Real-Time Digital Signal Processing

Prof. Rathna G N
Department of Electrical Engineering
Indian Institute of Science - Bengaluru

Lecture – 33

Adaptive Filter

So, welcome back to real time digital signal processing course. So, we will cover today little on adaptive filter. So, to give you as you can see this is the module 3 what you would be covering. So, what are the two modules we covered in the previous classes is listed here. That is we covered in module 1 basic architectures and some number system and then in the second one we discussed about filters both FIR and IIR filters. And then we went on to see frequency domain algorithms discrete Fourier transform and how this can be made faster using Fourier transforms.

So, in the last class we little bit discussed about random process and then we will be taking up today continuing on that. So, before that we will discuss about in module 3 what are the subjects we will be covering. So, first one is adaptive filters. So, we will be taking least square mean square algorithm and its applications and then we will just say normalized LMS and then even the RL is we will just discuss we will not go on to derive the thing, but more derivative we will be doing it on the least mean square LMS algorithm.

So, later on we will cover basics of image processing, basically we will be covering a discrete cosine transform with little bit of introduction to our image processing. And how we will be implementing in the hardware which is going to give us the full site just like our FFT algorithm. So, coming to continuation of the random process. So, little bit of theory in the last class we have covered. Today we will discuss about little on the autocorrelation function of the random process, $x(n)$ defined $r_{xx}(n,k) = E[x(n)x(k)]$.

So, we say random process is stationary if it is a statistic do not change with time. So, the most useful and relaxed form stationary is the wide sense stationary what we call it which is named as WSS. So, process that satisfies the following two conditions. So, that is we call it as wide sense stationary the mean of the process is independent of time in this case what it is given as $E[x(n)] = m_x$. So, we say thus m_x is constant that is what we say it is independent of time.

Then the autocorrelation function it is going to depend only on the time difference which is given by $r_{xx}(k) = E[x(n+k)x(k)]$. So, we say k is the time blank in digital domain. The two important properties of our autocorrelation function $r_{xx}(k)$ with respect to our WSS process is defined in this way. First is the even function. So, what do we mean by that? So, $r_{xx}(-k) = r_{xx}(k)$ which is this the other property is it is bounded by giving that that is the $|r_{xx}(k)| \leq r_{xx}(0)$.

So, here we call $r_{xx}(0) = E[x^2(n)]$. So, we call it as mean square value. or the other name is power of random process $x(n)$. So, if $x(n)$ is a zero- mean random process then what we have is $E[x^2(n)] = \sigma_x^2$. So, continuing with the thing consider the sinusoidal signal as an example which is given by $x(n) = A\ cos\ (\omega_0 n)$.

So, we have to find its mean and then autocorrelation for of the $x(n)$. So, for the mean what we substitute is $m_x A\ E[\cos(\omega_0 n)] = 0$, this is m_x equal to. So, we know that our cos function is given by So, you are taking the expected value of this cos omega naught n over the thing. So, when you add them up you will be getting 0.

So, for the autocorrelation function, so how we are going to calculate the thing which is nothing but $r_{xx}(k) = E[x(n+k)x(n)]$. So, here we have taken $x(n)$ here which is nothing, but $= A^2 E[\cos(\omega_0 n + \omega k)\cos(\omega_0 n)]$ this is the function what we have to solve. So, when we solve with respect to $[\cos(\omega_0 n + \omega k)\cos(\omega_0 n)]$ when you expand the thing then it becomes as we know that it is $\frac{A^2}{2} E[\cos(2\omega_0 n + \omega_0 k)] + \frac{A^2}{2}\cos(\omega_0 k) = \frac{A^2}{2}\cos(\omega_0 k)$. So, that is what is left out from that other the terms are going to get cancelled. So, you can expand it and then look at them.

