Bayesian/MMSE Estimation for MIMO/OFDM Wireless Communications

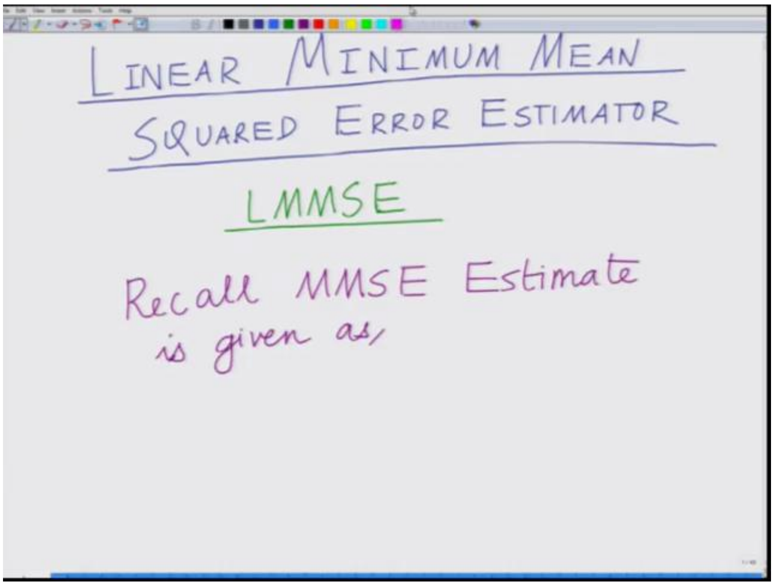
Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture - 15

Linear Minimum Mean Squared Error LMMSE Estimate Derivation - Part I

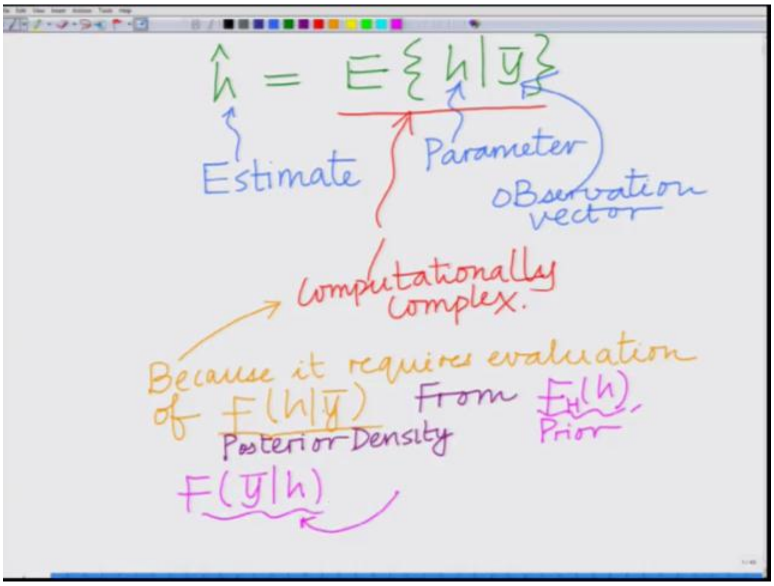
Hello, welcome to another module in this massive open online course on Bayesian MMSE Estimation for Wireless Communication Systems. So far, we are looking at the MMSE, we have looked at the optimal MMSE estimator when the parameter h and the observation vector y bar are jointly Gaussian and we have illustrated the application of MMSE estimation in several scenarios such as wireless sensor network, wireless channel estimation and we have derived both the estimator, the estimate and also the expression for the minimum mean squared error. So, starting with this module we will look at a different aspect of MMSE estimation that is we will look at the linear MMSE estimator.

(Refer Slide Time: 01:00)



So, now we will start looking at what is known as the linear minimum mean squared error estimator, this is known as the linear minimum mean squared error estimator and this is also termed as linear LMMSE where L is for Linear Minimum Mean Squared Error, this is the LMMSE estimate. So, in this module we will start looking at the LMMSE estimator and you can recall that basically the MMSE estimator, let us try to recall that the MMSE estimates the minimum mean square estimate. So, first we have to understand what the difference between the MMSE is and the LMMSE estimator recall that the MMSE estimate is given as the minimum mean squared error estimator.

