Advanced Numerical Analysis
Prof. Sachin Patwardhan
Department of Chemical Engineering
Indian Institute of Technology – Bombay

Lecture - 24
Model Parameter Estimation using Gauss-Newton Method

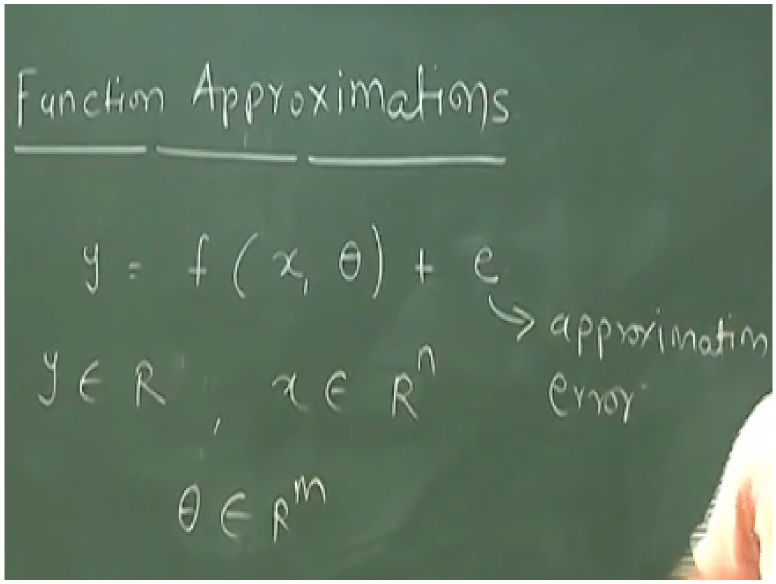
So pre-mid-sem, we were looking at problem discretization. And towards the end that how to discretize a boundary value problem using method of least squares and then more or to the Gelarkin method. And I suppose you have learnt more about Gelarkin method in the mid-sem. You know more about the Gelarkin method. Now before I move onto, now tools, I just want to finish of one topic that is least square estimation.

So one more piece remaining as far as least square estimation is concerned, we talked about model building using lease squares, method of least squares. And I gave several examples though through while talking about linear squares though I did not formally introduce to model building. I am going to do that little bit today and then I am going to discuss a method called Gauss-Newton method.

A Gauss-Newton method is a cross between Newton's method or so called Newton–Raphson method and least square estimation. So this method is used when the parameters in the model appear non-linearly. And there is no transformation by which you can linearize the non-linear model. If you have such a model then you cannot use linear least squares, you have to use something else and that is Gauss-Newton method.

So today I am going to spend time on a little bit on model building first then move onto Gauss-Newton method. So this function approximation in engineering is very, very common and that is why I want to concentrate on.

(Refer Slide Time: 02:37)

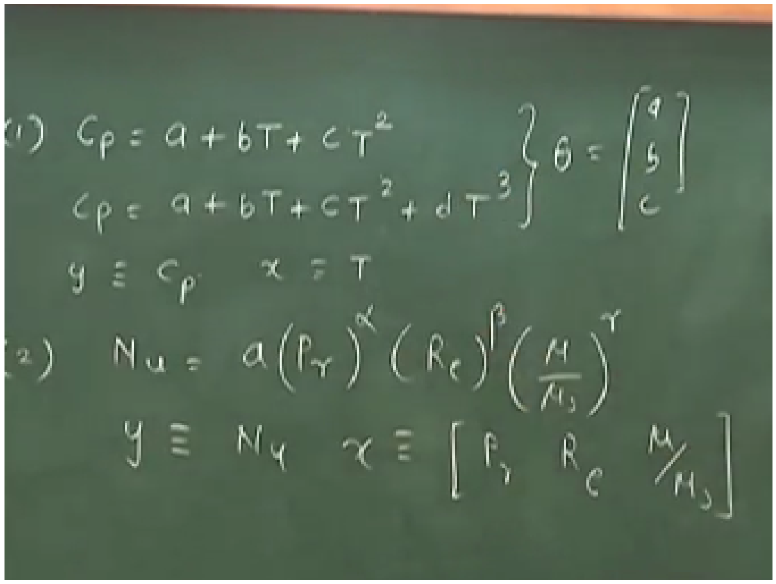


So I am not just restricting myself to problem of approximation I will just talk about function of approximations in general. And here we are talking about, we are talking about a function why is some function of a vector X and some parameters theta + error. We are talking about a model of this form where F is some function. Okay. Y here is typically a scalar value. X can be N dimensional vector in general and theta is an m dimensional vector.

E is the approximation of error. Well, when I am writing this model I am already making some assumptions and I want to free you from that particular assumption right now to begin with. In the begin let us now worry about this error, where the error appears. We will look at the error little more systematically. So y is some dependent variable, x is a vector of independent variables. Theta R the parameters in the model.

And then I want to fit a model of this form. Okay. I want to fit a model of this form. And I have given you examples.

(Refer Slide Time: 04:20)



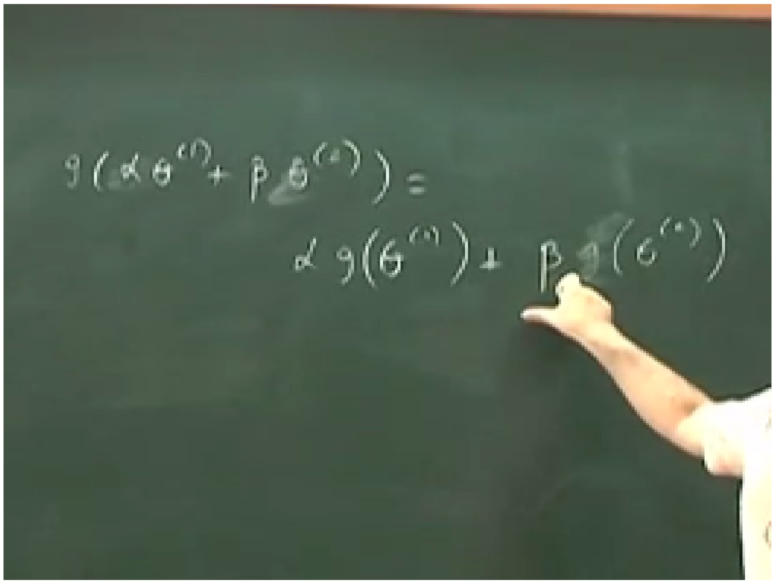
First example was Cp versus C square or Cp = a+bT+CT square+dT cube, okay. In this particular example, we have y corresponding to Cp and x corresponding to T. I gave you second example was, now set number equal to. You have correlations like this, right. And in this case y corresponds to Nusselt number and x corresponds to Parental number or Reynolds number and Mu/Mu s. Okay. X corresponds to these three variables. Parental, Reynolds and Mu/Mu s.