So, that what is the thing that autocorrelation function of a cosine wave is the cosine function of the same frequency ω_0. So, as we started with $\cos(\omega_0 n)$. So, we will be seeing that it is a function of ω_0 itself. So, coming with how we are going to calculate power spectrum and then cross correlation next function we will see the thing. So, we are using the widely used random signal for many applications which is we said it is white noise what we will be considering $v(n)$ with zero and variance as σ_v^2.

Then its autocorrelation function is given by $r_{vv}(k) = \sigma_v^2 \delta(k)$. So, is we know that $\delta(k)$ is a delta function with amplitude what we call it as σ_v^2 at lag $k = 0$. And its power spectrum is given by $P_{vv}(\omega) = \sigma_v^2, |\omega| <= \pi$. So, this shows that the power of the random signal is uniformly distributed over the entire frequency range.

So, now defining the cross correlation function between two wide sense stationary process $x(n)$ and $y(n)$ which is defined by $r_{xy}(k) = E[x(n+k)y(n)]$. So, this function has the property that is $r_{xy}(k) = r_{yx}(-k)$. So, what we said here it is a even function. So, even the cross correlation is a even function what you can look at it. So, then what happens to $r_{yx}(k)$ is simply the folded version of what we call it as $r_{xy}(k)$.

So, now take an example to see that what will be the cross correlation of FIR filter with input output equation what it is given by equal to $y(n) = x(n) + ax(n-1) + bx(n -$

2). So, you can assume this is small $y(n)$ equal to So, we assume the white noise with 0 mean and variance σ_x^2 in this case as the input signal $x(n)$. Find the mean that is m_y and the autocorrelation function $r_{xy}(k)$ of the filter output $y(n)$. what we have to do it. So, for the mean what we are going to substitute $m_y = E[y(n)]$ what we are going to take it which is nothing, but we will be putting it on the right hand side $E[x(n)] + aE[x(n-1)] + bE[x(n-2)] = 0$. Now the autocorrelation function, so what happens to the thing? So, we will be seeing $(1 + a^2 + b^2)r_{xx}(k) + (a + ab)r_{xx}(k-1) + (a + ab)r_{xx}(k+1) + br_{xx}(k-2) + br_{xx}(k+2)$

so on we will be substituting for these functions. So, which is given by what is the thing?

$$\text{So,} = \begin{cases} (1 + a^2 + b^2)\sigma_x^2, & \text{if } k = 0 \\ (a + ab)\sigma_x^2, & \text{if } k = \pm 1 \\ b\sigma_x^2, & \text{if } k = \pm 2 \\ 0, & \text{if otherwise} \end{cases}.$$

So, you have substituted this and then calculated our $r_{xx}(k)$ using the equation what we had looked in the previous slides. So, now how we are going to calculate for a finite length sequence mean and then autocorrelation.

So, what we have is we call it as $\bar{m}_x$ equal to the average value of it what we will be taking $\frac{1}{N}\sum_{n=0}^{N-1} x(n)$. So, where N is the number of samples available for the short time analysis. So, this is mean is defined with respect to this equation. And then how we are going to take for the same sample, what will be the autocorrelation function which is defined as $\bar{r}_{xx}(k) = \frac{1}{N-k}\sum_{n=0}^{N-k-1} x(n+k)x(n), \quad k = 0,1,\ldots,N-1$.

So, the next is we will take up an example for to see that what will be our mean and autocorrelation and cross correlation with respect to finite length signal. Here we have assumed $x(n) = A\cos(\omega_0 n) + v(n)$ is the error signal what we have put the thing is signal is given by $m_x = AE[\cos(\omega_0 n)] + E[v(n)] = 0$. So, from the previous example we know that for $\cos(\omega_0 n)$ it is nothing, but $\frac{A^2}{2}\cos(\omega_0 k) + \sigma_v^2\delta(k)$. So, what happens to its power basically $P_{xx}(\omega) = \frac{A^2}{2}\delta(\omega_0) + \sigma_v^2$

So, this one should be $|\omega| < \pi$ in this case. So, now we will see how to look at the adaptive noise cancellation. So, what is that adaptive noise cancellation? We will first derive it and then we will see that how we can do the cancellation of it. So, the adaptive noise cancellation is nothing, but this is an effective method to remove additive noises from the contaminated signals. When do we say that it is additive along with the input sequence you have the noise that is what we consider we often hear.

