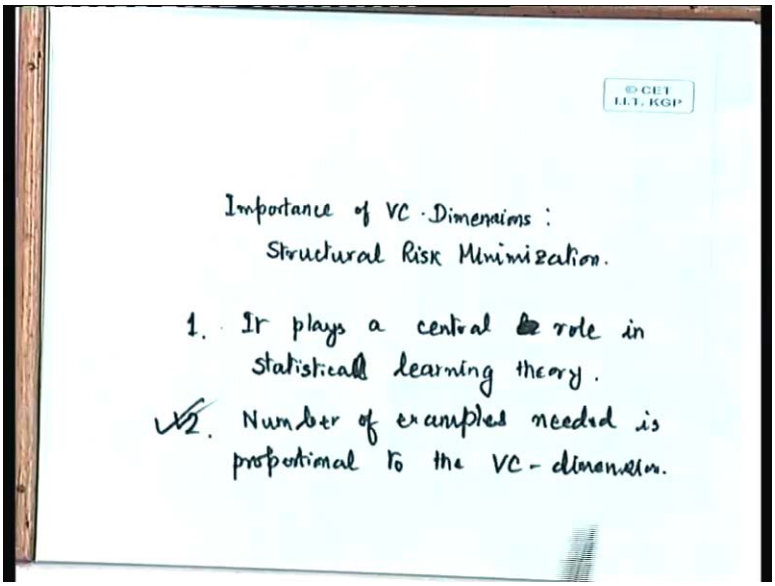
Neural Network and Applications
Prof. S. Sengupta
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture No - 11
Importance of VC Dimensions Structural Risk Minimization

We begin today's lecture, with the Importance of VC Dimensions and especially, we are going to talk about the Structural Risk Minimization.

(Refer Slide Time: 00:56)



So, I trust that in our last lecture the actual concepts of VC dimension has gone into your mind properly. Since, we had discussed with a few typical examples, although they were very much over simplified cases, because we could consider only up to two dimensions, but I think with the examples of the quadratic argument of signal functions and the linear argumental signal functions, I think that some of the doubts that you were having earlier could be cleared.

So, today we are going to talk mostly about the importance of VC dimensions and especially the thing is, that why do we need the study of this VC dimension because when, we are trying to consider a neural network that can accept a set of training patterns. Then, if the number of training patterns are fixed, then the question is that, what is the structure of the neural network that you are going to have or rather in terms of VC dimension, what is the VC dimension that we have to choose, so that the neural network gives the optimal performance.

And this optimal performance, actually should be in terms of its generalization capability because you know in a neural network, we are considering two types of error. Number one is the training error of course, it is important to have a very low training error, because without having a low training error you cannot think of achieving high generalization, a good generalization capability. But, even if the training error is low that does not necessarily guarantee, that the generalization capability of that network is good.

In other words, the generalization error characteristics or rather the test error probability, something that we were discussing yesterday that has to be minimized because, only then we can say that the neural network is optimized in its performance because, ultimately it is in the test environment that we are going to use the network. So, the thing is that by having an idea about the VC dimension, it is possible for us to find out that where exactly is that condition of optimality coming in.

The place of optimality is where the best kind of generalization capability exist means in terms of the error, there is some where a minima that we have to locate and VC dimension gives us some idea about that. Although one point should be kept in mind that let us not be under the impression, since we discussed few very simplified problems yesterday.

And there we could determine the VC dimension we showed that for the linear perceptrons, we could see that if, we have a dimension equal to two then VC dimension of that becomes equal to three if it is n, it becomes n plus 1. Let us not think that every time it will be possible for us to analytically calculate that what the exact VC dimension is. In fact, very often for the more involved neural networks which will, involve a number of neurons and which will involve feet forward connections between the neurons.

So, this especially multilayered neurons, it is not always possible for us to I mean. In fact, we should say, that in general it is not possible to analytically determine the VC dimension. Although, the idea of VC dimension is still very important because yesterday you remember that the relation that I had shown, which was in fact, suggested by Vapnik that the relation that exists between the R test, the R training and the expression that we had got in terms of the VC dimension.

VC dimension, then the number of training patterns, we had obtained that expression. In fact, we are going to present that once again today because most of our discussions for

today will start with that expression, but before that let us, list out that what the importance's of the VC dimensions are. Firstly, is that it plays central role in the statistical learning theory.

In fact, that is what I mean our original discussion was that we where, in fact, I mean we are still deliberating on the chapter related to the learning theory and towards end of that chapter, we discussed about the statistical learning theory, so this one plays central role in the statistical learning theory that is first importance of VC dimension. And the second factor is that the number of examples, needed to learn a class is proportional to the VC dimension of the class.