Whereas y corresponds to Nusselt number. They are correlations of Nusselt number, you have correlation for number, mass transfer. Now the nice thing about these two particular models, in models in engineering I am talking about function approximations because here you are using exponential function, here you are using polynomial function. Okay. Using exponential function for approximation, you are using polynomial function for approximation.

Different kinds of functions you are using for approximation. Okay. Polynomial is the simplest one but not always use polynomial, use all kinds of other functions. Okay. And then the problem is given data of Cp and temperature you want to estimate a, b, c abcd or in this case given data of the set number, parental number, Reynolds number and Mu/Mu s estimate alpha, beta gamma, right. And what was nice about these two models was you could transform them to linear m parameter form. This is already in linear m parameter form. Isn’t it?

The model parameters appear linear, reduce there something in linear. You have to remember the basic function, any function G of x.

(Refer Slide Time: 06:45)

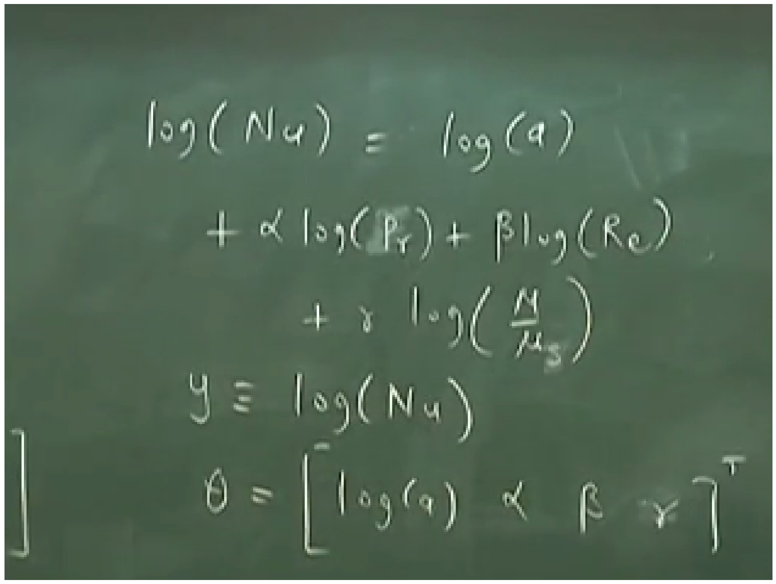


So g of theta. Okay. If you can write alpha theta 1 + beta theta 2 = alpha G of theta 1 + beta g of theta 2. When I saying that this particular model is linear in parameter, what is the parameter vector here? Theta is a b c. What is the parameter vector here a alpha beta gamma. Okay. So when I saying that this particular model is linear in parameter which means this definition holds with respect to theta. Okay.

If I take two theta, theta 1, theta 2 okay, I can write F of theta, G of theta will be in this G of theta will be a+bT+CT square. Okay. If I take two theta parameters a, b, c and a prime, b prime, c prime then I can write as addition of two functions. Okay. Which means a, b, c here appears in the linear manner. If you try to apply this definition to the second function it will not hold. Okay. Because if you—this is not in linear parameter function, right the parameters are alpha, beta, gamma obviously they are not appearing in the linear manner, okay.

So but a saving raise here was I could do a transformation. Okay. So I could actually make a transformation which make the model look like, look like a linear in parameter model

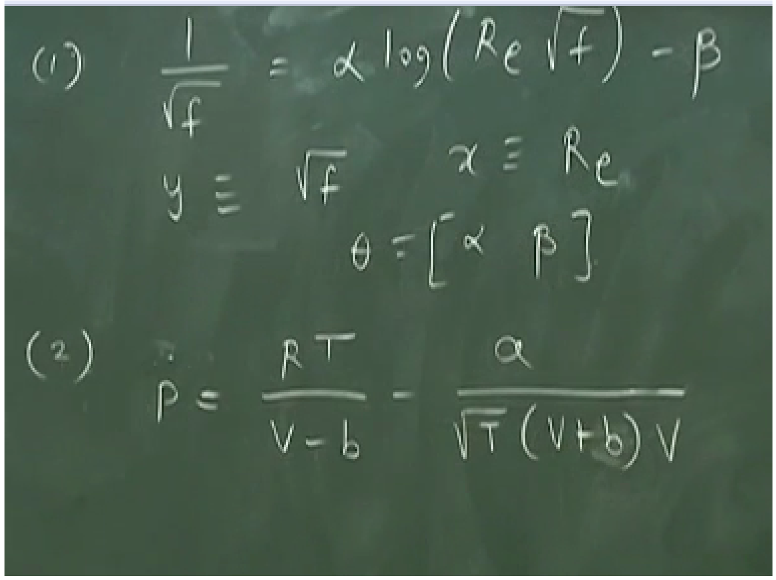
(Refer Slide Time: 08:36)



And that transformation was log of Nu was log a + alpha log Pr + beta log Reynolds number + gamma log Mu/Mu s. Okay. So with respect to this transform model, when I said the y was y corresponding to log of Nu theta correspondent to log a alpha, beta, gamma. When my theta was this, with respect to these parameters these parameters and a transform model okay, we have to linear in parameter model and we could use linear least squares. Okay.