So, with the desired signal if there is a noise then we call in the added fashion we call it as a adaptive sorry additive noise. So, it has been widely used in the fields of what are the applications in telecommunication, radar and sonar signal processing. So, telecommunication we know that communication channel basically has the noise. So, it depends on which channel you will be using it. So, how to eliminate or cancel the noise we know about it and we have looked in the little bit of examples of radar and sonar I discussed in the previous classes.

So, radar when it is sending the signal when it comes back. So, you know the medium may affect the signal. So, how we are going to adapt to the different noises that is what it is going to be. And, even in the sonar we know that it is in the sea. So, there will be different signals which is going to be refractive in nature.

So, those how you will be getting it and then from along with the rest of the noises how you will be separating it. So, we have to cancel the noise and then take the signal that is what we call it as it is going to happen in adaptive way. So, we say that one more example is most of you use ear earphones and then headphones. So, what it says is your earphone and then headphone if they fit perfectly. Then what you say is you are unable to hear any outside noise.

So, they have to be filtered out and then you will say that high frequency noise is coming from your co-worker which has to be cancelled out. So, you call them as chatty co-worker I do not want to listen to them. Then if nothing is going to come when you are wearing your earphone or headphone then you say that you have cancelled the noise perfectly. So, this we call it as active noise cancellation in this case it can be there are two noises what we call it one is passive noise and the other one is the active noise. Here we call it as if you are unable to hear from outside as a passive noise.

So, we will define it in a while basically and then in the active noise cancellation. So, what is the thing is going to happen? Your headphones are here, but neutralize the ambient noise. So, you will be seeing that in an aeroplane this thing engine noise or in a car engine noise is going to cause. So, if you are outside noise when you are driving or whatever may be the thing if you want to cancel it out. You can use the noise cancelling technology basically, but still you will be left out with noise not completely gone.

So, this we call it as a active noise. So, what is it active noise work by incorporating microphones into your headphones. which listen to the outside noise and generate a phase inverted sound that effectively cancel out your ambient noise before it reaches your ears. So, exactly we want to cancel it which is may not be possible. So, that is the reason why in the active noise cancellation you may hear little bit of noise presented it most of it is suppressed. That is what it says in other words your adaptive noise cancellation or active noise cancels noise by creating equal but opposite noise.

So, when it will be desirable not desirable one can look at the literature and then work out what you want to design that is what it is going to depend on. So, in the active noise cancellation it is best suited for real time implementations because when you are using the microphone recording some of the open air what you will call it as speech or music or whatever may be the thing. So, if you can take with the one more mic the surrounding noise and use it as a noise and you want to suppress that you can do that. This is how real time is going to work. So, we will see some examples how we will be doing this cancellation.

So, we will ask the question whether the which has better sound quality, whether whatever you have used the earphones and then earbuds or when you do the active noise cancellation which one will be good. So, we will see 2 sets of earbuds when you will say quality I have to compare 2 sets basically of earbuds. So, in this case we will say it is from the similar build and tuning quality what we have it is also similar. And then the earbud with better passive noise isolation will be sounding better than the ones that rely on active noise cancellation to block out ambient noise. So, you will be seeing that sometimes passive noise cancellation is better compared to active noise.

So, passive noise cancellation what we call it as PNC or isolation is when your headphones, earbuds or earphones or your monitors in ear monitors naturally block outside noise. So, in other words in your earbuds are isolating you from ambient noise instead of actively using technology to cancel it out that is why you will hear this technique called both passive noise isolation and passive noise cancellation. So, that is what the name given to it when you are fully disconnected from the external noises. So, now we will see why we need adaptive filters. So, we have discussed about the linear filters.

