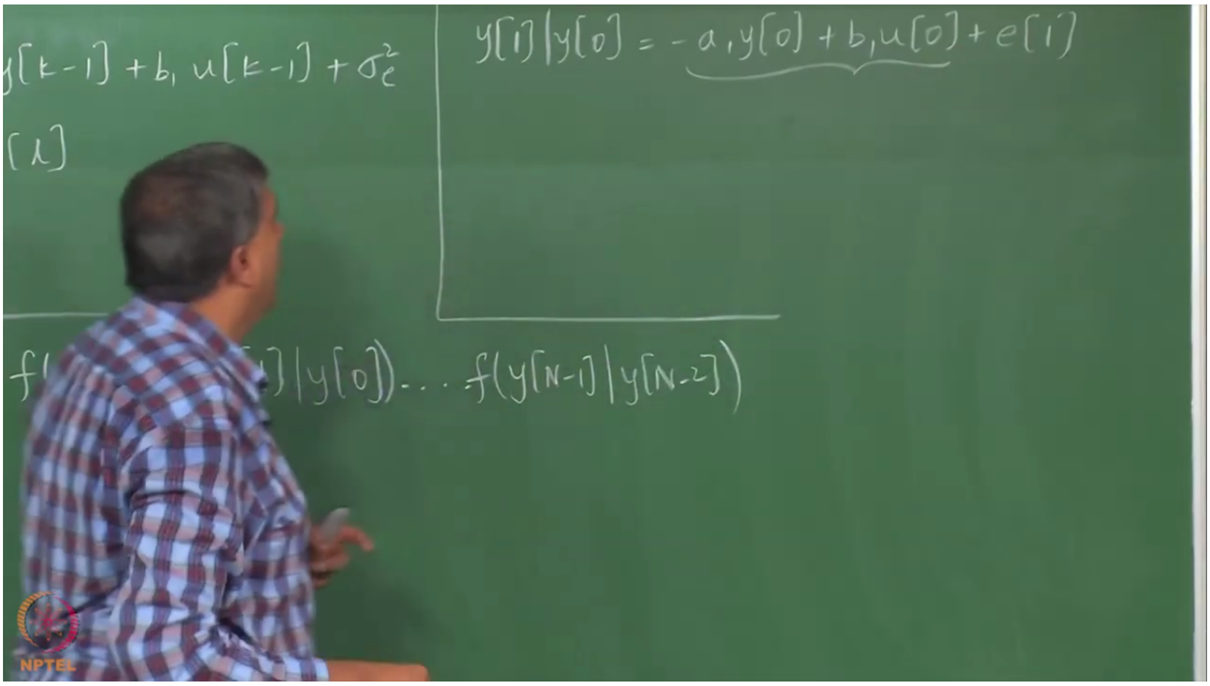


It should be a straightforward extension. If you are not able to answer that means you still not understood the basic idea, behind the factorization.

We have f of y of 0 into f of y of 1 into f of y of 2 given y1, y2.

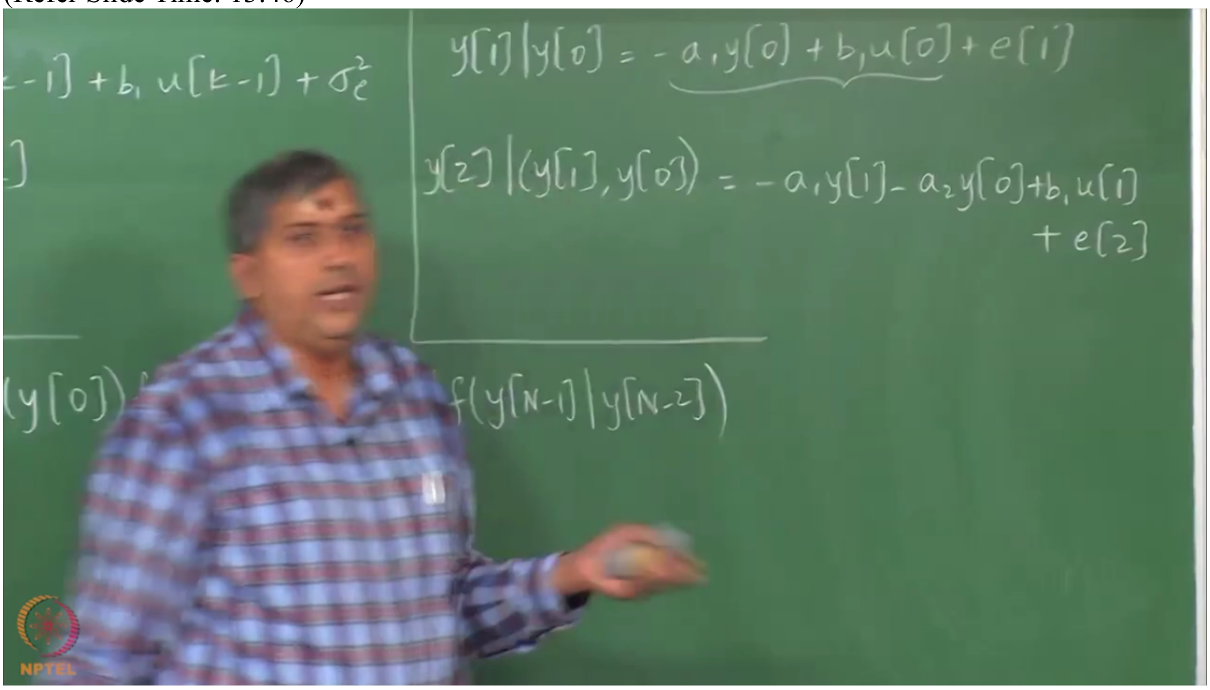
Very good. So that's all. So now you will have to condition on the first two observations. Because the series, now the conditional series would be, y2 given y1, y0, y3 given y1, y2 those would be uncorrelated. See basically what you're trying to reach at; when you write y1 given y0, we wrote this yesterday, y1 given y0 doesn't matter but even if you don't give y0, this is right hand side expression is the same, right? When you say it y0, this is no longer random, so this part is fixed. So what governs the randomness of y1 purely is e1, I mean, governance randomness of y1 given y0.

(Refer Slide Time: 13:03)



If you had a second order ARX model that you are dealing with, then you would have a-- you cannot begin with this because you need y at minus 1 as well. So you start with y of 2, right? So for a second order ARX model, you would writey2 given y1, y0 to be minus a1 y1 minus a2 y0 plus b1 u1 plus e2, right? The same thing as it just -- we have just included more terms now.

(Refer Slide Time: 13:46)