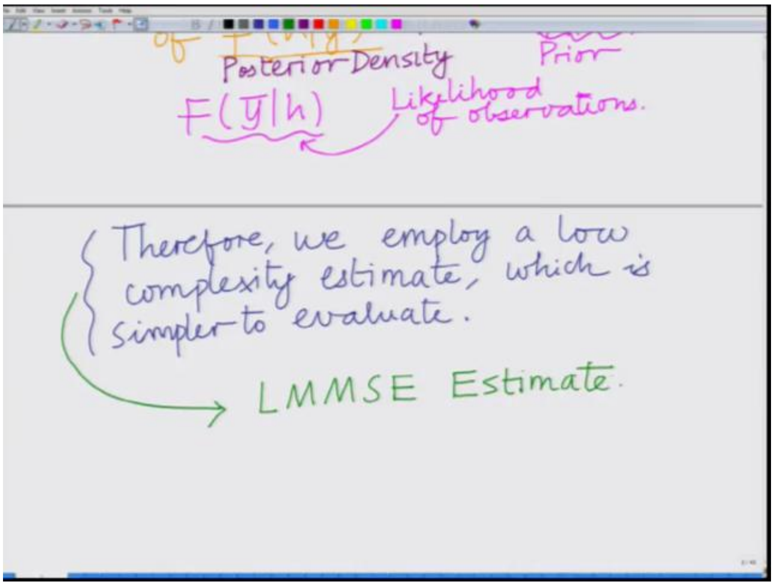
(Refer Slide Time: 03:08)



This is given as h hat equals the expected value of h given y bar, we have to say this is the optimal MMSE estimate where h hat, remember this is the estimate or rather the MMSE estimate, this is the parameter and this is the observation vector y bar.

However, this is computationally complex task, why are we saying that this is computationally complex because if you remember to compute this expected value of the posterior probability density function that is the expected value of h given y, first we have to compute f of h given y that is the probability density function of the parameter vector h given the observation vector y. So, this is computationally complex because it requires evaluation of f of h given y bar, remember this is the posterior density from f of h, this is the prior f of h given h comma f of y bar given h, remember this is the likelihood of the observation f of y bar, this is the prohibited density of the observations given this is the likelihood of the observation.

(Refer Slide Time: 05:40)



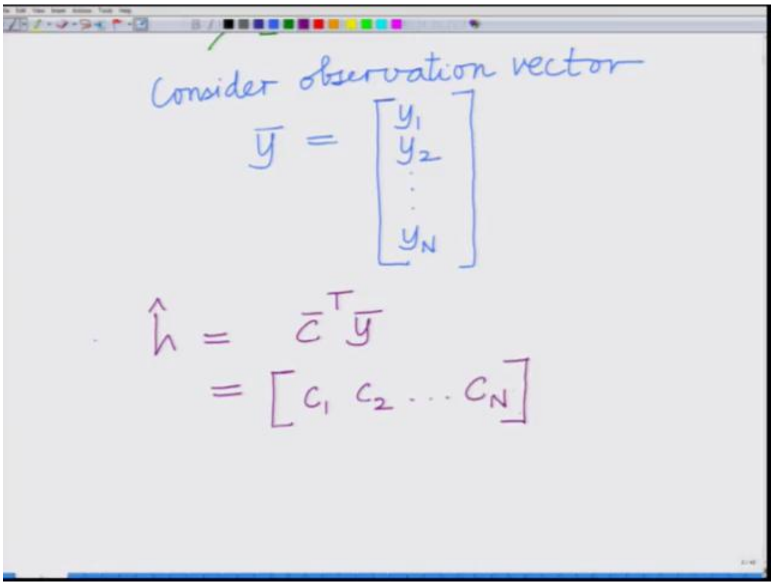
So, why is this computationally complex evaluating the MMSE estimator because we are given the prior probability density function f of h and we are given the likelihood that is f of the observation vector y bar given h, from these we have to evaluate the posterior probability density function which is f of h given y bar which is a computationally complex task and if you remember we had computed this for the Gaussian scenario that is when the single observation y and the parameter h are jointly Gaussian and even for that scenario if you remember we had to go through a lengthy and very elaborate derivation which is not possible to run and this was possible for a jointly Gaussian scenario.