We could use linear least squares. But the problem of estimated in model parameters appears in many, many engineering problems where the model is not transformable, I will give you simple examples from what you know in Chemical engineering. So let us take two examples which you cannot do any linearizing transformation.

(Refer Slide time: 10:04)



So one of this is the friction factor. Just remember, all these are correlations, function approximations, these are not many time derive from some fundamental physics, there is some fundamental physics of course when you write dimensional group. But beyond that when you find out those alpha, beta, gamma, okay those are correlations. So I would call them as semi empirical models because part of it actually comes from physics.

The dimensional groups arrive from—you arrive at a dimensional group from physics, okay. But then the correlations coefficients are from data. Okay. So we would call such models as a gray box model. Okay. So this is the friction factor correlation, so this is alpha log okay. Now this is a very, very funny thing because I want to estimate alpha beta, my y here let us say if I call y as you know, root f, if I call y as root f okay and if I call x as Reynolds number, okay.

I want to fit this correlation between friction factor and Reynolds number and my theta here is alpha and beta okay. If you are given a data of friction factor versus Reynolds number okay. And if you want to find out alpha, beta for this particular model there is no linearizing transformation. Because y appears here. You see here. Because y appears here and its very funny. Okay. So you cannot do this in a very simple way, you have to think about something else.

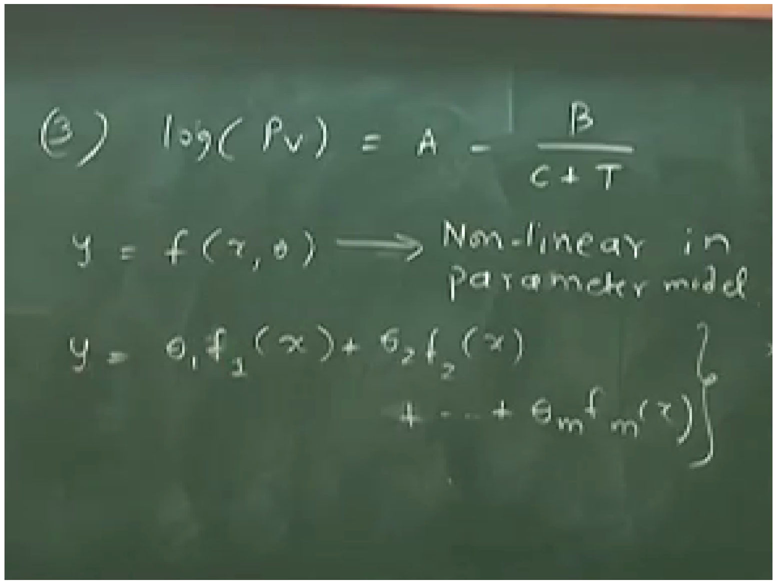
Well, the other correlations which are very, very familiar with are say Redlich–Kwong equation, say p = RT – V-b – a upon root T V+, I hope I am correct. So this is the Redlich–Kwong equation, you do not know a and b. okay. You will be given data of P V T okay. I want to fit, R of course you know, I want to fit I want to find A B values okay and estimate this from data. I just have P V T data for a particular substance I want to find out A B, okay.

So this is a classic problem in thermodynamic. You want to find out A and B by least squares. Now mind you when you say that this is equal to this okay when you write this that this is equal to this, this is equal to this, these are all approximate correlations. In reality they are not equal. Okay, in reality they are not equal. Actually the correct way of writing you should say that estimate of p = this, that is what you should write. Okay, it is not true p.

But nevertheless we while using, in engineering we do not really get into these niceties of statistics, we just assume that this is the exact correlation and then we will use we do not worry about the errors in the approximation and so on. And when we take care of errors in approximation by doing some kind of over design so that you know, you modeling errors are somehow covered. Okay.

So when you do all these, when you use heat transfer coefficient estimated using the correlation Nusselt number or when you use mass transfer coefficient, all these are approximations, they are not exact. And then that is the reason why we do some kind of over design and try to get over the problem. So this—now here the problem is there, how do you estimate the model parameters. Well, I would like to still have my paradigm of least square estimation in there. And I want to get a least square estimate of theta nevertheless.

(Refer Slide Time: 14:53)

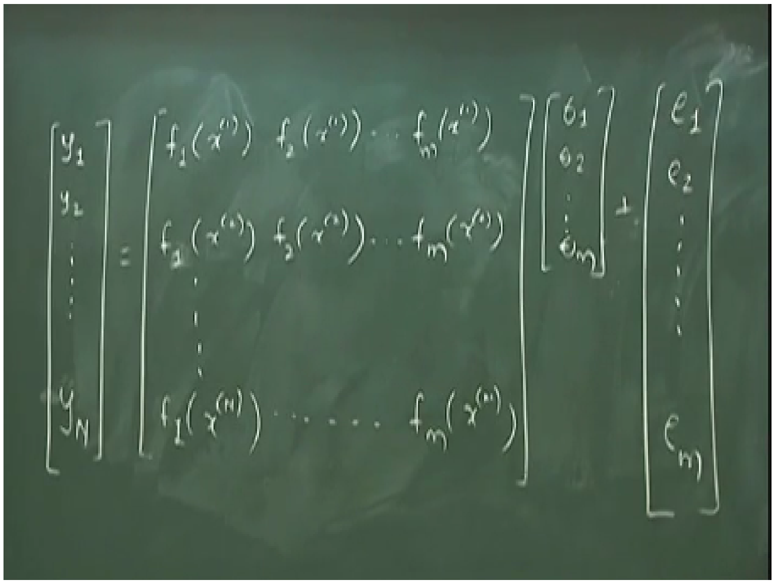


But you know the third example is. What is this correlation? (()) (15:01) correlation. So (()) (15:03) correlation again is not exact. For a particular substance you go to Perry's handbook and find out A, B, C okay, these are fitted coefficients. This is not a true relationship, this is a, this is a correlation this is a function approximation. Okay. That correlates vapour pressure with temperature. Okay.