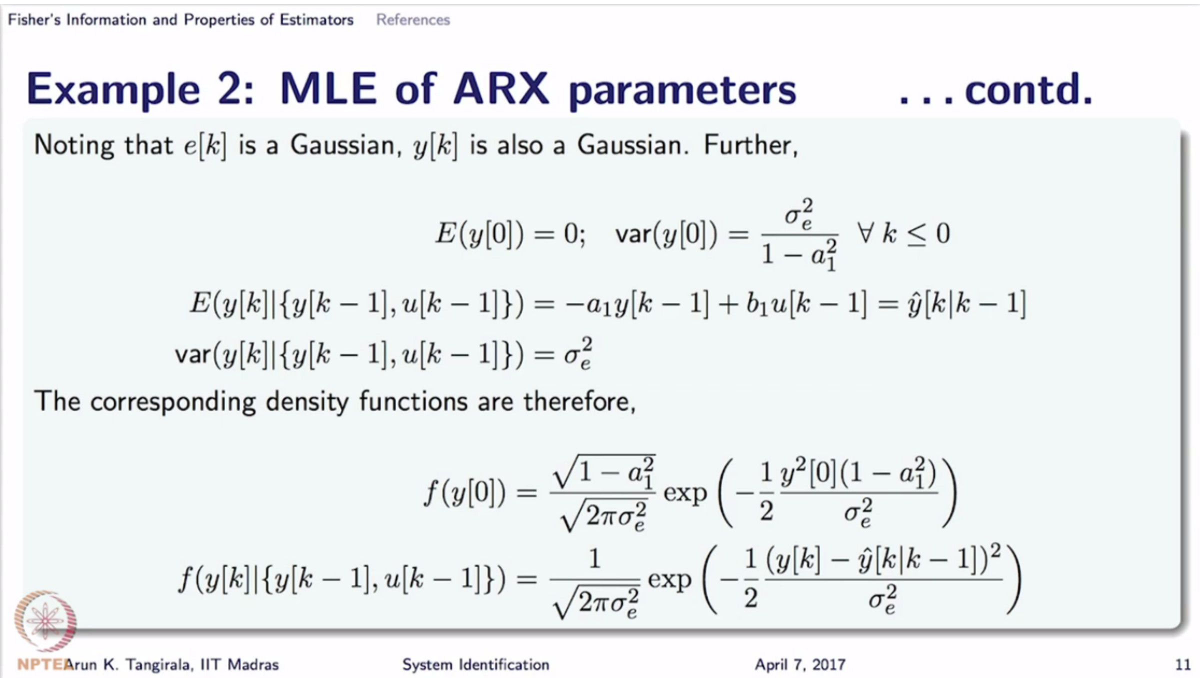
Now you see, since y1 and y0 are fixed, and u is known to be free of error, the only source of randomness in this is e2. And now you're write y3 given y1 y2, the only source of-- when I say source of randomness, the only random component there is going to be e3. Now you should understand the trick that we are working out. We are trying to get to the ideal residual, one step ahead prediction error, because the ideal one step ahead prediction errors are white noise. And we know that white noises are uncorrelated.

So what we're actually doing is, from the given series we are generating, we are somehow extracting the unpredictable component. We are constructing a new series, that is the trick here. The trick here is given the series with the help of the model, well, symbolically, we are extracting the uncorrelated part of each observation, so that we reach there. That uncorrelated part you can say is white noise, or you can say the ideal one step ahead prediction error. And you will see that, the prediction errors now quickly come into picture.

So if I were to say, overall you're given y0 y1 up to yn minus 1 from where you are actually retaining y0 and then for the ARX case, you are constructing a new series. You understand that? Those ideal -- what are the ideal epsilons? Theoretically, what are those epsilons? They are white noises. Optimal epsilons are white noise themselves. So you are constructing a new series with the help of the data and the given model, well, given model in the sense symbolically. Now this series here, conditioned on y0, the epsilons are all uncorrelated.

For a given y0 all epsilons are uncorrelated. So, epsilon 1 is uncorrelated is -- so, epsilon 1 is uncorrelated, epsilon 2 and so on. More or less, I mean, in the sense they are uncorrelated, but more or less this is what we are doing. What I meant by more or less is essentially generating a new series. When it comes to ARX second order, you would be generating epsilon not, you will be generating a series is y0, y1 and then epsilon 2 up to epsilon n minus 1. So, slowly this concept of prediction error is making its way through. That'll become more clear now. Let's go back to the ARX example, first order example. As we said, all I need now is this f of y0 and the general conditional PDF, f of y0 is straightforward.

(Refer Slide Time:17:09)



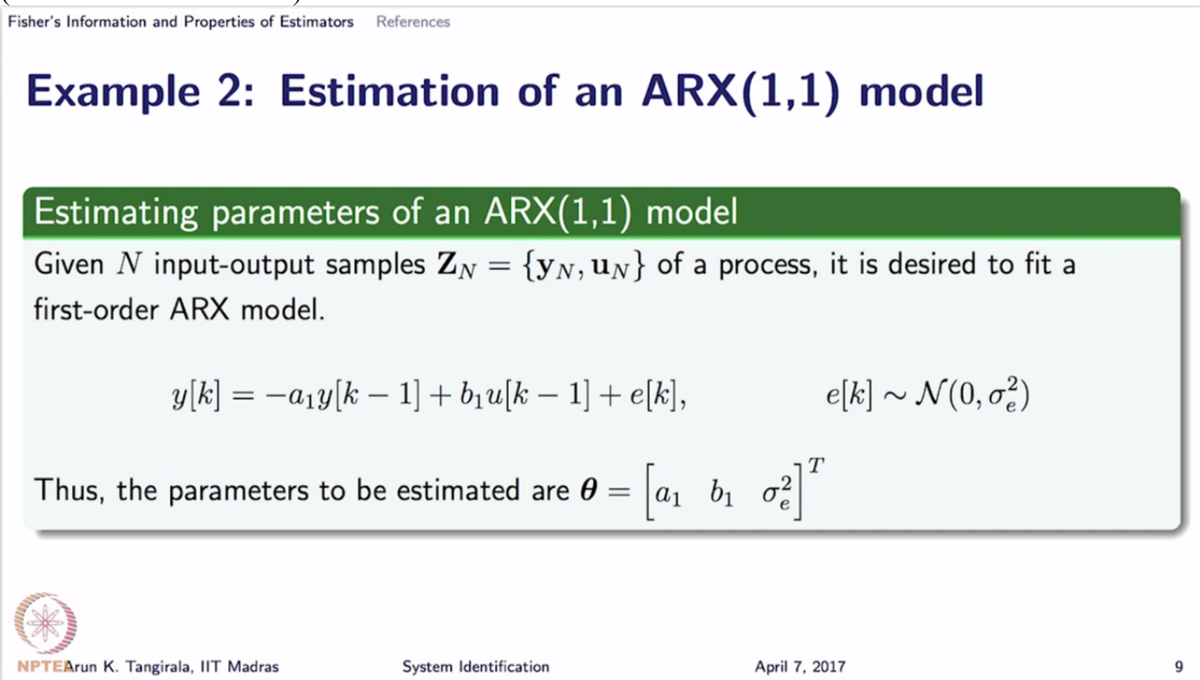
I know first of all that each observation comes out of a Gaussian distribution. That's clear, right? When you look at the model, it's pretty clear that if ek is Gaussian, is it so clear? What do you think?

How do you convince yourself that yk is at any instant, y0 or yk falls out of a Gaussian distribution? How do you convince yourself, given this model? And given that ek is white noise-- Gaussian white noise?

Very simple question here. I'm given this generating model for yk and I'm given that ek is Gaussian white. Now, I have to prove that yk also follows a Gaussian distribution.

How do you prove it? How do you prove it? Is it so difficult? What is the difficulty? Or is it too obvious that you're in so it's obviously follows it Gaussian distribution? Suppose there was no yk minus 1, would y, would it be very obvious that y follows a Gaussian? Yes or no? I don't hear yes from others. What happened?

(Refer Slide Time: 18:21)



[19:16 inaudible]

Yeah, so the difficulty is yk minus 1. How do you handle that? Sorry.

Previous data we already know.

What? We're not talking about conditional ones. Unconditionally will yk follow Gaussian? How do you do this? You can use a shift operator and what you can do is, you can express yk as a summation of past inputs and past white noises, right? So you can write this model as 1 plus a 1 q inverse operating on yk, as b1 uk minus 1 plus ek and then use a long division. And also assume that you're working with a stationary model, stable model, right. We are only working with stable models.

