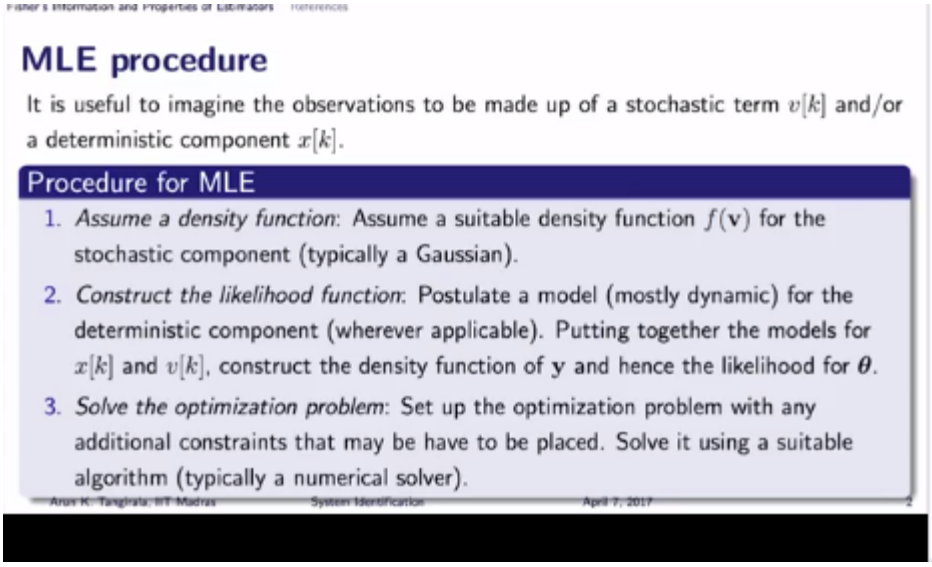
CH5230: System Identification

Fisher's information and properties of estimators

Part 14

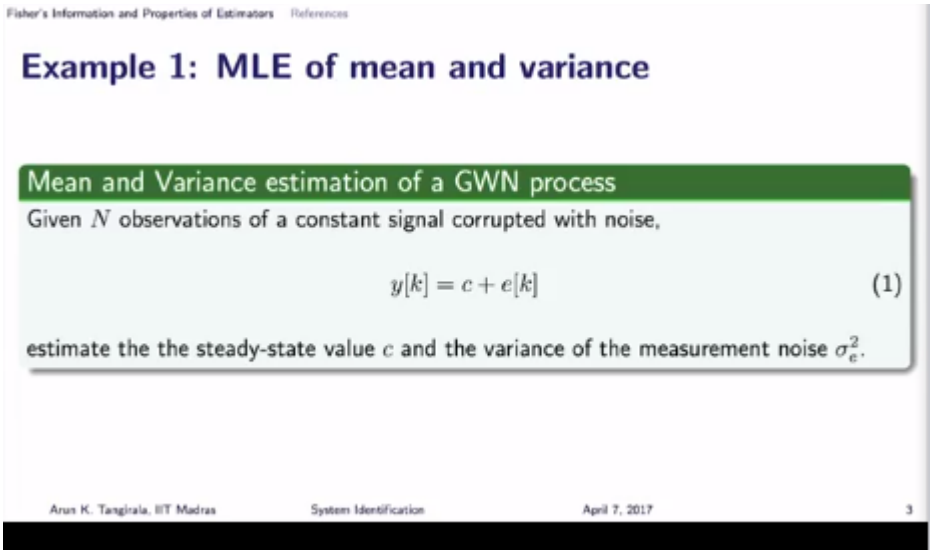
So let me just quickly go through MLE. We've already talked about MLE. Again, the lecture notes must have given you the notion of MLE and I have talked about likelihood also early on. What was the first time you were introduced to likelihood? Fisher's information. Right? And I said, the maximum likelihood principle is simply based on maximization of the likelihood. And the only job the user has to do is set up the likelihood. Now having said that, that is the challenging part. The optimization part is done by the solver, where you can sit back and relax and there's computer's job to do it.