When the probability densities are general, when the probability densities of the parameter and the likelihood are arbitrary in several scenario this is going to be very complex. So, it is not possible to evaluate this posterior probability density function rather easily in several scenarios. In fact, it is computational and very frequently also intractable and often frequently impossible to get a nice closed form expression for this posterior probability density function.

Therefore, we employ a simpler approach or a less complex approach or a less computationally intensive approach which will help us evaluate this estimator which will help us evaluate this estimate in a simpler fashion. So, he compromised a little bit on the optimality, but derived an estimate which is slightly simpler to evaluate and that is the linear MMSE estimate or LMMSE or the linear minimum mean squared error estimate. So, we choose a lower therefore, we employ a low complexity estimate which is easier to evaluate or which is simpler to evaluate, let us put is that way which is slightly simpler to evaluate. However, because it is not the optimal MMSE estimate, therefore it results in a slight loss of performance or yields on estimates or slightly lower accuracy, but we are accepting that because it is slightly simpler to evaluate alright we are accepting that slightly decreased estimation accuracy because we want an estimate which is simpler to evaluate and this is precisely the motivation for the LMMSE of the linear minimum mean square error estimator.

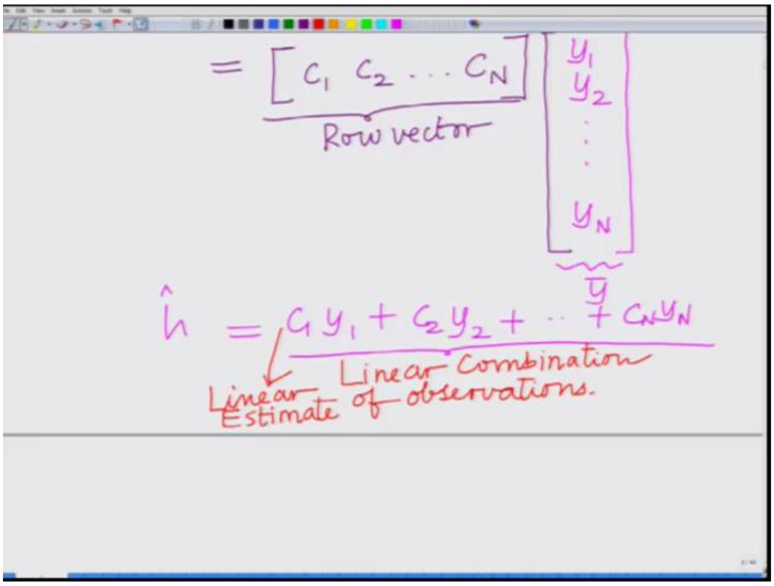
So, it is slightly simpler to evaluate and that is precisely the motivation for the LMMSE estimator.

(Refer Slide Time: 09:34)



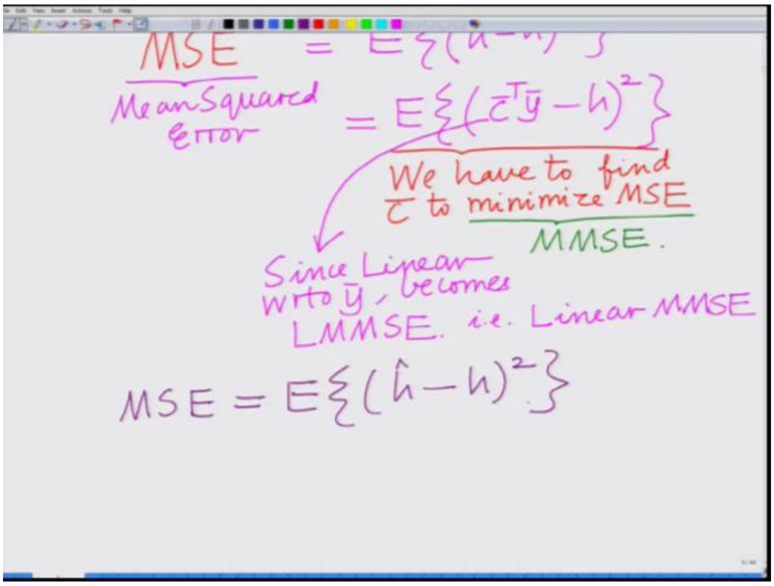
Now, how do we come up with a LMMSE estimator? Let us consider a scenario, consider the observation vector y bar which is given by again similar to previous we have the observation vector y bar which is comprised of N observations y 1 y 2 up to y N. However, now we will form the estimate h hat by linearly combining these observations as c bar transpose y bar which is c 1 c 2 c N, this is the row vector, this is the row vector times y 1 y 2 y N which is the column vector recalls c bar is being defined as follows, this is a column vector c 1 c 2 up to c N.