(Refer Slide Time 20:15)

Example 2: Estimation of an ARX(1,1) model

Estimating parameters of an ARX(1,1) model

Given N input-output samples $\mathbf{Z}_N = \{\mathbf{y}_N, \mathbf{u}_N\}$ of a process, it is desired to fit a first-order ARX model.

$$y[k] = -a_1 y[k-1] + b_1 u[k-1] + e[k], \qquad e[k] \sim \mathcal{N}(0, \sigma_e^2)$$

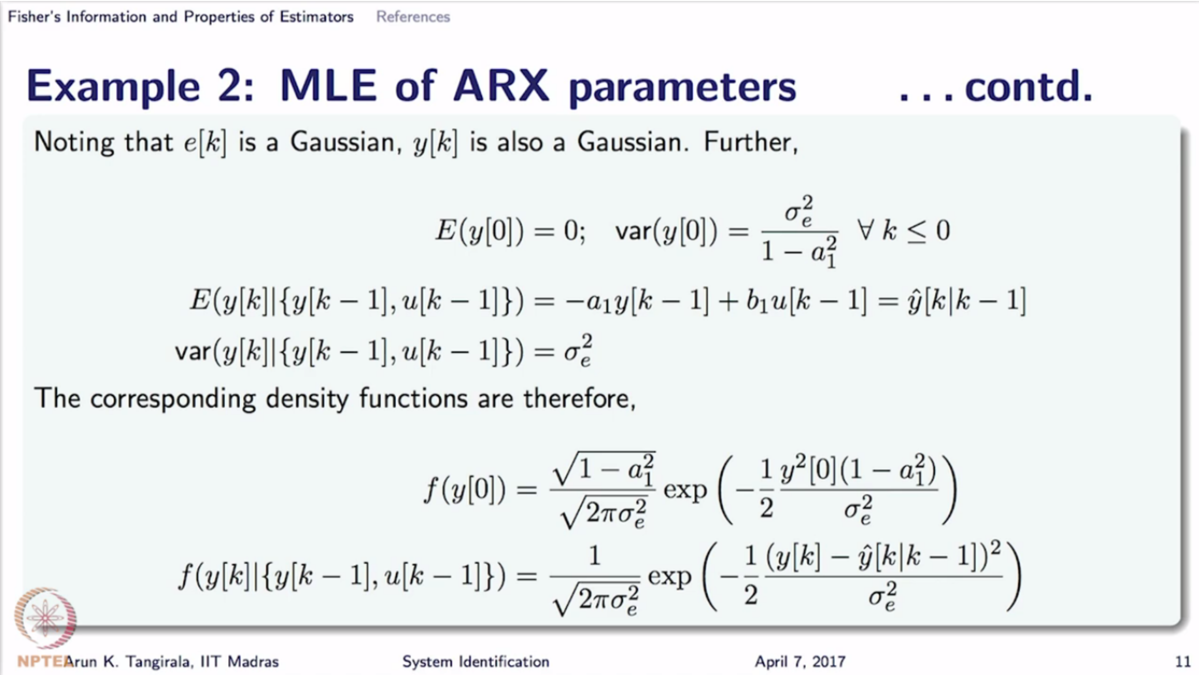
Thus, the parameters to be estimated are $\boldsymbol{\theta} = \begin{bmatrix} a_1 & b_1 & \sigma_e^2 \end{bmatrix}^T$

So then what would be the case? You have here 1 plus a 1 q inverse yk, let me write it here on the top. So from this, we know that 1, we can write this in terms of fifth operator, equals b 1 uk minus 1 plus ek. Now you can use a long division and express yk as an infinite summation. We have done this before, maybe you have forgotten in the case of auto regressive models. When we talked about stationary condition of AR models, we have done this. Then what happens? So now yk is going to be expressed purely as a sum of present or the delayed inputs uk minus 1, uk minus 2 and so on up to infinity. And then ek, ek minus 1, ek minus 2 up to minus infinity. Will that help you see whether yk is Gaussian or not? How?

Inputs are deterministic. They are going to only cause mean shift. Right. Good. So that's it. So that will, that should help you see that y0 or yk for that matter follows a Gaussian white noise distribution. Clear now? You should think, I mean, see at no point you should just blank out like that. Think as to what are the different ways in which I can write this equation, so that I can, first identify the difficulty. As I said, if yk minus 1 were not to be there, then it's so obvious that yk follow a Gaussian.

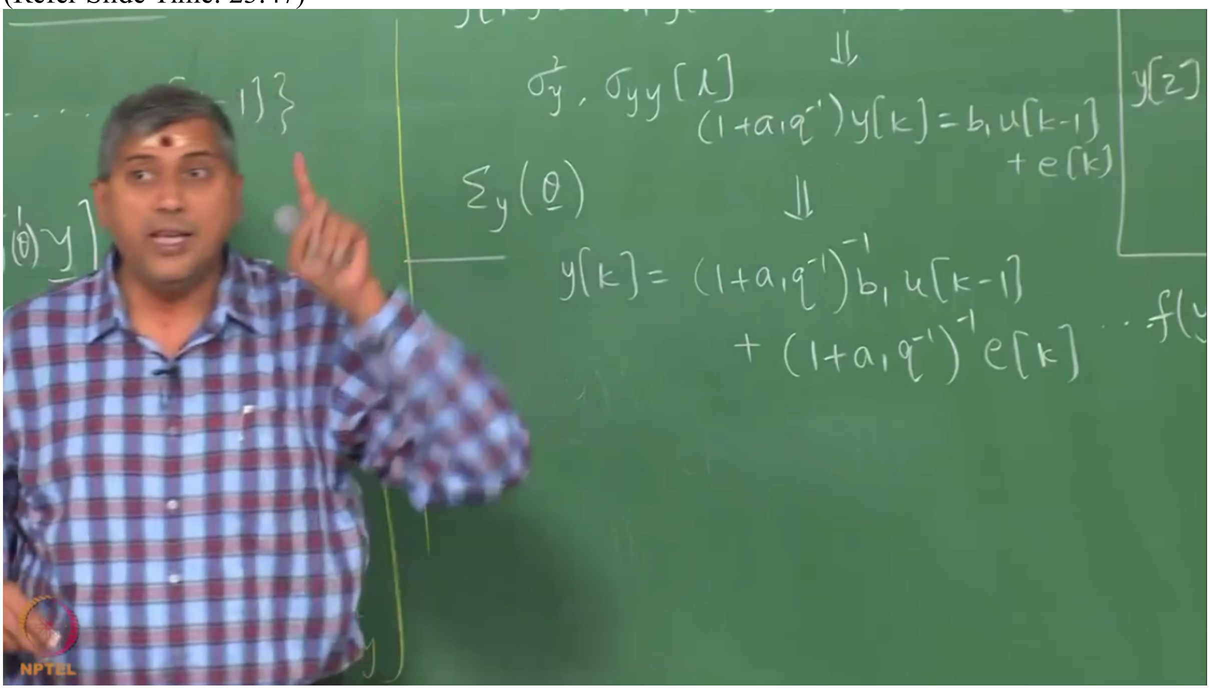
So it is a yk minus 1 that is really causing the impediment in your thinking. Now you ask, how do I handle yk minus 1? This is one of the ways in which you can. Anyway, so now that we are convinced that yk follows a Gaussian distribution, you write here, y0 expectation of y0, we know is 0. In fact, once you write that, you can clearly see that the mean of y0. Assuming input to be, well, strictly speaking, if you look at the mean, although I write here, expectation of y0 to be 0, in this case, I'm assuming that-- what constitutes y0, so like, we have to be very careful.

(Refer Slide Time: 22:59)



When you write this in general, what would be expectation of yk? I'm going to erase this. Right? So if I write from here, I write yk as 1 plus a 1q inverse, inverse of that, b 1 uk minus 1 plus 1 plus a 1q inverse, inverse operating on ek. From this the expectation of yk in general is going to be this part. That's clear, because this ek is zero mean. But for y0 alone, you will, I'm assuming to be zero mean, strictly speaking, it is zero?