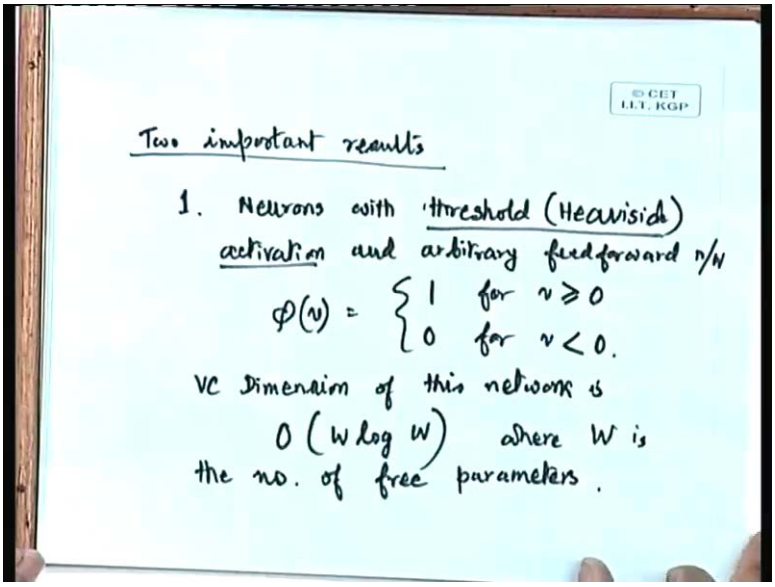
A number of examples needed, so I mean you know in this case the number of examples needed when I say number examples needed then I mean it is from the optimality consideration. Ideally zero error, but since we will not be able to achieve zero error at least we should going for the minimum error criteria, where the VC dimension plays a very important role. In fact, we are going to study about this aspect more.

And statistical learning theory I mean more than what we have discussed, I do not want to deliberate further because there are other topics related neural networks, which we need to cover in this lecture series. Now, in most of the cases as I was telling you, that the VC dimension can be determined in terms of the free parameters of the network. Like yesterday, you had seen that for the example of two dimensional cases and linear activations.

We had that it is equal to number of free parameters in that case it was two, two plus one in fact, it was three or to say that even if you take the case of a two input neural network, the number of free parameters are three because you have to include the bias. So, it is indeed equal to the number of free parameters, so I can say that, if W is the number of free parameters then it is equal to or it is of the order of W, for the simplest case it is equal to W, but talking in terms the order it is order of W.

I mean, why I am talking of the order is because, ultimately we are going to talk in terms of the bounds and they are for some typical neural networks, for some typical activation functions, we can determine that what those bounds are and in these case.

(Refer Slide Time: 09:36)

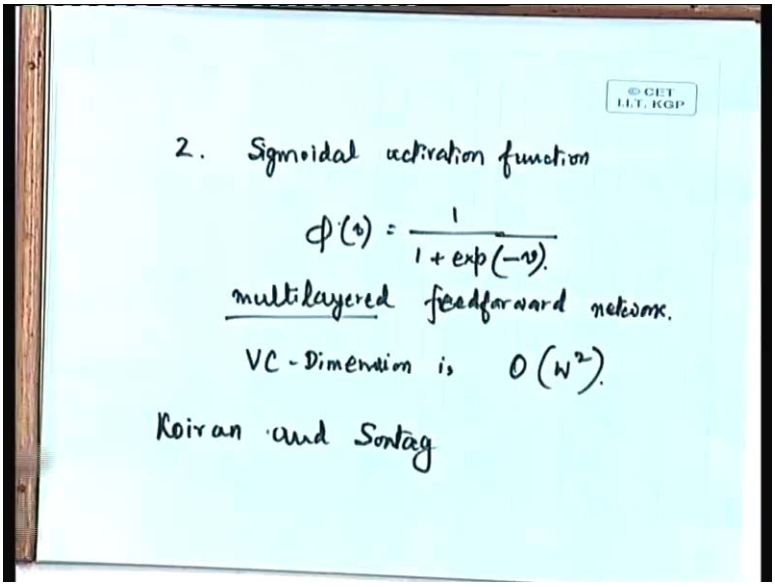


Let me cite two very important results, which has been reported by the researchers in literature and the two important results that must be stated is that, firstly is related to the neurons with threshold activation functions. So, for neurons with threshold activation, so threshold activation, we also describe that as the heavy side function, so as the heavy side activation which of course is of the form phi v is equal to 1, when v is greater than or equal to 0 and this is equal to 0 otherwise that is for v less than 0.

So, for such kind of neurons that is with the threshold activations, the VC dimension of this is I mean it has been shown analytically, the VC dimension of this network is given by order of W log W, where W is the number of free parameters. So, this is quite an important result that we can make use of so; that means, to say that with threshold activations it is of the order of W log W, so this in fact, happens to be better as compared to the linear case because in linear case, it was simply order of W.