So, it should be triggering in your mind that both FIR and IR filters we used in the previous classes we have a they are called linear filters. It is not linear phase filters, they are linear filters. Output is linear function of the filter input. So, what are the design methods that are available? So, we will see the classic approach one is we can use the frequency selective filters such as we can use low pass band pass or notch filters as you know low pass if I want to eliminate higher frequencies. band pass is only the frequency of interest what I want to pass it.

And notch filters you know certain single tone or multi tone you know the this thing frequency of them you can use the notch filters to suppress that. As an example we had taken it as a line this thing frequency that is our electric lines 50 hertz what it is. So, we can use the notch filter to eliminate it. The other one is what we can design is the optimal filter basically. So, how we are going to use this we will see it in a while and then this is mostly based on minimizing the mean square value of the error signal.

So, from the desired signal so will be this thing subtracting from the original signal and see how much error is left out whether we can try to minimize that error what we look at it and then design an optimal kind of filter. So, there are 4 aspects involved with our adaptive filters which are they. The first one is the signals being processed by the filter. So, whatever input signal you are going to feed it how the filter structure is going to behave with it. The other one is the structure to design this filter defines how the output signal of the filter is computed from its input signal.

The third one is the parameters within the structure that can be iteratively changed to alter the filters input output relationship. So, usually we call this as weights by modifying the weights whether my input and output structure can be changed that is what we will be looking at it. So, the next one is the adaptive algorithm that describes how the parameters are adjusted from one time instant to the next times. So, these are the 4 aspects of definition what we are going to follow. So, now we will compare with the real world signals basically.

So, what we want we are looking at is it is desired to extract a certain component of we call it $d(n)$ as the desired signal from the $y(n)$ whatever we have the output from that we want to extract our desired signal. So, that this was contained in our input signal what we call it $x(n)$ or it may be to isolate a component of $d(n)$ within the error $e(n)$. that is not contained in $x(n)$. So, whatever error which was introduced in between whether we can minimize that or isolated from that desired signal what we are looking at. So, what does it look like here it is input is contaminated with noise.

So, we are trying to extract the desired signal. The other one is if we know the error then we can model it and then get the thing here. What is it from the it is not contaminate input is not contaminated with the noise, but the channel make have a noise just like your communication channel is going to introduce the noise any of that for that matter. So, whether we know the thing whether we can extract from that the desired signal. So, then what happens to get these things only we may vary the weights $w(n)$. and we may not be interested in what is my input, what is my output or even desired this thing input what I am going to have or a function.

So, I want to see the weights so that if I can match $x(n)$ and $y(n)$ I know that what I want to have it as a result. So, there are situation with in which what we say in the real time $d(n)$ is not available at all basically. So, then what is the what you are looking at is also one of the important thing if you do not know what you have to look for it. In such situations adaptation typically occurs only when $d(n)$ is available. So, if you know only $d(n)$ or you pinpoint something what why I want to look at it then I can apply adaptive algorithm.

So, when $d(n)$ is unavailable then how we are going to deal with this kind of signal. We typically use our most recent parameter estimates to compare our $y(n)$ in an attempt to estimate the desired response signal $d(n)$. So, I know in the previous case I have got this output $y(n)$ and this output similarly looks like the previous one. So, then I will try to see that this is what I wanted to have because in the previous one I had the desired signal basically. So, then I will be looking in the present situation this is what I am looking at.

So, in some more real world situations what is it? $d(n)$ is never available, there it was not available, it will be never available. In such cases use your hypothetical blind that is predefined statistical behavior or amplitude characteristics. to form suitable estimates of your $d(n)$ from the signals available to the adaptive filter. So, such methods are collectively called blind adaptation algorithm already I have told you hypothetical basically blind.