(Refer Slide Time: 23:47)

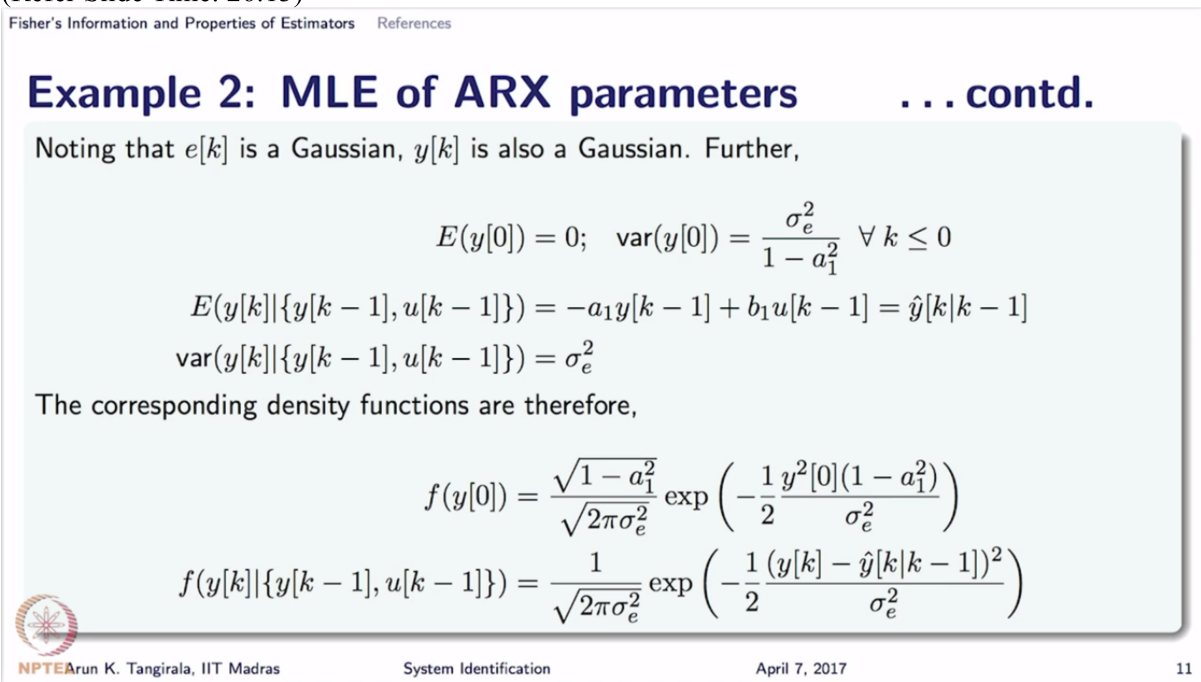


Why have I assumed expectation of y0 to be 0? As I said expectation of yk in general, at any k the observation is going to be this portion here. I'm giving you the big hint. We assume input at negative times to be zero. That means when I say input here, again, you have to keep telling yourself, this input is not absolute input. It's a deviation from steady state. We assumed that previously that is, before I began my experiment, the system was at steady state. Now, many a times that may not be true. So in which case you cannot right expectation of y0 to be 0, you have to take the initial conditions into account.

So assuming input to be zero at negative times, the expectation of y0 is 0. What can you say about the variance? Now, you'll have to work out the variance. I'm not going to prove here exactly, but you should be able to see that variance, unconditional variance of yk, unconditional variance, that means just variance of yk. What is going to govern the variance part? This part is only going to result in a mean shift, whether this is present or not, the variance of my yk is unaffected because variance is a central measure, right?

So practically, you can ignore this part here, when you are computing variance of yk. So this is only one that's going to cause contribute to the variance. That you should be able to easily see. Yeah, because they are uncorrelated, it'll be sigma square e, it will be times the infinite series, right? 1 minus a1 plus a1 square, and so on. Assuming a1 to be less than 1, that means you're working with stable models, that series would converge. And that's how you get sigma square e over1 minus a1 square. So everywhere we are being very precise, we are not making any hand waving things here. That's it.

(Refer Slide Time: 26:13)



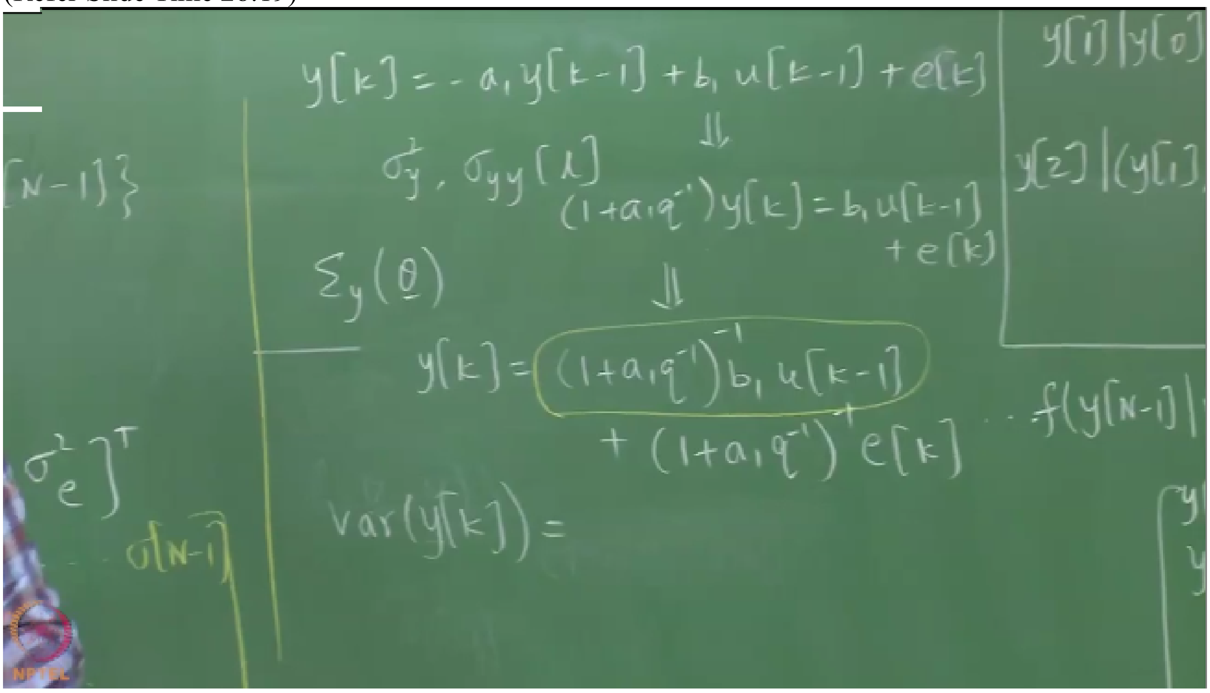
So I have figured out that yk is Gaussian, y0 is -- and that it's mean is 0, variance sigma square e over1 minus a1 square. Do you see now the parameters of the model are entering the likelihood function?

The parameters of the given model are a1, b1 and sigma square e. So, already they have made their way through. Now, the next part that is remaining-- by the way, so, this is your f of yk, right? Do you see that? This is simply, your Gaussian PDF. Now what remains is f of yk given yk minus 1.

Again first you have to ascertain the yk given yk minus 1 follows a Gaussian distribution. What kind of distribution does it follow is what you have to check. So, if you look at, again, going back to the model, if I fix yk minus 1, what distribution does yk follow? Gaussian, it's very obvious. This is easy to answer. The earlier ones, there was a lot of silence in the class. This is very easy. The moment I freeze yk minus 1, straight away it's possible to see that given yk minus 1 follows a Gaussian distribution, because both yk minus 1 and uk minus 1 are now only going to cause a mean shift. Correct?

So that's --now all I have to do is figure out what is the expectation of yk given yk minus 1 and uk minus 1 and variance. That's very obvious, minus a-- because ek is of zero mean, the mean now is minus a1 yk minus 1 plus b1 uk minus 1. Which is nothing but your one step ahead prediction. If I were to give you this model and ask you, what is one step ahead -- oops, nobody corrected me on this.