Now, in this case, in fact, we have been considering threshold activation and also arbitrary feed forward network, so we are also considering a feed forward case, because this result is valid for the feed forward networks. And now another important result that we can say is when the activation function is sigmoidal.

(Refer Slide Time: 12:26)



So, with sigmoidal activation function with the activation function of this phi, I mean which we had already discussed, so many times earlier 1 plus exponential phi for minus a v you say or minus v you say, it is all same I mean for sigmoidal case. And for the case of multilayered feed forward network, although we have not explicitly discussed multilayered feed forward network, but I think you can guess that what I am talking of essentially, where we are cascading the layers of neurons.

Simply, I mean you begin with an input layer and then all those input layers will be connected to some neurons, which will be having activation functions and those outputs in turn will be connected to another layer. So, if I have got two such layers, in fact, including the input it will be three then it is a multilayered network, so more than two layers, I mean from the inputs stages multilayered.

And if it is going only in the forward direction without considering any feedback connections between the neuron means from the earlier layer to the, I mean from the next layer to the previous layer there, if such kind of feedback connection do not exists then we should call those kind of networks as it is a feed forward network and of multilayered types. So, for sigmoidal activation and with multilayered feed forward the VC dimension, this is very interesting VC dimension is order of W square.

So, this I mean; obviously, raises some interest, for linear activation we had W, for threshold we had order of W log W and with sigmoidal with multilayered we are having order of W square. In fact, the tricks are been done by the multilayered, once we have a

multilayered feed forward connection then the VC dimension order increases and this comes to our advantage because one thing is very clear that as we increase the VC dimension of the network, its classification power naturally increases.

So, definitely we should expect out of this, I mean we should expect a better performance, better classification performance out of a multilayered feed forward network as compared to the single perceptron or single layer neurons obviously. Now, another interesting aspect that must be thought of, in fact this result of the sigmoidal activation function, was pointed out by Koiran and Sontag they are the people who in fact, all these theories are quite recent. In fact, Vapniks theory itself is published in the year 1992 and some further theories are published in the year 1998.

And Koiran and Sontag also quite recent results over the past six or seven years I think, where Koiran and Sontag they arrived at this VC dimension bounds by considering the multilayered network in this way, that where they had shown two types in multilayered feed forward, they had shown two types of networks. Number one that they had showed, the threshold functions and then they had shown sigmoidal function.

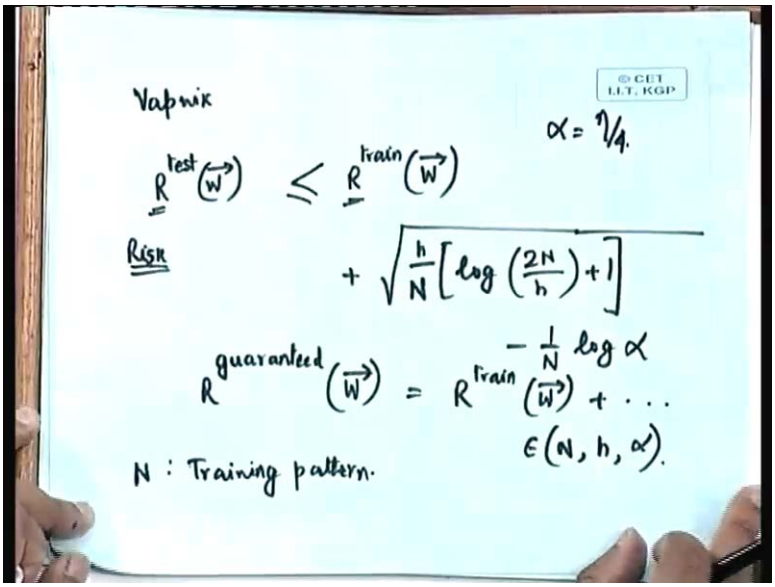
And then everything could be approximated, both threshold function as well as the sigmoidal function could be approximated using the linear activation because this also is pretty clear in your mind, that if you are having a linear network with very high values of synaptic weights. In that case what it leads to, what is the limit if the weights are increased, threshold functions, it leads to threshold function in the limit that when the weights are very high.

On the other hand when the weights are very small, then it could be approximated by the sigmoidal function or I mean to put it in a bit of other way, I mean for a sigmoidal activation with small synaptic weights could modeled as a linear and threshold functions could be modeled also as a linear, but with high synaptic weights. So, because the linear model can be used and we are using a cascaded version of that to form a multilayered networks.

So, that is why, instead of order of W we are getting order of W square I mean this is just to understand, this realization of order of W square in a very simplified way. So, the combination, so as we understand that the combination of two linears is giving us the order of W square and in fact, this is very interesting. Now, after talking about these two important results put into the order of the VC dimensions.