(Refer Slide Time: 10:36)



Therefore, c bar transpose is naturally going to be N dimensional low vector y 1 y 2 y N that is y bar this is N dimensional column vector, this is y bar. Therefore, c bar transpose y bar, this will be naturally equal to, so your h hat equals c 1 y 1 plus c 2 y 2 plus so on up to c N y N and as you can see this is linear because this is a linear combination. So, we are obtaining the estimate by a linear combination of the observations therefore, this is a linear combination of the observations, this is a linear estimate, this is a linear combination of observations, therefore you can see this is basically h hat equals c 1 y 1 plus c 2 y 2 plus so on up to c N y N. This is therefore a linear estimate.

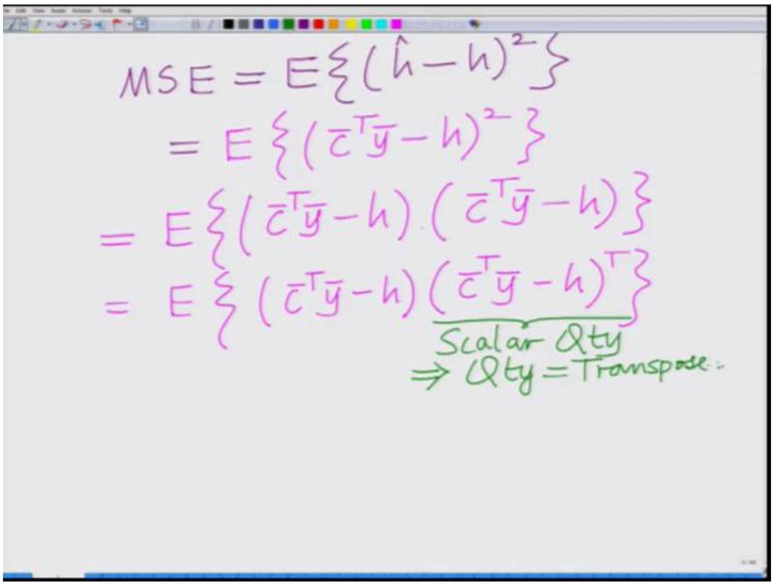
(Refer Slide Time: 12:38)



Now, we want to find the mean squared error that is the MSE. So, the MSE is the mean squared error, this is equal to expected value of h hat minus h Whole Square which since h hat is equal to c transpose y bar, and this is simply equal to c bar transpose y bar minus h Whole Square. Now, what is the key, basically the idea is we have to find this vector c bar such that this mean square error is going to be minimized. So, we are going to find c bar such that this mean square error is going to be minimized, that is going to give us the MMSE that is the minimum mean squared error and now this is also linear in the observation vector y bar. So, together this becomes the linear MMSE estimator that is basically the reason this is known as the LMMSE estimator.

So, we have to find c bar to minimize MSE, this becomes your MMSE that is the minimum mean squared error and this is also the linear in the observation y 1 y 2 y N that is linear in the observation vector y bar therefore, this is also the LMMSE estimator. Now, since this is linear with respect to y bar, it also becomes LMMSE that is linear minimum mean squared error estimator. Now, let us simplify this expression for the MSE, we have MSE which is the function of this vector c bar, MSE this is equal to expected value of h hat minus h whole square which is equal to expected value.