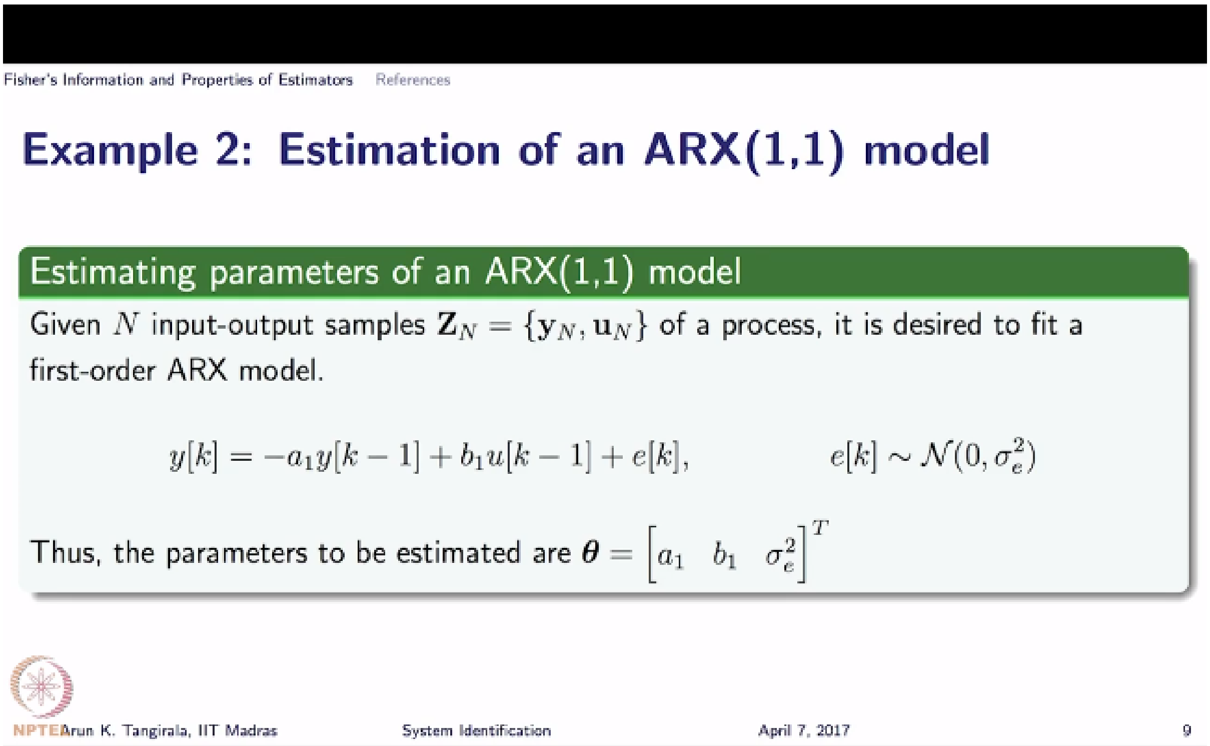
(Refer Slide Time 28:19)



You should correct me. So if I were to give you this model and ask you what is one step ahead prediction, the one step ahead prediction would be this part, right? So the mean conditional expectation-- we are coming back to the same point, the conditional expectation is the optimal one step ahead prediction, and that is what is coming out here. And what about variance?

How do you prove that the conditional variance is sigma square e?

(Refer Slide Time 29:03)



Sorry? Correct. So, this two terms are only going to cause a mean shift, yk given yk minus 1. This is different from unconditional variance. Do you see the difference? In the unconditional case, the variance is going to be different. In unconditional case, what was a variance? Sigma square e over 1 minus a1 square. But the variance of the conditional yk is sigma square e. Now, does this make sense, can you explain?

By the way which is lower and which is higher? Is a variance of the conditional one higher or the unconditional one? Unconditional one is higher because a1 is less than 1. What does it mean? What does it tell you? Why should the variance of the condition yk be lower?

Sir, in conditioning we are fixing random path -- y of k minus one.

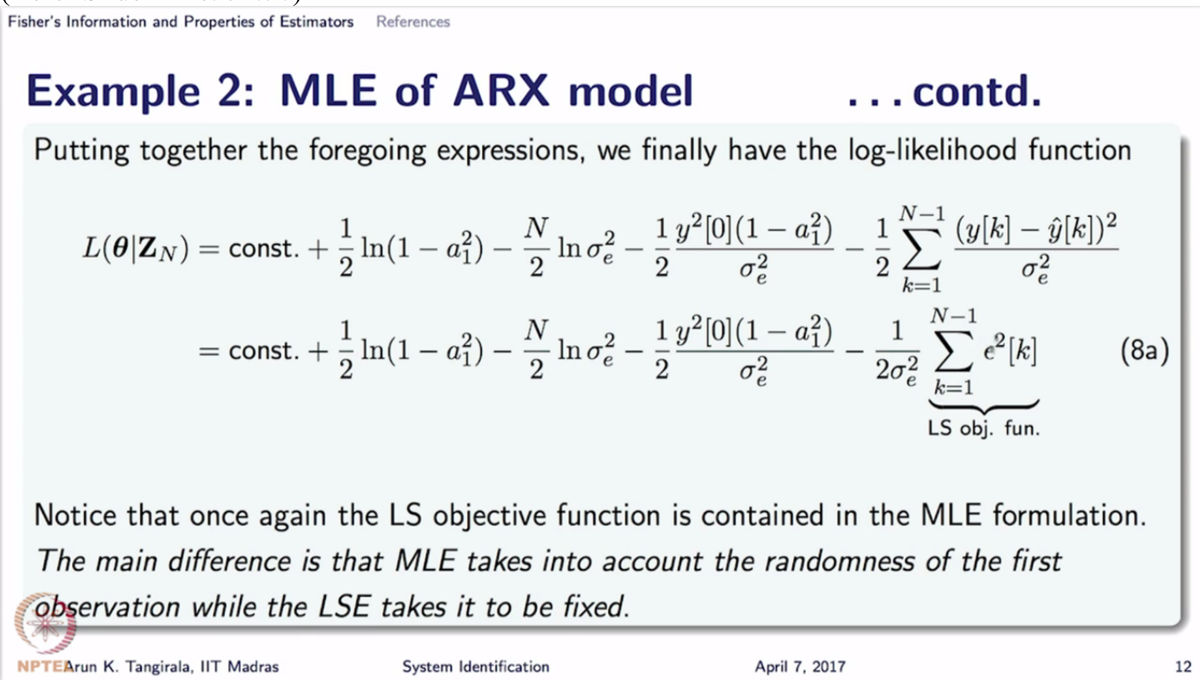
Okay. Okay, that's one way of explaining, good. So you're fixing one randomness, you're freezing the variability in one of the contributors, thereby it comes down. That's a good observation. The other perspective that you should develop is, we have just seen that the moment I condition-- that I'm evaluating conditional expectations, and so on, what I'm doing is I'm making a prediction. When I make a prediction, I've already taken into account hopefully, wall that was predictable and what I should be left out with the least variable component of yk.

Right, that means the uncertainty in unconditional one, which I don't make, where I am not making any prediction, just the observation, that is going to be larger than the conditional one, because the conditional one is more or less like a prediction. So whatever is left out will be the residual. Always it is the case that, if you have chosen the right variables to condition, the variance of the conditional one is going to be less than the unconditional one. Look at it this way, in the unconditional one; I'm not giving you any information. And I'm asking you, what is the variability? What is the randomness? Versus conditional one, I'm giving you additional information. I'm giving you yk minus 1.

In unconditioned one, I'm only asking what is the variance of yk? I'm not giving you any previous observations at all. So your uncertainty is going to be higher in yk, whereas in the conditional one I'm giving you that this is what happened in the past, so your uncertainty should shrink if there is a correlation, if there is a predictability, and in this case, we assume there is a predictability. Therefore, the variance shrinks that means, now you have more knowledge of the process. So, that is an important point to observe.

Anyway, so, now we have found out again the expectation and the variance. We know that yk given yk minus 1follows a Gaussian, now, we put together everything. The rest of it is now algebra. So, this is the log-likelihood function. And observe something very interesting here. If you look at this equation here, the second one, we have, of course, a constant term will ignore, we have one, two, three, four terms, you can say so, or you can club a few of these terms together. But just observe the last term here. This is nothing but your least squares objective function. Of course, you have 1 over 2 sigma square e, but that doesn't matter.