So, that is what you will be applying to get the desired signal. So, these algorithms there are two varieties, one is the steeped descent algorithm, the other one is the maximum likelihood optimization. In this present situation we will be covering steepest descent algorithm, those who are interested can look into the literature for maximum likelihood optimization algorithm to get the signal from the real world. So, we will see now what is the generic block diagram for adaptive filter. So, the signal along with noise or without noise characteristics are often non-stationary and the statistical parameters vary with time that is what we define our signal basically. And then adaptive filter has an adaptation algorithm that is meant to monitor the environment and vary the filter transfer function accordingly.

So, based on the actual signals received, so what we are going to do? Attempts to find the optimum filter design is going to be considered. So, you are given the generic block diagram of an adaptive filter here, what does it contain? $x(n)$ is an input and then what we have is the digital filter here, how these weights are going to be altered is from the adaptive algorithm. As you can see dotted line which is going to change the weights of this filters. Based on what this algorithm is going to work on, it is going to take the input and then it is going to take we know in this case desired signal is known and then we will be subtracting our output from the filter and then the desired signal which we call it as error $e(n)$.

These are the two inputs to our adaptive algorithm. So, based on the thing, so we will be minimize the error by varying our weights. So, that is what error minimization here what we are going to do it. So, how we are going to apply this adaptive filter for our FIR filter as an example we will see it. So, this is our block diagram of FIR filter. So, our data flow diagram what you can call it x(n) is the input and earlier in a FIR case we had taken it as $b_0(n)$ $b_1(n)$ and then $b_{n-1}(n)$. So, L length filter what we have considered here we will call them as weight function because we are going to vary these weights w_0 w_1 and w_{L-1}.

And our input z^{-1} we know that in the z domain it is going to delay our input. So, we will be delaying $x(n)$ as $x(n-1)$ and then last one will be $x(n-L+1)$. So, this is the L length filter. So, we know that convolution theorem $y(n)=\sum_{l=0}^{L-1} w_l(n)s(n-l)$.

So, instead of giving x's input And this is our source signal what we call it s and w will be weight of the filter what we will call it. Now, represent this in terms of equation what we call it as a vector basically notation. So, which is given by $x(n) \equiv [x(n)x(n-1) \dots x(n-L+)]^T$ what we will have it. So, and the coefficients vector also represented in this format, $w(n) \equiv [w_0(n)w_1(n) \dots w_{L-1}(n)]^T$, l length coefficients what we are taking its their transpose what will be looking at.

So, now how we are going to represent now our $y(n)=w^T(n)x(n)=x^T(n)w(n)$. So, you can see our summation in terms of vector multiplication is going to be represented in this fashion. So, the filter output $y(n)$ is compared to the desired signal $d(n)$ to obtain the error signal. So, we said from this equation we know that $e(n)$ is given by or $y(n)$ will be equal to what is it error function is nothing but desired signal $d(n)-y(n)$ So, which is equal to $d(n)$ by substituting $y(n)$ with this it is going to be $w^T((n)x(n)$ then what we call it as error as in terms of our this thing random variable $\xi(n)$ what we call it as error function.

Which is equivalent to our expectation of $E[e^2(n)]$ basically what we represent what is the thing here are $\xi(n)$ is given by. So, expected value of we are taking the square error squared what we want to minimize basically. So, we are taking the square on both sides of these 2 sides. So, it will be $\xi(n)$ is nothing, but $E[d^2(n)]$.