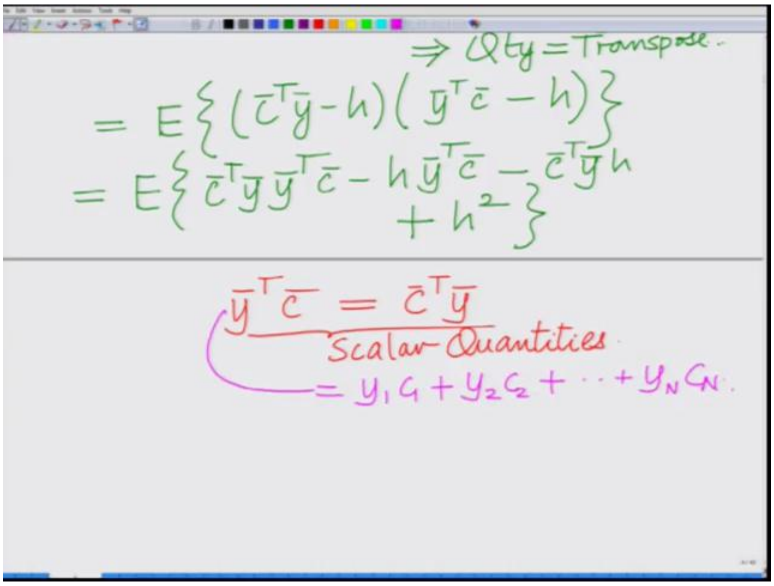
(Refer Slide Time: 15:42)



For h hat I am going to substitute c bar transpose y bar minus h whole square which is equal to expected value of c bar transpose y bar square of itself c bar transpose y bar minus h into c bar transpose y bar minus h, realize that this both are scalar quantities.

Therefore, the square is simply the quantity times the product itself and now I am going to use a property that is since c bar transpose y bar minus h is a scalar quantity that is it is a number, basically it is equal to the transpose of itself. So, if you have a scalar quantity, basically it is equal to the transpose of itself. So, I can write this as c bar transpose y bar minus h times c bar transpose y bar minus h transpose, since basically this is a scalar quantity implies quantity is equal to its transpose.

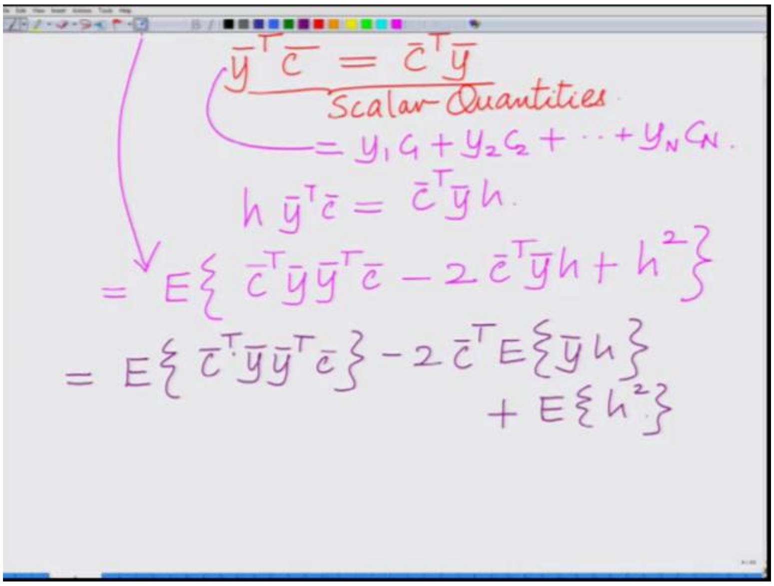
(Refer Slide Time: 17:18)



Now, we can simplify this as expected value of c bar transpose y bar minus h times c bar transpose y bar minus h. So, c bar transpose y transpose is basically y bar transpose c bar minus h is of course a scalar parameter. So, h transpose is h, this is what we will get y bar transpose c bar minus h and now we will expand this product term by term this is c bar transpose y bar into y bar transpose c bar minus h y bar transpose c bar. Of course, h is a scalar quantity. So, we can write that anywhere minus c bar transpose y bar into h plus h into h, h is a scalar quantity and I can simply write it as h square. Now, observe that y bar transpose c bar equals c bar transpose y bar right, y bar transpose c bar is a scalar quantity correct, y bar transpose c bar equals c bar transpose y bar. In fact, both of them are equal to c 1 y 1 plus c 2 y 2 plus so on up to c N y N. So, these are scalar quantities.