(Refer Slide Time: 32:58)



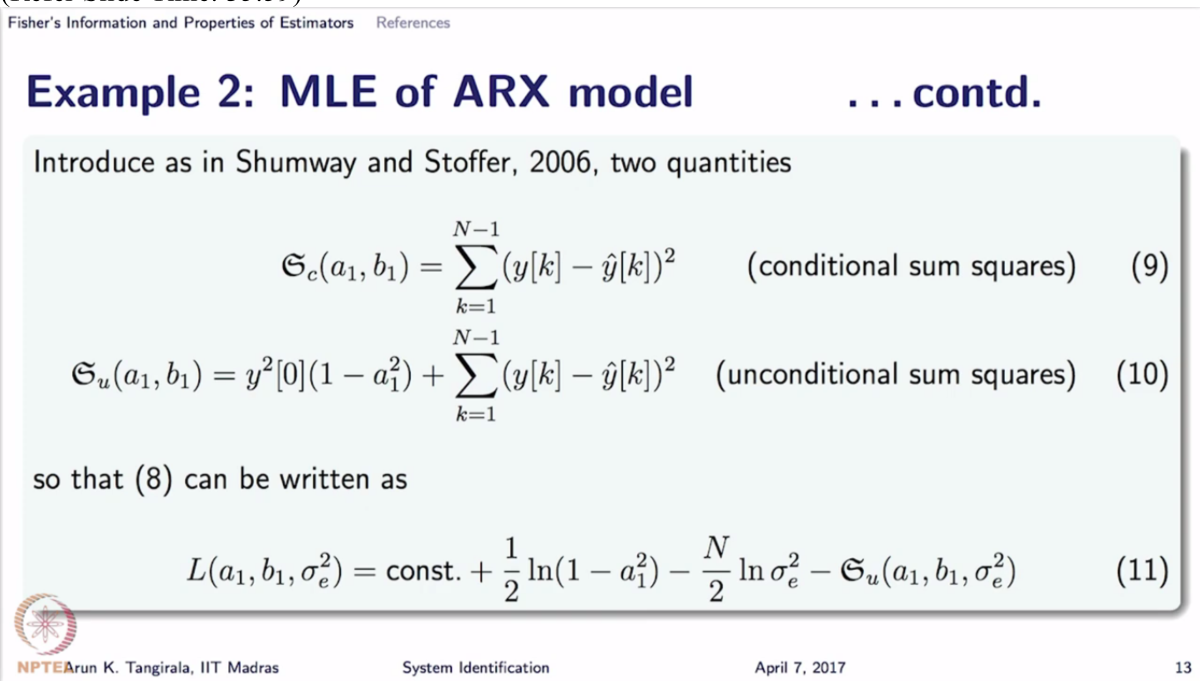
In the least squares objective function, whether I have a one over two sigma square e or not, the solution is going to be the same, right? That is the first observation that the least squares objective function is a part of your MLE. That means MLE is more than the least squares. Again, I explained this towards the end of yesterday's lecture, that in the least squares-- when you set up the objective function, what do you do? For this model, you ignore y0, ignore meaning, although I say ignore, you are conditioning everything on that. You're starting your predictions from y1. Here also, you're starting a prediction from y1 but what is the difference?

The difference is that you take into account the randomness in y0. In least squares, you just assume y0 is fixed. That's one way of looking at it. Because you have taken into account the randomness in y0, there are these additional terms appearing in the MLE, right? That is a first observation. Second observation is that although we started with MLE, we ended up with an objective function involving minimization of some square production errors. So somehow whether you explicitly state it or not, the prediction errors are making their way through. In the least squares we were very explicit, right from step one; we said we want to minimize the sum squared error predictions.

But in MLE we never said that. We said that will maximize the log-likelihood. But somehow it turned out that the prediction errors are making their way through. And this is the beauty, which means the unifying concept for least squares, MLE and in fact, even Bayesian to certain extent, is this concept of minimizing some function of prediction errors. And that is what is a crux of prediction error methods. That is why the prediction error methods are universal; they actually bring in least squares, MLE everything.

Now at this point having observed the difference between least squares and MLE, we'll introduce to summations, okay? One we call it as --so, remember here this summation runs from 1 to n minus 1, not 0 to n minus 1, correct? Why is that? Because we'll start making predictions only from one onwards. So, introduce this sum here which we call as conditional sum squares, and the second sum which we call as unconditional sum squares, what is the difference between these two? In the second sum, we are taking into account something else, right?

(Refer Slide Time: 35:59)



What does this conditional sum squares consist of? Epsilon 1, epsilon 2 up to epsilon n minus 1. This is the series that we are considering for constructing the conditional sum squares. We call that as s subscript c, okay? It's a function of a1 and b1. In fact, the second one should be a function of a1 b1 and – here, a1 b1 and sigma square e. You didn't have breakfast? Volume should be higher. The mic is unable to pick up such low volumes. We'll come to the technology soon. Okay.

So what is the difference here, in the conditional sum squares we are only looking at this series and we are saying we want to minimize the sum square of this. Whereas, in the unconditional sum squares, we are additionally considering epsilon 0 and the rest of this. You may say where epsilon 0 is, I don't see that. Do you see that epsilon 0 appearing in unconditional sum? See now we have developed a perspective. MLE is like the super dada, it takes into account all the prediction errors from 1 to n minus 1 and randomness in y 0.

Least squares is you can say lazy squares, that means it doesn't worry about randomness in y 0, it simply says freeze y 0, simply look at the prediction from 1 to n minus 1, minimizes sum square prediction error. There is a middle ground also now. The middle ground is, where I do not take into account the full randomness in y 0, but I only worry about the variance of epsilon 0 or y 0. Remember when we look at-- if I were to ask you, what is your prediction of y 0, what is it?

If I don't fix y 0, you're not giving anything. What is your prediction, standing at time zero? Zero because you don't have previous input, the expectation of e k is 0, right? I'm not given y at minus 1, so the only prediction that I can make, well, there are other ways also, but the simplest prediction that I can make is zero. Which means what is epsilon 0? y hat of 0 is 0. Epsilon 0 would be? y 0, right? Epsilon 0 is nothing but y 0 itself. Although I have written on the board epsilon o, now you see why y 0 has appeared in unconditioned sum squares. What is the logic behind constructing the unconditioned sum squares? Weighted least squares.

Suppose I were to ask you to minimize sum square prediction errors of this series. Sum square terms of this rather than this, this is what least squares looks at which is what conditional sum squares is, you know, born from. But suppose I asked you to work with this and I say minimize the sum square of this, you should take a weighted least square approach, why? Why am I saying that you should take a weighted least squares approach? Any quick answer? When do we take a weighted least squares approach?

When the errors are different. When the variance is different, variability is different for different observations. Now, the point is these ones have a variance equal to sigma square e. We already proved that. Why? Because those epsilon 1's epsilon 2's they're all conditioned ones, right? They're constructed by given yk minus 1, I mean, y1, y2, y3 and so on. We already proved that the variance is sigma square e. And we have also proved that epsilon 0 is y0, I mean, we have kind of concluded that epsilon 0 is y0.