So in all these models you have a problem. So I would classify the models in two classes, one is I have a model which is, I have this model y = f of x theta, in some cases if theta appears in linear model, if the theta is appearing in non-linear way in the model, this is the only way I can write. If theta is somehow by some transformation or something is appearing in linear model, then in that case you will get y = theta 1.

See this is linear in parameter model, this is general non-linear in parameter model. Okay. And this is linear in parameter model. Okay. When you had a linear in parameter problem, you had a nice way of estimating theta 1 to theta m least square estimate. What was the method? When we collected the data for linear in parameter models--

(Refer Slide Time: 16:47)



We could write this from the data that is y1, y2 okay. We could write it as a linear equation. Large number of equations, less number of parameters and we could solve this like this up to y capital N lets say. F 1, we have collected N data points, right. We wrote like this. We said that we have large number of equations. And we assigned error in approximation only to error in the equation. We assigned error in approximation to error in equation.

What if your x itself is wrong? Temperature measurements. I am collecting temperature measurements from a plant okay and my temperature measurements have error in measurement errors when I collect temperature measurements. I never get exact value of what is there in the plant. What is there in the system I get always an error picked up because of varieties of reasons because, it could be if I have electronic system which is collecting data there could be noise entering my data because of varieties of reason some fluctuation in the because circuit, some loose contact. Okay.

So temperate inside let us say, my system is 100 degrees, I might get 101 I might get 199. I know that boiling point of water I am dipping a thermometer at you know in Bombay and boiling water it should be 100, I do not get 100, I get 100.5, 100.2 in fact if you collect the data in a computer you see that it all fluctuating around 100, you will never get 100. Actually you expect 100, right. It should be exactly 100.

So when I am developing a model of this form okay. My x is T but my temperature itself could be wrong. Okay. Well, there could be errors in the estimation of Pv. Okay. So there are actually three kinds of errors, three kinds of errors. One is Errors in the measurement. Okay. Well, in this one lecture I cannot answer how to deal with all three of them. It is very, very complex business. How to separate these errors and get a correct model; would probably require almost half a course to deal with all of them.

The simplest one is the so called equation error. So we have three possible sources of error. Your x here, your Y measurement could be wrong. Okay. Your x here inputs to the model. They could be wrongly measured. Okay. P V T data I am collecting. Pressure sensor can give me wrong data slightly wrong data. Volume estimates could be wrong. Temperature measurements could be wrong.

So now-- see when you propose a model, you see the trouble when you are developing a model, when you propose a model you are saying that true pressure = true volume * true whatever P V T relationship. You say that it holds for the true volume and true pressure and true, but when you measure there are errors. You do not know what is the truth and you take 100 measurements, each one of them have an error. Okay. So how do you arrive at the correct?

So it is very statistically or in terms of statistics system it is not so easy problem to solve. Well, these are the two possible sources of errors in the measurement itself. The third is that your expression which you are fitting itself is an approximation, right. T P versus temperature. I may choose to fit a line or I may choose to fit a parabola or I may choose to cubic equation depending upon the range I am covering okay and the fit I get you know, is there is a temperature range is large I may want to fit the cubic equation.

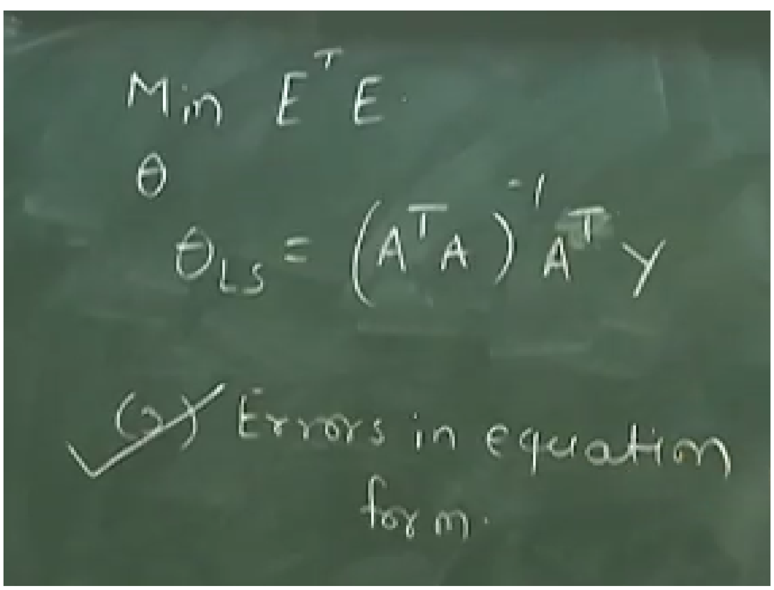
If temperature range is small linear equation might do. Okay. So there is a when you propose a model because these are approximate correlations. The error committed when you do that and there are errors committed in the measurements. Right now, I am going to blame everything to equation error. I am not going to worry too much about these errors. Okay, I am not going too much worry about errors in the measurement. Let us say we try to remove the errors in the measurement by taking repeated measurements and taking averages.

And we try to do some kind of compensation for, let us assume for the time being because dealing with modeling problem when you blame everything to equation error is much easier than dealing with the modeling problem when you want to consider all the three errors. In the notes I have discussed briefly about what is done there in all the three errors. But solving that problem can be become fairly, fairly complex. Okay.

Now I just want to do one thing now, my aim right now is that if I have a non-linear in parameter model okay. Can I somehow extend what I could do here, what you could do here? You could use method of least square, how was the method of least square, so I call this vector as capital Y let us say, this my tricks was my A matrix, this is my theta matrix, okay and this is my error vector, this is my error vector.

And then I had very nice way of estimating or generating the least square estimates and we looked that its variety of interpretations from geometry view point and from –we arrived at the results from the algebra and so on.

(Refer Slide Time: 23:22)



So minimum of with respect to theta Error transpose Error, this we could find analytically and we got theta least square = A transpose A inverse A transpose Y. Everyone with me on this? We got this least square estimates, okay. That is because; this particular model was linear in parameter model. We could get least square estimates for the special case when all the errors where blame to equation errors, these are called as equation errors. Okay, I am going to blame everything to the equation errors.