So, it is nothing, but what I am putting from here is. we are calculating $e^2(n)$ which is nothing but $\left(d(n)-y(n)\right)^2$ what we will be taking it by substituting $y(n)$ and then expanding it. So, you will be getting this equation. So, what is it? $E[d^2(n)]-2p^Tw(n)+w^T(n)Rw(n)$ because why we have called it as $p^Tw(n)+w^T(n)Rw(n)$. So, expand this where p is the cross correlation vector defined as $E[d(n)x(n)]$. which is nothing, but the sequences $[r_{dx}(0)r_{dx}(1)\cdots r_{dx}(L-1)]^T$ what we will have it and then $r_{dx}(k) \equiv E[d(n+k)x(n)]$ and then R is our autocorrelation matrix which is given by $E[x(n)x^T(n)]$. So, when you equate it here you will be seeing that $(d(n)-w^T((n)x(n))^2$. So, which comes down to you will be seeing that equal to $[d^2(n)]-2d(n)+w^T(n)x(n)$ and then what is the thing, $(w^2)^Tx^2(n)$. So, when you take the expectation on both the sides we have represented $E[e^2(n)]=\xi(n)$ So, you will be substituting it here in this equation. So, you are seeing that $r_{dx}=E[d(n+k)x(n)]$ that is our desired and then n what will be taking combining with the thing ok. So, that will be our desired signal and then R is the autocorrelation matrix which is given by you will be seeing that which is nothing, but $R \equiv E[x(n)x^T(n)]$. So, which is given by $\begin{bmatrix} r_{xx}(0) & r_{xx}(1) & \dots & r_{xx}(L-1) \\ r_{xx}(1) & r_{xx}(0) & \dots & r_{xx}(L-2) \\ \vdots & \dots & \ddots & \vdots \\ r_{xx}(L-1) & r_{xx}(L-2) & \dots & r_{xx}(0) \end{bmatrix}$

So, you will be filling up this matrix in this way, this is $r_{xx}(0)$.

So, what is this matrix? We call this is a symmetric matrix and topology matrix. since all the elements on the main diagonal are equal. So, you will be seeing that all the things are equal here also you will be seeing that they will be equal ok. So, then consider the optimum filter with a fixed coefficient w_1 as which was illustrated in figure which will be shown here. So, we say $w^0 = 1$ and w_1 is the weight of the filter that is what shown in this figure we are assuming that. And if the given signals $x(n)$ and $d(n)$ have characteristics given by this that is expected value of $E[x^2(n)] = 1$ and then $E[x(n)x(n-1)] = 0.5$. and then expected value of our desired signal $E[d^2(n)] = 4$ and then the cross $E[d(n)x(n)] = -1$ and the this thing. what we call it as $E[x(n)x(n-1)] = 1$. Then what is the problem to solve? Find the minimum square error function eta based on the fixed coefficient vector. So, what are the coefficient vector we are going to have it.

So, in this case R has to be $\begin{bmatrix} r_{xx}(0) & r_{xx}(1) \\ r_{xx}(1) & r_{xx}(0) \end{bmatrix}$ So, which is given as one is 1 the other one is going to be 0.5 as you can see the thing. $x(n)$ and $x(n-1)$ is 0.5 are $E[x^2(n)] = 1$ substitute this in the as a matrix form.

So, this is $\begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$ and your cross correlation matrix as you will be seeing it what is given as your $r_{dx}(0)$, $r_{dx}(1)$ that is what we are going to have it. So, after substituting it is going to be -1 and then 1. And then $\xi = E[d^2(n)] - 2p^T w + w^T R w$ So, substitute all the values. So, we have been given $= 4 - 2[-1 \quad 1]\begin{bmatrix} 1 \\ w_1 \end{bmatrix} + [-1 \quad w_1]\begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ w_1 \end{bmatrix}$. So, now substitute the optimum the equation boils down to $w_1^2 - w_1 + 7$. How we are going to calculate optimum filter we call the name as w^0 basically. Minimizes the mean square error function this $\xi(n)$ that is by substituting $Rw^0 = p$ basically. So, we will say that it is expected value of sorry $\xi[d^2(n)]^{T^0}{}_{\min}$ what we have to compute. Thus, the optimum filter can be computed as $w^0 = R^{-1}p$ So, something is going to trigger in your mind. So, we have to calculate the inverse of a matrix in this case. So, you know the challenges will be facing in hardware implementation. To give an example for the optimum filter, so we are going to consider an FIR filter with two coefficients w_0 *and* w_1. The desired signal $d(n) = \sqrt{2}\sin(w_0 n), n \geq 0$ and the reference signal $x(n) = d(n-1)$ delayed function of it.