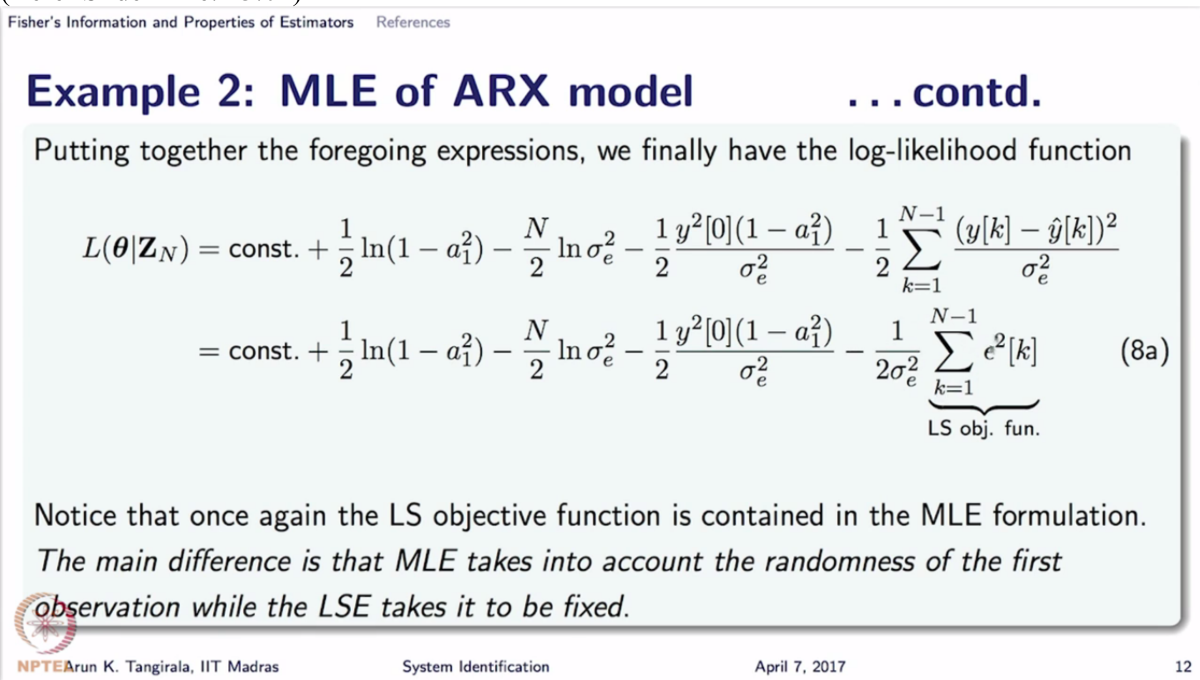
What is a variance of epsilon 0 therefore? We have the answer, sigma square e over 1 minus a1 square. Correct? So variance of epsilon 0 alone is different. So the variance of this is sigma square e over 1 minus a1 square, correct? Now, what is the weighted least square, say if I want to now setup a weighted least square kind of problem for these prediction errors, the bottom one, what would be the optimal weighting? Inverse or the variance.

So that is what we have done exactly here. Except that sigma square e doesn't appear because that's common to both. Ideally, I should have had here y square 0 times 1 minus a1 square by sigma square e, right? Plus 1 over sigma square e. But that doesn't make any difference at all. Therefore, you have this unconditional sum squares. This is the middle ground, this is unconditional, why do we call it unconditional? Because we are not conditioning this on y0, we have taken into account the variability of y0. But why do I say this is middle ground between least squares and MLE?

The reason is, in least squares I don't even bother about the randomness in y0. In MLE, I'm fully bothered about it. That means, I straight away take the PDF into account. But in this minimization here of the unconditional sum squares, I am bothered about y0 but only about it second moment. I'm not looking at the PDF. If I were to look at the PDF, then I will go back to the MLE, this would be the objective function. You see that.

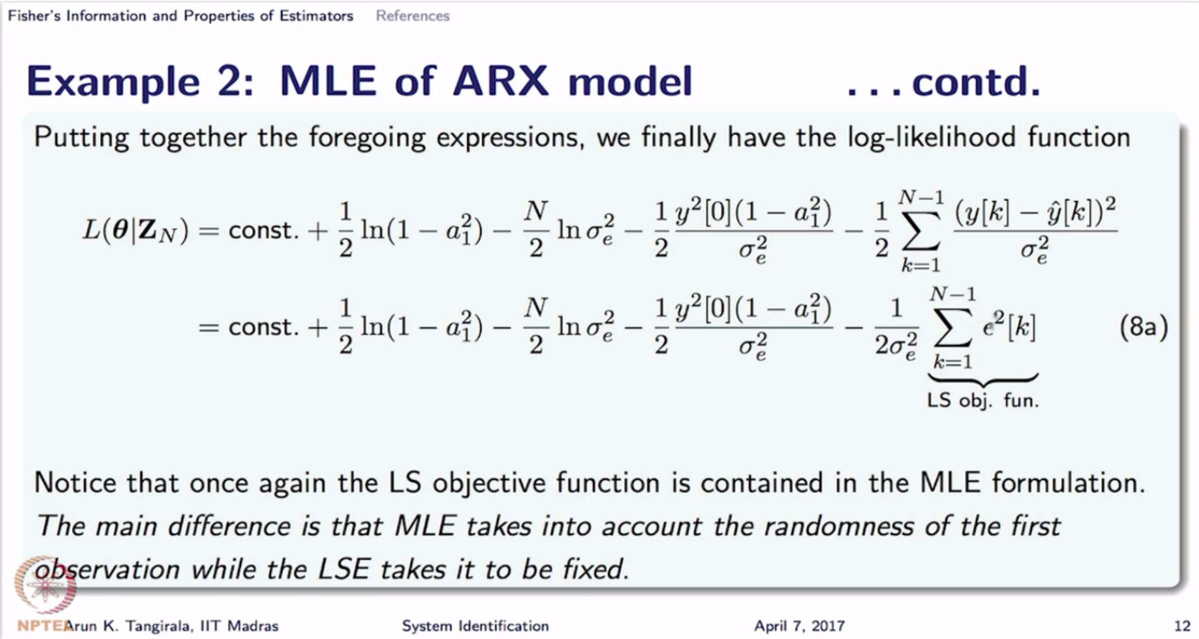
So you see, now hierarchically the first term here is least squares. If I take these two into account, what do I get? Unconditional sum squares. We don't call as weighted least squares. And then if I take these two terms here, I'm looking at MLE, the full exact likelihood. So that is how you have hierarchically always in any MLE problem, you will have three hierarchies. CSS called the conditional sum squares, then the unconditional sum squares and then the MLE. If the full MLE is difficult to be solved, you turn to unconditional sum squares. In this case, it's very easy to solve. But in many other problems, likelihood can be very complicated.

(Refer Slide Time: 43:02)