(Refer Slide Time: 00:54)



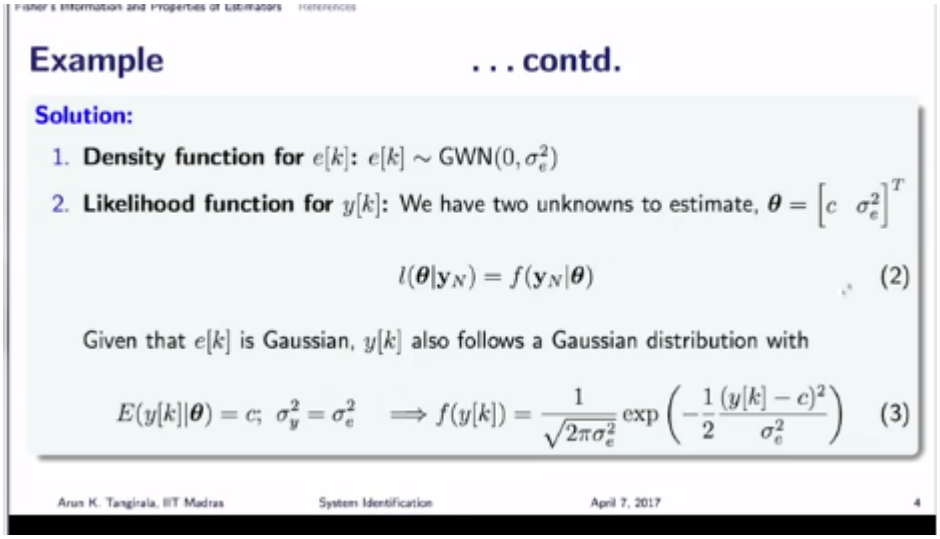
So the typical procedure to set up the to, you know, come up with MLE to solve an MLE problem consists of four steps. Again, I've talked about the steps also in the lectures earlier. Just to recap, the first you identify. Sorry, you assume density function or the probability density function for the source of randomness, and then, construct the likelihood function. In fact, there is a baby step in between this, you will have to figure out how the randomness propagates to randomness in the source of your error. So for example, here, the source of randomness will be e[k]. So if this is the model that you're going to fit for example, you will have to figure out or even here, suppose this is the model that you're estimating, let us say. This is the model or this is the model. In fact, I'm going to talk about this example. If this is the model that you're fitting, we have learned how to do the least squares way. How do we do the least squares way? The model is the same. If I use least squares method, how do I go about estimating a1 and b1? Set up the regressor, and then simply use the least quest solution, right? And in fact, for that, this example, do we run into linear least squares or nonlinear least squares? You have to be quick. Linear least squares. Those who are meditating, I'll ask later on. Okay. Do you understand? Some of you are meditating, I'll ask you later on the answer. Those who are awake, you can actually answer. It's a linear least squares problem. So you set up the regressor and then you estimate the parameters. In MLE, the way you estimate is, you begin from here. You begin from here, you assume a density function for this and then you ask how the randomness in e[k] propagates to randomness in y[k]. And to set up the likelihood function remember, I have to set up a joint density function of n observations. And that is where the challenges is. In the simple case, in this example that I've gone through in the lecture, you must have seen how to set up the likelihood.

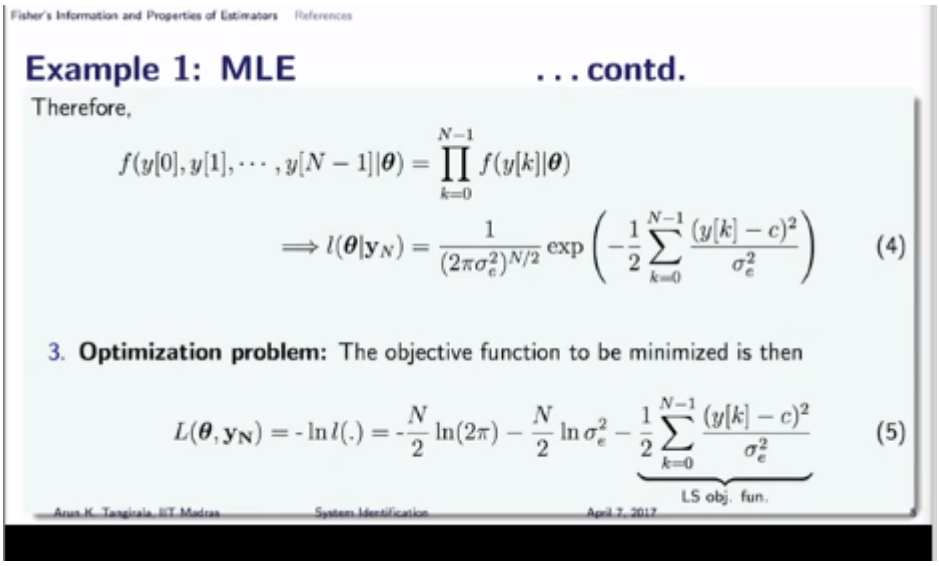
(Refer Slide Time: 03:19)