This model general model which is non-linear in parameter, can I do something so that I use this kind of expression to arrive at a solution for the non-linear model. I am not able to transform. If I am removing here all those models which are transformable. Okay. So all those models you know, Nusselt number equal to something which are transformable I am just removing them. By linearizing transformation.

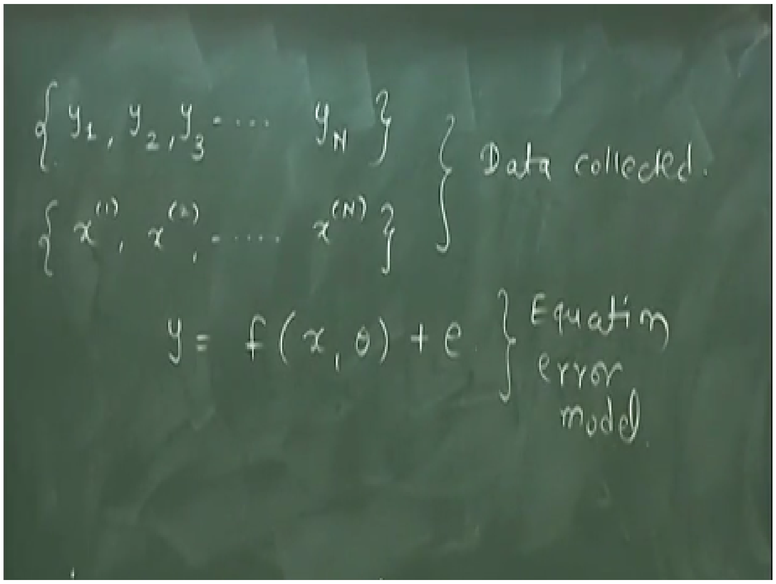
I am talking about a models like P V T relationships Redlich–Kwong equation where you cannot transform or that friction factor that that Blasius correlation, it is called Blasius correlation, friction factor correlation is you cannot transom, there is no way you can do it. Okay.

How do you now come up with – a way of using this business A transform A transpose A inverse A transpose Y, okay. And still, estimate these parameters of this theta this method which I am going to derive is called as Gauss-Newton method. Okay. Gauss-Newton method plays a trick. Well, as I told you that there are very limited options when it comes to approximating something.

We have looked at two options previously; one option was interpolation other option was Taylor series approximation. In fact, the Taylor series approximation gave rise to Newton's method if you remember. I am going to use a same trick I am going to use Taylor series approximation. Okay. Transform this model which is non-linear in parameter to locally linear in parameter model. Okay. And form an iteration scheme by which I will be able to use something like this.

Okay. I have able to use something like this. Okay. And then use it for parameter estimation. Okay, so let us see how we do this. Okay so I have this model which is non-linear in parameter model. I have collected database first of all.

(Refer Slide Time: 26:33)



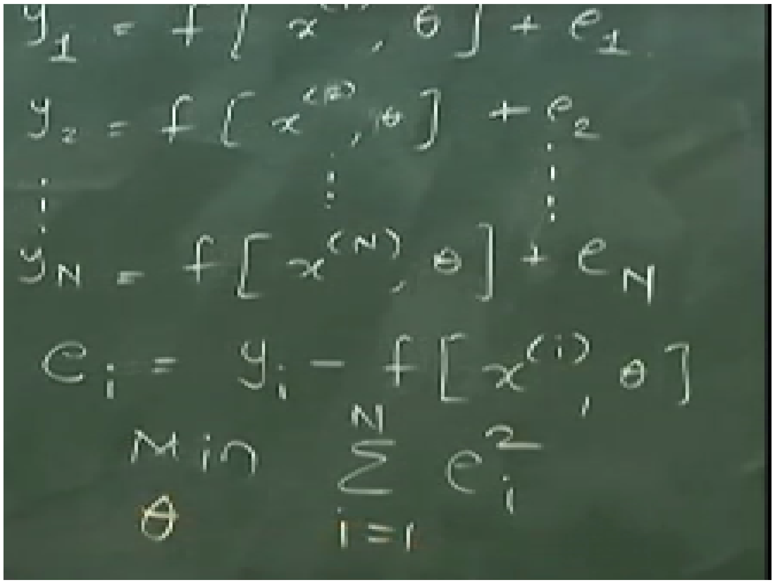
I have collected data y1, y2, y3 up to y capital N. This is the dependent variable data which is collected. And well, x could be a vector in general x could be a vector that is why I am giving this superscript, this is the notation which we have been doing throughout the course. x1, x2, correspondingly xn. These are the Data Collected. This is the Data Collected. And I want to fit a function form which is y = f of x theta + error. I want to fit a function form.

Now before we go to this method before we go to this method of least squares or successive least squares Gauss-Newton method let us first look at what is the raw problem then we will see how to form the. So what kind of equations that you have here? If you have this data and this y, if I am proposing a model which is equation error model, I will call this Equation Error Model, this is equation error model.

Why Equation Error Model? Because I am not saying that there is any error in x or y measurements of x or y, I am saying there is an overall error which is blame to equation error, okay which is blame to equation error. So this is a combination of all possible errors that can occur in approximation that means blame to one additive term here E. Okay. Now, this is a non-linear in parameter model, okay.

And what is the least square problem that would arise from this when I want to estimate theta. So what was the problem here? E transpose E, if you remember here. What is E transpose E? This is summation i going from 1 to n, ei square, right. Okay. So in this case my problem would be very, very similar, so I have these equations coming from this data.