Find w^0 and ξ_{min} what it is been given. So, as we calculated with the previous example $r_{xx}(2) = E[x^2(n)] = E[d^2(n)] = 1, r_{xx}(1) = \cos(\omega_0), r_{xx}(2) = \cos(2\omega_0), r_{dx}(0) = r_{xx}(1)$, and $r_{dx}(1) = r_{xx}(2)$.. So, by substituting these values so we know that $w^0 = R^{-1}p$ this is nothing but $\begin{bmatrix} 1 & \cos(\omega_0) \\ \cos(\omega_0) & 1 \end{bmatrix}^{-1} \begin{bmatrix} \cos(\omega_0) \\ \cos(2\omega_0) \end{bmatrix}$.

and then you will substitute your other parameters. So, then by simplifying it. So, what you will be getting is ξ is going to be with this function minimum what you have to calculate. So, in practical applications what we call it as the computation of the optimum filter requires continuous estimation of our R and p when the signal is non stationary. So, in addition if the filter length L is very large. The dimension of the autocorrelation matrix that is $(L \times L)$ is large, thus the calculation of inverse matrix is going to come it becomes a bottleneck which requires intensive computation.

So, coming with the other example. So, if the length of the filter if it L is assumed as 2, the error surface forms a 3 dimensional space called an elliptic paraboloid. So, now, if we cut the paraboloid with planes above ξ_{min} that are parallel to $w_0 - w_1$ plane. We have obtained concentric ellipses of constant mean square errors values to give you a flavor of it this is how it will be when you cut into the thing you will be getting the concentric circles what you can see it. Then these ellipses are called the error contours. So, that is what was written there also and then we are going to consider an FIR filter with coefficient w_0 and w_1 and the reference signal $x(n)$ is the zero-mean white noise with unit variance.

So, we have derived the thing this is our desired signal $d(n) = b_0 x(n) + b_1 x(n-1)$. So, the coefficient b_0 and b_1 have been given as 0.3 and 0.5. So, you have to calculate the error surface at the error contours basically using this equation as the derived signal. Because, it is a second order with the previous example and this example combined actually your autocorrelation matrix is going to become $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and then your p vector is going to be b_0 and b_1 and we have this is the error function ξ what we are going to calculate.

So, put it in this equation and then substitute all the values then you will be combining $(b_0^2 + b_1^2) - 2b_0 w_0 - 2b_1 w_1 + w_0^2 + w_1^2$ and substitute b_0 =0.3 and $b_1 = 0.5$ then your $\xi = 0.34 - 0.6w_0 - w_1 + w_0^2 + w_1^2$. So, the error function is given as this fine and if you calculate the contour for that using Matlab function defining giving these are your $w0$ are the 2 coefficients. and then w other one coefficient what you will give it and error for 15 that is your 15 contours what you want to have it and then you calculate right plot the surface function. So, using the steepest decent algorithm what you will be. putting the thing ok. So, when you calculate using the MATLAB what is it descent method is an iterative recursive technique that is starts from some arbitrary initial weight vector $w(0)$ and it is going to descends to the bottom of the bowl.

So, somewhere here you will be selecting it and you will be moving on the error surface basically in the direction of the negative gradient what you are going to go down. So, that is estimated at that particular point and then go down. So, examples of the error surface and error contours for $L = 2$ what it is shown. So, we will consider the what is it $w(n + 1)$ is the future weight vector is going to be computed by this equation. How we are going to derive and then how we are going to calculate the steepest descent algorithm technique we will see in the next class.

So, we will be deriving the least mean squared error algorithm also using this technique. So, happy learning and thank you. We will meet you in the next class.