We haven't already done this for the Fisher's information case. This is a very simple case, why? Because whatever is the density function for e[k] that is y, except that, y is being shifted. And what can you say about the joint p.d.f. of the observations, are these observations, if e[k] is assumed to be Gaussian white, what can you say about the n observations? They are independent, right. Therefore the, writing the likelihood is very easy. All you have to do is simply write the joint likelihood as a product of the marginal p.d.fs and the marginal p.d.f. is a Gaussian with this variance and mean being c. That is easy.

(Refer Slide Time: 04:03)





But now what is a challenge? So I'm going to go pass this, we have already done this now for this ARX case what is the challenge that you see in setting up the likelihood? Still assuming e[k] to be Gaussian white. What's the challenge that you see? We have talked about this earlier also.

Y[k] not be able to [4:32 inaudible] .

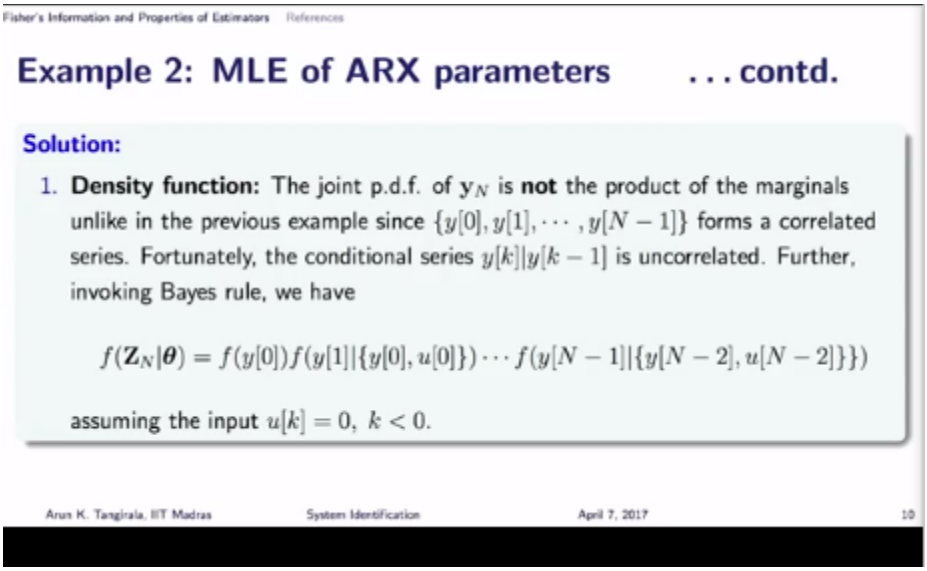
In fact, they are correlated. Because y[k] is correlated is y[k] minus 1 and in fact correlated indirectly with y[k] minus 2 and so on. So, I cannot write likelihood as or the joint p.d.f. as a product of marginal p.d.fs. So have I reached a block that I cannot overcome? No. Now here is where we play a small trick and we just use a slightly very nice trick to overcome this, and then you will see the intuition behind it. The intuition, so the idea here is, suppose, I look at not the marginal p.d.f., but the conditional p.d.f. In other words, let me put it and then I'll explain the intuition behind it. Suppose, I like ask you, we know now for sure that y[ks] are all correlated that we all agree, right? That's very easy to see because I'm given, I'm assuming such a model, therefore, there's no doubt about it. We all agree that y[ks] are correlated, correct? But what if I ask you this? So there is y[1] given y[0]. So it's conditioned random variable. And I also ask you, y[2] so these are two random variables that I'm considering. y[1] given y[0]. I'll tell you why we are looking at this. Suppose I take, just take these two random variables. Instead of, if I look at y[1] and y[2] are they correlated? Are these two correlated?