(Refer Slide Time: 29:22)



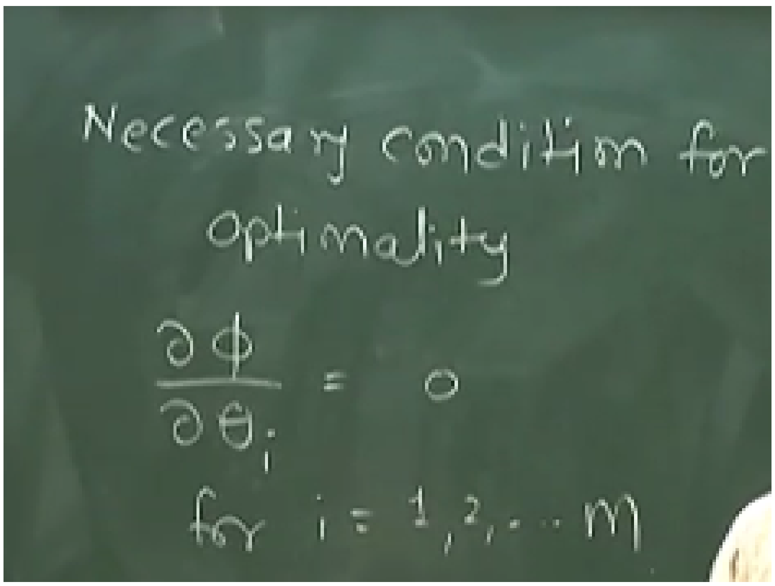
y1 = f x1 theta + e1 y2 = f x2 theta + e2. These are my data points, y1, y2, x1, x2 I am just substituting them in my equation form. How many equations are we get? Capital N equations I will get, large number of equations. Okay. So likewise I will get yn = f xn theta en, right. I got this N equations capital N equations, in how many unknowns? Theta, there are only m unknowns. So here theta belongs to Rm.

There are m parameters to be estimated. M could be typically 3 or 5 parameters to be estimated. You have data which is 100, 200 data points large number of data points. So now here you have large number of equations in only 4 unknowns. Okay. How do I want to estimate the estimate theta? So I will define an error ei that is yi – f xi theta. If I guess a value of theta if I somehow guess the value of theta I can estimate an error because x is known me, y is known to me.

Somehow I guess the value of theta. Okay. See for example, this relationship you are fitting in some homologous series, you take A B C values for the previous series, you get a good guess. There may not be too far away from the next compound then you are giving a good guess and you can compute ei, right if you give a guess. Now the way I want to estimate theta is again by least squares.

So I want to estimate minimize with respect to theta sigma ei square i going from 1 to n. Okay, let us call this, let us call this objective function as phi, this is my objective function. This is my objective function, okay. How do you solve this problem? What is the-- where will you get the optimum, what is the necessary conditions for optimality? dou phi/dou theta = 0. So you will get the necessary condition from optimality.

(Refer Slide Time: 32:25)

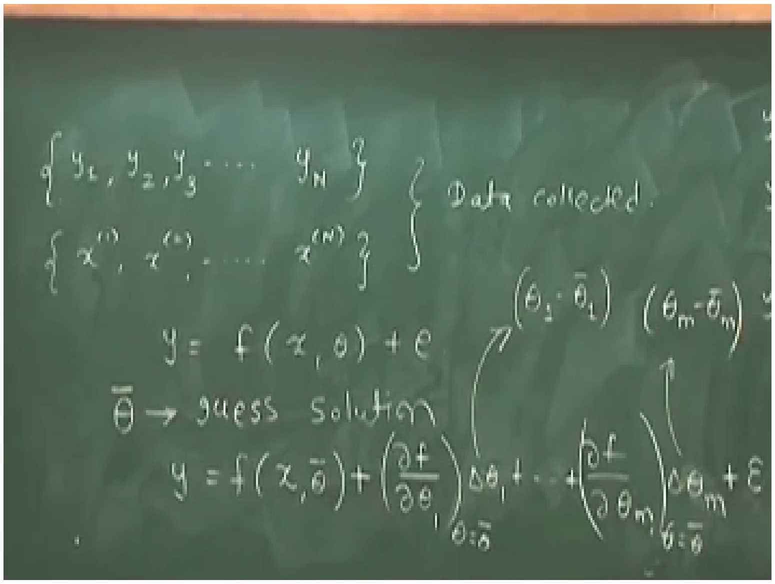


The necessary condition for optimality is. There are m parameters. I should differentiate this dou phi/dou with respect to theta i set = 0. How many equations I will get? I will get m equations in m unknowns typically this m equations and m unknowns will be non-linear with respect to theta. I can solve it by Newton–Raphson. I get m equation in the m unknowns, theta unknowns, I can solve this equation by Newton–Raphson, that could be a way to go. Okay, that could be a way to go.

You could just give this problem to the Newton–Raphson solver, keeps guessing. Every time it differentiates very, very painful task every time it differentiates and then it. Well, I can do this same thing by a little bit modification which is called as Gauss-Newton method. So that is what I am going to now derive at. Okay. So now what I am going to do is I am going to come up with a iterative procedure to solve the same problem.

I want to solve this problem, okay but with a little bit of modification.

(Refer Slide Time: 33:43)



Okay so what I am going to do now here. Okay. I am going to going to linearize this model. Okay. If I linearize this model that will help me solve the problem in slightly different way. Okay. So let us say theta bar is my guess. Theta bar is my guess solution. Okay. Now, I am going to linearize this model in the neighborhood of theta bar. Okay. So what it will be? Y = f x theta bar + dou f/dou theta 1. Okay.

I am going to linearize this using Taylor series expansion. Okay, now this e here was defined when the model was exact. Okay. Now I am approximating the model. So I am going to replace this by a term epsilon. I am going to replace this by term epsilon. Now just remember this data x, this data x is known to me, x is given to me I have collected data, y is known to me. If x is known to me and if I have a guess theta this value is computable. This is a known value.