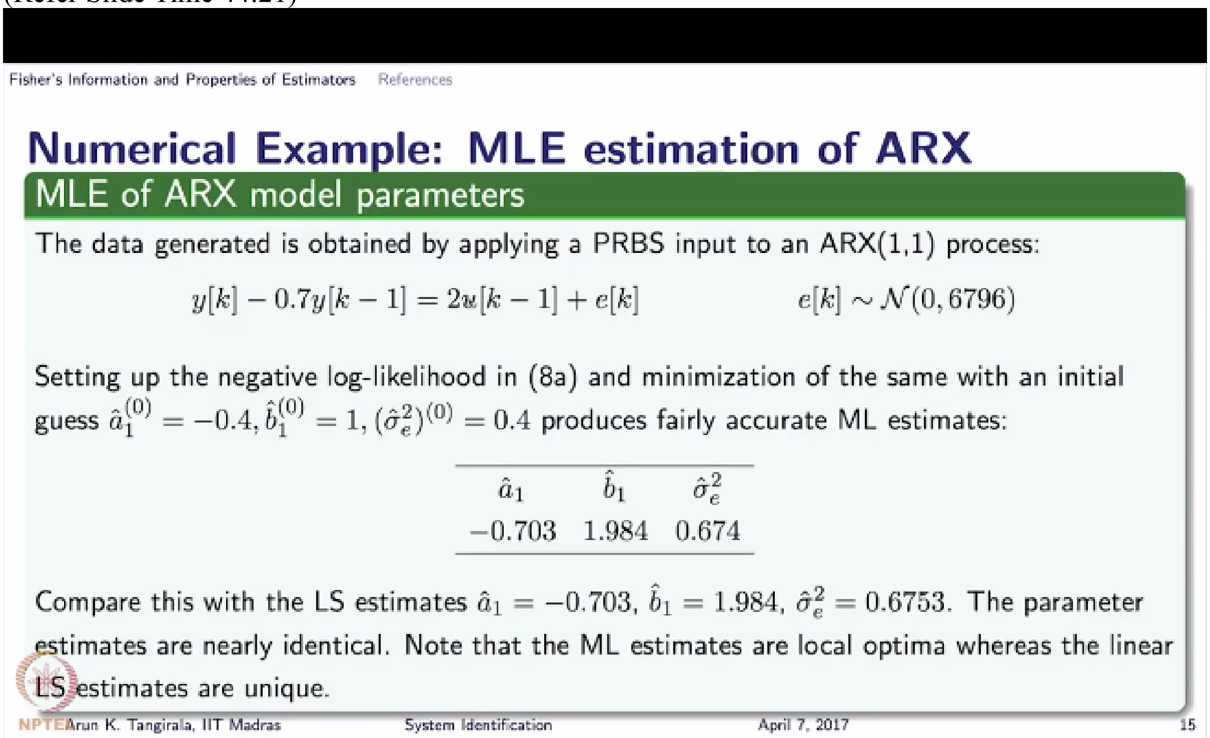
So you say that full likelihood is very difficult to maximize log-likelihood, I'm going to work with select terms in the likelihood. The discussion that we have had until now is to give you the interpretation of what it means to work with selected terms and likelihood. So if you were to-- in this case, if you were to work with only these two here, the last two terms you will be working with unconditional sum squares, that means you're going to look at this prediction errors. But even if that is difficult, then you work with the only the last term which is the least squares. And naturally you should expect the solutions to be lesser and lesser optimal, as you start sacrificing the terms with respect to the full likelihood.

(Refer Slide Time: 43:57)



So that's it. That's all I wanted to say. So that is now your likelihood, right? And unfortunately, there is no analytical solution to this. You will have to use a numerical solver. Once you get your parameter estimates, you can straight away estimate your sigma square e. So it is possible to first estimate a and b if you want, and then get your sigma square e. Let me just quickly show you a numerical example.

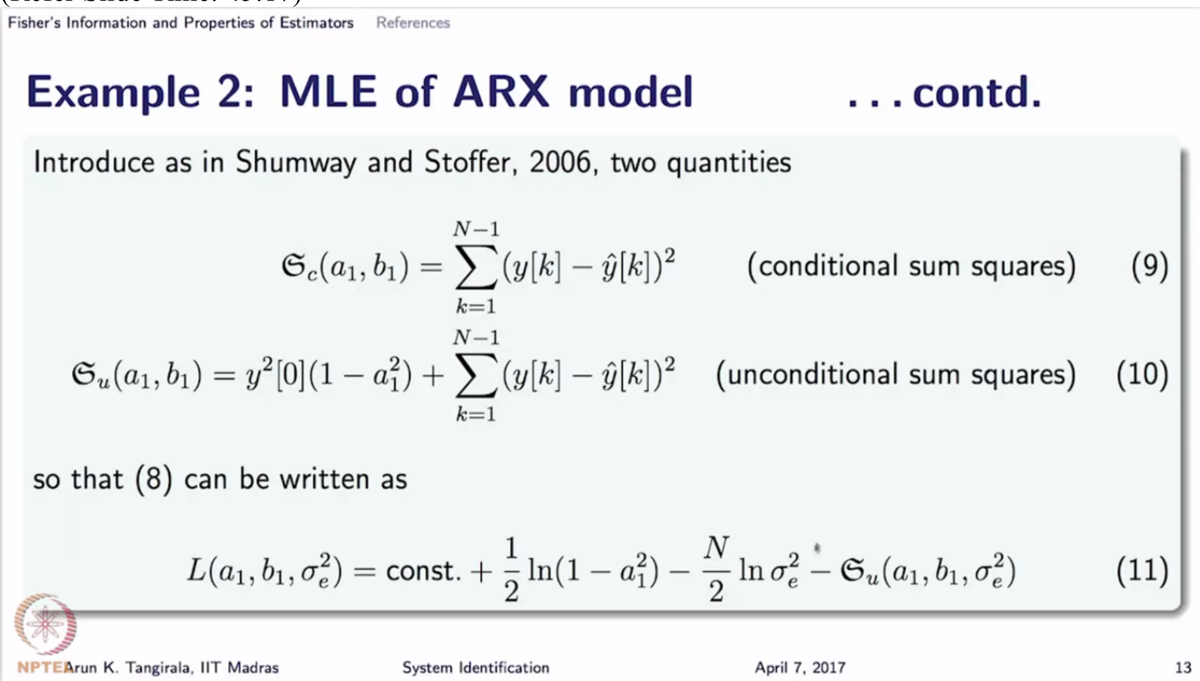
(Refer Slide Time 44:21)



This is the process that I've used for generating the data. And I have, by the way, as I said; the MATLAB scripts for this example are available on my website. You can go and download. You should do that because you should know how to write the likelihood function. There is no MLE or anything like that in MATLAB by default, because likelihoods keep changing with the problem. So your job is to write always for a given problem, parameter estimation problem, write a function that computes log-likelihood and that should be passed to an optimizer.

So the optimization routines are available in MATLAB. Your job is to only write a function and which I have written already for this problem, look at the script. It's a very simple one. It just takes into account this expression here, that's all. Or you can say this expression, whichever, doesn't matter, that’s the same.

(Refer Slide Time: 45:17)



That's it. So I have written a function that computes a log-likelihood and that's passed to the optimizer and you need an initial guests, by the way, because your MLE problem is a nonlinear optimization problem. You need an initial guest, and there is a whole lot of literature on water. Good initial guesses for MLE. In this case, actually MLE is an overkill. Why, why do I say it's an overkill? I can simply use least squares. Why is that? It's a linear predictor. So I can use a linear least square, I can get analytical solution. Whereas, with MLE I'll get a locally numerical optimum, local optimum only.

I am not guaranteed global optimum. But just to illustrate the idea and so many other points that we have learned that although you work with MLE, there is a prediction error that comes into play, that least squares is contained in MLE, then weighted least squares concept is also contained in MLE. All of these points, we could easily understand through this example. So typically to estimate ARX models or AR models, nobody uses MLE. But it's a very good example for educational purposes.

The actual power, full power of MLE comes into play when you are dealing with ARMAX models or Box–Jenkins models and so on. And then there is a moving average component, it becomes a bit tricky, but already people have worked out ways of setting up the likelihood for those cases as well. So this kind of concludes the illustration of MLE on the estimation of parameters. I'm not going to show you how to set up the MLE for ARMAX and BJ and so on.

I've given that in the textbook. But as long as you understand that setting up the MLE involves conditioning, constructing a conditioned series or constructing what is in the literature you will come across what is known as an innovations algorithm. Your innovations are nothing but you're white noise sequences or the ideal one step ahead prediction errors. So, there is an innovation algorithm, general innovations algorithm that bypasses the exact likelihood and rather sets up the likelihood in a, in a simpler way, which is used for estimating ARMAX models and all other model structures, right?

But we will not get into that in this course. As long as you understand that there is a likelihood that you have to set up, you have to respect the correlation and so on, and all the other points that we have discussed that should be more than enough for now. So your OE algorithm, BJ mod routine that you have in MATLAB, ARMAX routine and so on. They are actually using MLE. But for large samples, whether you use MLE or nonlinear least squares, more or less, you'll get the same results, for large observations.

So although I said just now MLE, what lies underneath ARMAX, BJ and OE is a nonlinear least squares optimizer, which uses a Gauss Newton Algorithm. So tomorrow when we come back, we'll now put together everything and look at estimation of non-parametric and parametric models and then, you know, in the next lectures we'll look at only two topics Input Design and State-Space Identification. Okay? In process, I'll go through a case study as well. Thank you.