In fact, both are equal to y 1 c 1 plus y 2 c 2 plus up to c N y N. Therefore, we have y bar transpose c bar equals c bar transpose y bar implies.

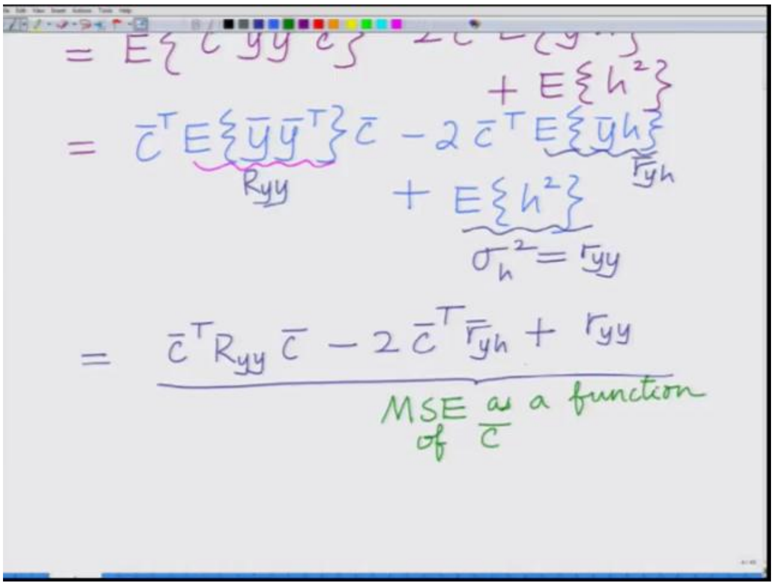
(Refer Slide Time: 19:16)



Now I can multiply both sides by h, h is scalar quantity h y bar transpose c bar equals when c transpose y bar into h which means now basically these two quantities c bar transpose y bar h right h y transpose y bar transpose c bar and c bar transpose y bar h, these are equal. Now, what we are going to do? We are going to use this principle to simplify this and this is therefore going to be equal to expected value of c bar transpose y bar y bar transpose c bar minus twice because both these quantities are required twice c bar transpose y bar h plus h square where we use the property that h into y transpose y bar transpose c bar equals c bar transpose y bar into h.

Now, we will take the expectation operator inside take the term by term expectation that is expected value of c bar transpose y bar y bar transpose c bar minus twice c bar transpose expected of value y bar into h plus expected value of h square.

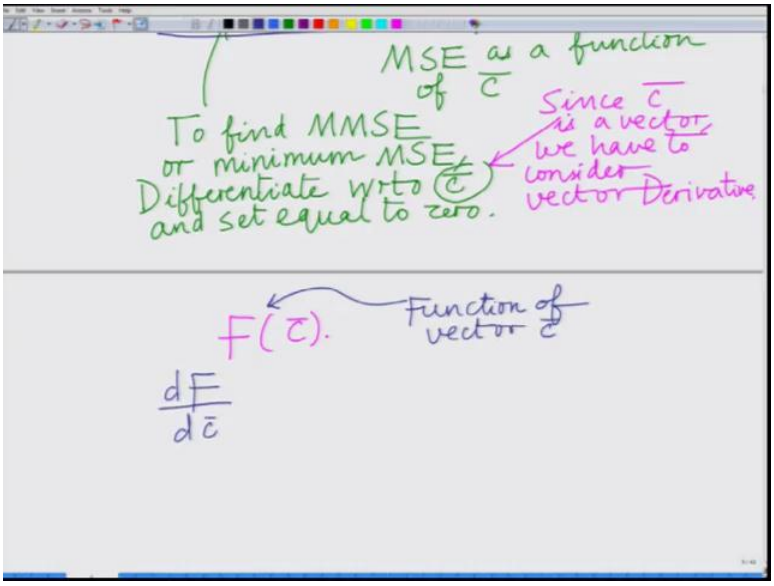
(Refer Slide Time: 20:53)



Now, we can further take this expectation operator inside, this is c bar transpose expected value of y bar y bar transpose c bar minus twice c bar transpose expected value of y bar h plus expected value of h square. Now, realize that expected value of y bar y bar transpose, this is nothing but your r y y expected value of y bar h, this is a vector r y bar h expected value of h square, this is simply sigma h square which we can also denote by r y y and therefore you can simplify this quantity as c bar transpose r y y c bar minus twice c bar r bar y h plus r y y, what is this? This is your MSE as a function of c bar which is the combining vector.