Yes.

No doubt. These are two new random variables, which, this is a condition random variable now. That means I'm fixing y[0]. Now, I'm not looking at the free random variable, unconditional random variable, y[1]. I'm looking at conditional random variable. Y[1] given y[0], y[2] given y[1]. Are these two new random variables correlated? Remember, whenever I condition a random variable on some other random variable, like suppose I condition x on z, then for all working purposes z is deterministic. Because I fix z. How do I read? I say, y[1] given y[0]. Whenever I say I'm given I'm anchoring if the entire analysis around fixed y[0]. So I'm fixing y[0] now freezing y[0] and I'm looking at y[1]. That is what is y[1] given y[0]. Then I freeze another random variable, I look at y[2] that means all realizations of y[2] for a fixed value of y[1]. You understand? The top one that I've written y[1] and y[2] are all possible realizations of y[1] and y[2] that that I generate by simulating this process again and again. You agree? Even I simulate I'll get another set of y[1] and y[2]. So all possible values of y[1] and y[2] are what we mean by unconditional random variables. But I do a

different kind of simulation. I fix by y[1] and I generate many, many, I fix y[0] and y[2] like zero. And I generate all possibilities for y[1]. Do you think that this random variable is different or these realizations are going to be different from unconditioned y[1]? They will be, correct. Likewise, I will generate now, I'll y[1] and generate y[2]. Now, I want you to tell me when I consider these new, two new random variables, they are different random variables now. Are they correlated? Sorry. Sorry, then you can see from the equation right, y[1] given y[0]. What is it? Minus a1, y[0] plus b1, u[0] plus e[o]. I've just use, right now we're assuming that I have the correct model. Don't worry about the model plant mismatch and swap. Assume that I know the model structure, don't complicate life more than what it is already. So likewise, y[2] given y[1], minus a1, y[1], plus b1, u1, plus e[1]. Now what do you think? Are they correlated? So that means I have to, this is some random variable, this is another random variable. Are these two correlated? Remember, in evaluating, in answering your question, you have to remember that why is deterministic now. Because why it's been fixed. Y[0] is fixed. What about u[0]? That's also deterministic in a, when I generate different realizations of y[1] fixing y[0] is the u[0] going to be the same or not? Although, I don't say that, u[0] is assumed to be deterministic. You understand? Which means, these two are going to be some fixed numbers? What is going to generate different realization of y[1]? e[0]. Did I make a mistake here? Sorry, e[1], yeah, you should have corrected me. Okay, thank you. I'm thanking myself. Okay. So what is responsible for different generation of y[1] given y[0]? E[1] only. What is generating different realizations of y[2] given y[1]? e[2]. Because the rest terms are fixed, remaining terms. Now, you answer the question whether these two random variables are correlated or not? Simple, finished. You may say what is the connection? What are not that you're talking about? Why are you not talking about these condition variables? The trick is this, when I consider the joint p.d.f. of, let's say two observations, I can use base rule and write this as f of y 0 times f of y1, given y 0. Right? Now I extend this argument to three observations. What do I get? So when I write this for three observations. That's all, if you understood that then the rest is only math, which I'll go through tomorrow. Fairly simple. Now what do I right here? F of y(0) given, in fact, what I should do is, I have to have baby steps here. I should have f of y(0) given y(1), oh, sorry. Let me right the baby steps, so that becomes clear. First I consider these as a partitioning, so that I write this as a f(0) y(0), y(1) times what? F(y2) given these two. Y(0) and y(1). Sorry, I'm going to erase this. Now, according to the model, this is the interpretation that you need to remember. When I have a model like this, what is the model telling me y(2) is only dependent on y(1). Not on y(0). It is indirectly dependent, but not directly. Whenever you have an auto regressive model, let us of order one, you will see in many statistics textbooks, this interpretation would be given. The interpretation is that, of an Ar1 model is f of y[k] given any amount of past is the same as a f of y[k] given the immediate past. That means I don't need y[0] at all to know y[2]. I just need y[1]. If that is the case then it is Ar1 process. On the other hand, if I had an Arx second order, then I need both observations. That means, they say conditionally, they're independent. That is, y[2] given y[1] I don't need y[0] at all. The information of y[0] is embedded in y[1]. Whatever information is required to predict y[2] that's embedded. On the other hand, if I had a second order here, then I would need also by y[0] additionally to complete the prediction. You understand? So therefore, I can throw away y[0] here, because it doesn't matter to me. Whether you give me or not, it doesn't matter. The p.d.f. is not going to change. What about this? I'm going to use this, right? So now if I extend the idea where do I get? The joint p.d.f. of n observations is f of y[0] times the of all conditional periods.