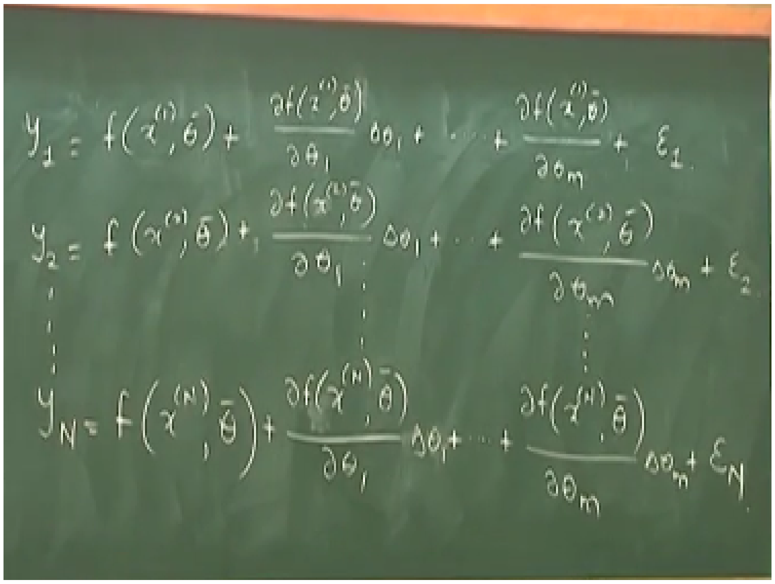
Okay. What is this delta theta 1? Delta theta is theta which is unknown – theta 1 bar, perturbation from guess solution this is perturbation from the guess solution. Okay. So this is my theta 1 bar. Likewise, this 1 is theta m – theta m bar. You have this partial derivative appearing here. When you do linearization where do you compute the partial derivatives? At which point? At a known value right. You compute the partial derivative with a known value.

So this partial derivative is computed at theta = theta bar, this partial derivative is computed at theta = theta bar, these partial derivatives are known to me. Okay. Now, if you look at this linearized model so you can view this as a transformation a local transformation of the model in which the transformed model looks like a linear in parameter model. Why this become a linear in parameter model?

With respect to delta theta 1, delta theta 2, delta theta 3 this is a linear in parameter model. Original model is not linear in parameter. The transformed model through Taylor series approximation is linear in parameter. Okay, so instead of solving for theta directly I could choose to solve for delta theta. I could choose to solve delta theta. If I solve for delta theta, then I can recover a new theta by adding delta theta to theta I will get a new guess.

And I go on doing this like in Newton–Raphson. Okay. So the next step after we linearization is, is to this equation for the linearized model.

(Refer Slide Time: 37:43)



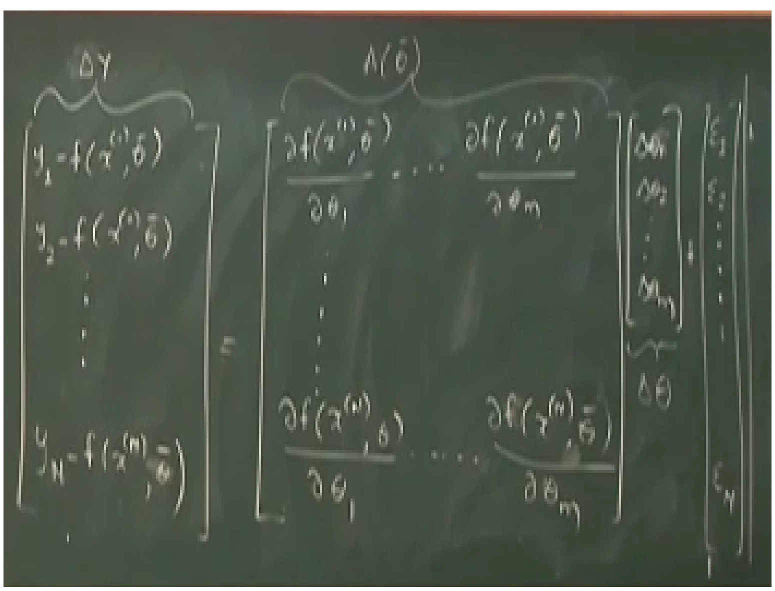
Okay, so what I am going to do is this will be y1 = f x1 theta bar + dou f x1 theta bar dou theta 1 delta theta 1 +, this is computed at theta bar epsilon 1. Then we have this second equation y1 = f x2 theta bar + dou f x2 theta bar dou theta 1 delta theta 1 dou f, epsilon 2. And likewise I have capital N equations. The notation become little bit complex but if you understand the concept it is not at all difficult.

So I write these equations I write these equations at theta m delta theta m + epsilon N. I have capital N equation with me now which are linearized, linearized in transformation through Taylor series, okay. These are partial derivatives computed at x1 theta bar, x1 theta bar, this partial derivative computed at x2 theta bar x2 theta bar and so on. So at each point, each data point you are linearizing the non-linear equation using Taylor series approximation.

Ignoring the terms higher than the second order and getting this transformed linear equation. Now with respect to delta theta 1, delta theta 2, delta theta m these are linear in parameter because these partial derivatives are computed at a fix point they are known to you. Okay. These are known to you. This f is known to you. This y is known to you. Okay, so now I can apply A transpose A inverse business. Okay. I have to do iteratively.

So now how to come up with the A transpose A inverse business, let us go back here. Now watch carefully what I am doing.

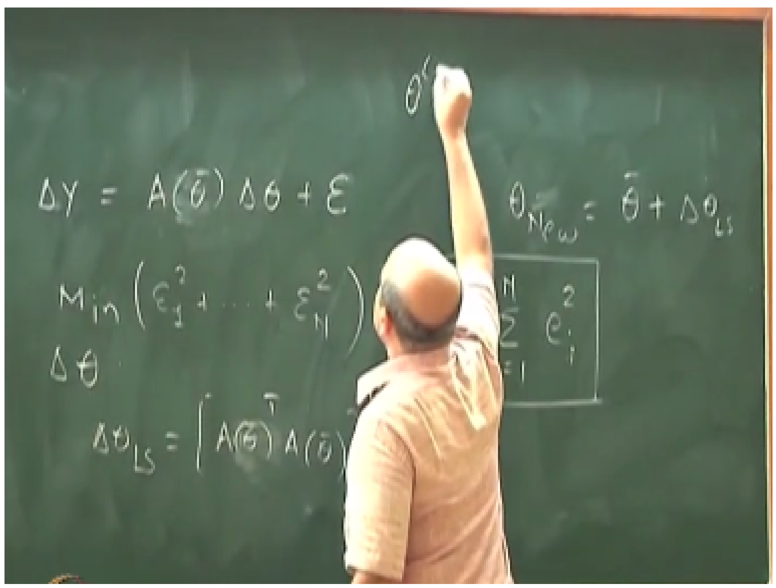
(Refer Slide Time: 40:39)