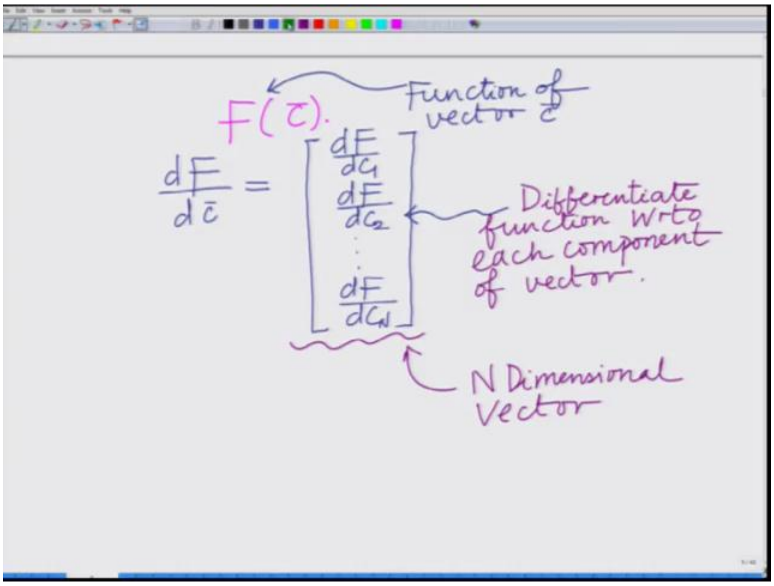
So, now we have found the MSE that is the mean squared error as a function of this combining vector c bar, but we are interested in the minimum MMSE that is the minimum mean squared error.

(Refer Slide Time: 22:49)



Therefore, we have to find that c bar which minimizes the mean squared error which means we have to differentiate this with respect to c bar and set it equal to zero. To find the MMSE, differentiate with respect to c bar and set it equal to zero. However, c bar is a vector therefore, we have to consider a vector derivative since c bar is a vector, we have to basically consider vector derivative, how to derive, how to differentiate this quantity with respect to the vector c bar.

(Refer Slide Time: 24:43)



So, next we are going to introduce this notion of a vector derivative. Let us consider any function f of c bar, this is f of c bar is a function of vector c bar, then the derivative of f with respect to c bar equals d f by d c 1 and d f by d c 2 so on d f by d c N, that is we have to differentiate the vector with respect to each component, differentiate the function with respect to each component of this vector c bar that is basically your vector derivative, that is if you have a function f of c bar, what I have to do is I have to basically differentiate this function f of c bar with respect to each component of this vector c bar. So, if c bar is an n dimensional vector because I have to differentiate with respect to each component of c bar, I will get basically n components.

The vector derivative is going to be an n dimensional vector because it is a vector containing the different components d f by d c 1 d f by d c 2 so on d f by d c N. So, naturally this is going to be an n dimensional vector. We are going to consider that the vector derivative which is an n dimensional vector.

So, let us stop this module here, what we have started looking at? We have started looking at different paradigm or a different framework known as linear MMSE estimation.

So, far we have looked at MMSE estimation which we have said is very complex because it requires the evaluation of the posterior probability density function. In order to avoid that complexity, what we have done is we have started looking at this linear MMSE expression that is the linear minimum mean squared error estimator which involves this vector combining vector c bar of coefficient c 1 c 2 up to c N, using that we form the LMMSE linear estimate c 1 v 1 plus c 1 h 1 y 1 plus c 1 y 2 plus so on c N y N which can be represented as c bar transpose y bar. Now, we have to find this optimal vector c bar which minimizes the mean squared error that gives the MMSE minimum mean squared error and therefore, since this is linear in nature that will become the LMMSE estimator.

So, we will stop here and will complete this derivation of the LMMSE estimate in the next module.

Thank you very much.