(Refer Slide Time: 14: 27:)



Which condition p.d.fs? F of y[k] given y[k] minus 1. Very simple. That is another way of saying that the conditional observations, conditioned observations are independent. In fact, we are saying uncorrelated, but it also independent, because it's a Gaussian white noise that we're dealing with. So that is the basic idea. Now all I have to do is basically find out. So let me complete expression here. I have f of y[0] times f of y[1] given y[0] and f of y[2] given y[1]. That's all. I'm going to erase this as well. So that you don't confuse. Now all I need to know is this f of y[0] and in general, F of y[k] given y[k] minus 1. There two p.d.fs I need to construct and put them together. F of y[0] is an unconditional p.d.f, p.d.f of y[0] as it. The rest are all conditional p.d.fs. Now, we'll do the math tomorrow, but I leave you with a thought. When we did, I asked you earlier how do you set up the least squares problem for this. And your answer was, yes, I'll set up the regressor vector. What is the size of the regressive vector? What is the size of the regressor vector? Vector and then I'll ask you about phi. What about the regressive vector? What will be the size? Two gross one. What about phi? Suppose I give you 1,000 observations? Minus 1mark, 2 by 1,000. Another minus 1. Think, we have gone through this discussion. You have to be careful when you construct phi for dynamic cases. How do you say 1,000? Do I have?

[16:40 inaudible] minus.

So, it's 999 by 2. Phi is going to be 999 by 2. Why is it that it is 999 by 2? What are we actually doing? We are starting our prediction from 1 onwards. Because and nowhere does y[0] to appear in your phi?. It appears, but in the regressor, so what, how does the regressive matrix look like? You're aggressive matrix consists of y[0] and up to y[998], right, assuming that our index runs from zero to n[1]. But does your y vector, if you look at y, does it run from 0 or 1? 1 and goes up to 999, but here the y vector, even the first observation is included. So the difference between least squares and MLE is that MLE recognizes, yes, that y[0] is required to make a prediction but does not leave behind y[0]. It takes into account the fact that y is zero is a random variable and your entire thing is condition is a condition on that. So in other words, you can think of least squares as a conditional MLE, where you throw out y[0] your entire parameter estimation in least squares is conditioned on y[0]. Right? It is condition. You're leaving out y[0] completely, when it comes to the y vector. But MLE does not do that. MLE actually also takes into account y[0] and the fact that it is a random variable. And that's why you have the f of y[0]. Even the first observation is a random variable, it has some randomness in it. And that is why MLE is powerful. Already, in the previous example, if you recall the lecture, where you look at MLE of mean, the least squares is automatically contained in MLE. Least squares is a

special case of MLE for the mean, with the mean estimation example, I've already made this observation. MLE with a Gaussian distribution is the same as least squares. Here also you will see least squares, but some additional terms. So tomorrow when we come, the first 15, when we assemble for the class, the first 10-15 minutes we'll spend on this MLE, I'll show you a MATLAB code. The script is there, show you some numerical example and then we'll move on to estimation of parametric and non-parametric models. Tomorrow is a lovely Friday timetable. Okay.