Now on the left hand side is going to be y1 – f x1 theta bar; y2 – f x2 theta bar likewise yn – f xn theta bar = this dou theta 1 and we create this matrix. Okay. This plus of course we have all these errors which are coming up. So + epsilon 1, epsilon 2 up to epsilon N. Okay, epsilon N. So let me call this my tricks as A theta bar. Why theta bar? Because theta bar is a guess, theta bar is a guess. Okay. So this matrix is a function of your guess. Okay.

But if you give me this equation here y1 to yn is known, theta bar is a guess, so this part f is known so this vector is known. Okay, let us call this vector as delta, delta y, let us call this vector as; let us call this vector as A theta; let us call this vector as delta theta okay. Then, we know how to solve this. We know how to get least square estimate of delta theta, isn't it? We know how to get least square estimate of delta theta.

(Refer Slide Time: 43:12)



So the model that we have finally got is delta y = A theta bar, A which is function of theta bar into delta theta + let us call this, that is all error vector. Okay. These are all small vector and this is a big epsilon, which you can see it, you should think about it as epsilon, so let me draw it as capital epsilon. So this is capital epsilon and what I would like to do is to find out now minimize epsilon 1 square to epsilon N square with respect to delta theta.

This problem is can be solved analytically. How do you solve this problem? So what I do here is delta theta least square = A theta R transpose A theta bar inverse A theta bar transpose delta y.

This is my solution. Okay, this is my solution. And then what do I do with this solution. What I do is to generate a iteration? So I will say now new theta, theta new = theta bar + delta theta least square. Okay, delta theta least square.

"Professor – student conversation starts" What do I do with new theta? I put it back I re-linearize around this new theta, okay and then keep on doing this. When will you stop? (()) (45:09) Not, not, yeah first vector. delta y. delta y should become as small as possible, delta y should become as small as possible. **"Professor – student conversation ends"** or in other words, you want sigma i going from, so we had this phi which was sigma i going from 1 to 1 ei square.

This should become smaller than certain value, these specified value. You wanted to minimize error in the original model, this is the transform problem. Okay, minimizing problem in the transform problem does not mean minimizing error in the original problem. So your termination criteria are based on the original errors not on the -- Original errors can be calculated actually what she had rightly pointed out is that this first vector will give you the original errors for every guess. Okay. If that vector becomes small you are done. Okay.

So you are iteratively going to calculate just like Newton–Raphson or Newton's method which we call, this is Gauss-Newton why Gauss neither Gauss nor Newton invented this but the word Newton comes because you are doing Taylor series approximation and locally linearizing, Why Gauss? Because we are using least squares, linear least squares, okay. So these two joints are merged to form this name, name of these two joints is from merged to give this name Gauss-Newton method to this.

So if I write a formal algorithm of course I will say that instead of theta bar I will say that you start with a guess theta 0, okay and then you get theta 1 from theta 0 and theta 2 from theta 0 and so on. So my iterative algorithm would like the theta k+1 = theta k + delta theta k. My iterative algorithm will read like this and where delta theta k is a solution of A theta k transpose A theta k delta theta k = A theta k transpose delta y.

I will call this delta y k because this right also theta depends upon theta bar. So if I write an iterative algorithm where k goes from here k starts from 0, 1, 2. And you stop only when the termination criteria is better. Okay. So this is Gauss-Newton method. This can be used to estimate parameters of any non-linear in parameter model, you do not have to worry about linear in parameter case.

And just like Newton–Raphson you can use this iterative to come up with an estimation of the parameters. Well, remember that whether you get a meaningful solution or not will depend upon what is your initial guess. So this guy is most important theta 0. Okay. You have to give a correct guess and this will come only from your engineering knowledge, okay. So how do you get a good guess? That is where you need your engineering physics, chemistry whatever background to come up with a good guess, if you give a good guess this method will converge to the solution.

This is an iterative procedure unlike the earlier case where you can form you could find the global solution in one shot, okay that is not possible here. Here you may not get a global solution you might get a local solution depending upon your guess. You might go towards a local minimum, you are minimizing problem iteratively by you know formulating a linearized problem which looks like linear least squares and there is a sequence of linear least square.

See what is done in Newton's method? Non-linear problem is solved by sequence of linear algebraic equations, you know, you form set of linear algebraic equation and you hope that sequence converges to the true value of the solution that is the true solution. Okay. What is done here? You form a sequence of linear least square approximations, okay. You form a sequence of linear least square problems which you hope to converge to the true optimum solution. Okay.

So this was the last in the series of lectures of approximations, and now on from—we come to close of this module on problem discretization and problem transformation. In a next class onwards, we will start looking at a tools okay. So the tools will be solving Ax = B, non-linear algebraic equations, OD initial value problems and a forth tool is of course stochastic tools which I am not going to discuss in this class.

So I hope to cover now three tools post mid-sem and these three tools are Ax = B. What we will see is that solving Ax = B; there is lot more to it than what we you have done in undergraduate. Okay. I will be spending almost three weeks on just how to solve Ax = B. Okay. You already have taste of large problems right, 10,000 equations in 10,000 unknowns, not uncommon when you are solving partial definition equations.

So you better have some better you know, y schemes to solve Ax = B otherwise we will end up into you know, the computation time become exceeding the large. So we will look at Ax = B, look at non-linear algebraic equation solving and move onto OD initial value problems and that is where we will close the post mid-sem. So the tools will be covered in the post mid-sem that will cover the entire course.