Let us go back, again to the result that we had discussed yesterday from Vapnik's theory. So, in fact we were discussing yesterday about the relation, that should exist or the inequality that should exist between R test and R train, remember.

(Refer Slide Time: 18:50)



In fact, yesterday as arguments I was putting alpha, where alpha vector was the free parameter vector, but I mean it was pointed out to me that will create some confusions because, so long we should be consistent in the nomenclature because, so long we are considering always the W vector as the free parameter vector. So, let us stick to the original nomenclature and instead of using the alpha as the vector of free parameter, we take the original notation of W as the vector.

So, W vector the free parameter vector and again R train also will be in terms of the W vector, in fact the relation that Vapnik had pointed out is that R test W should be less than or equal to square root of, in fact because I am using R over here, I do not want to use.

Student: ((Refer Time: 20:06))

No, here the here the free parameters are W only, we will not retain the alpha because here, since we are consistently talking about the free parameters as the W vector that is the weighed vectors and synaptic weighed vectors. So, it is the free parameter and we will talk in terms of the W only, since I mean that is what we all along understand by the free parameters, so no point bringing the alpha, I am just writing a corrected nomenclature expression, so please make a note of it.

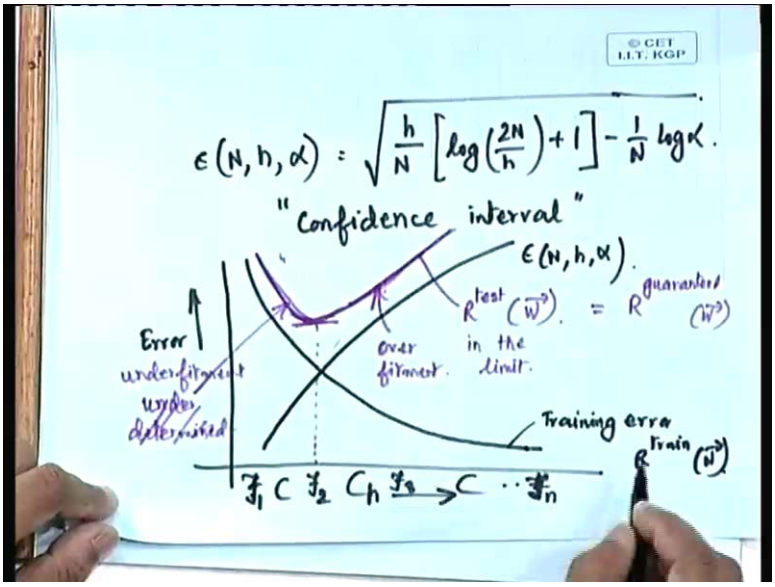
h by, in fact here instead of talking about the R, because R is the error probability or what we can call as the risk factor, so R stands basically for the risk and because R is used for these, we will not use R for the number of training patterns. So, instead we will use the number of training patterns as N, so N as the number of training patterns, so the expression that we are rewriting now is h by N I mean everything under root, h by N log of 2 N by h plus 1 the square bracket, the bracket closes here, minus 1 by N log.

In fact, I was yesterday telling it as eta by 4, again some confusion about our nomenclature because eta was the learning rate. So, long since, we have been always using eta to be the learning rate, so let us not make that confusion and I now substitute alpha is equal to eta by 4, so alpha in this case will just a scalar quantity, just a constant quantity which will be expressed as this eta by 4. So, there we will be writing as 1 by N log off alpha, so this is the expression that Vapnik has pointed out.

And In fact, if you look at it there are two terms in this expression, in fact I mean this expression R train plus this whole thing together, these two's when summed up gives us some kind of a guaranteed risk factor for the test cases I mean for the test patterns, the guaranteed risk function will be given by the summation of this. In fact, it say, it is less than or equal to, it will be better than that, but at least it can never exceed R train plus this expression.

So, that way this bound has got very significant purpose, so in fact we are going to call R guaranteed, so let us call it as R guaranteed as a function of W will be equal to R train as function of W plus this expression. In fact, I mean instead of writing this expression in terms of h which is the VC dimension, N the training pattern and alpha, so we are going to use a more simplified nomenclature, we are going to call this as epsilon, which is expressible as function of three things N, h and alpha, where epsilon N h alpha is going to be….

(Refer Slide Time: 24:02)



Is going to be this expression, that is h by N into log 2 N by h plus 1 minus 1 by N log off alpha, so this is what you are getting. So, now definitely define having defined R guaranteed this way R test is going to be less than or equal to R guaranteed that is what it means. Now, what is our variable, since we are investigating with the effects of VC dimension, let us now think of varying h. In fact, there are two terms here one is the training error, so the first term is training error and the second term this epsilon expression that we have got.

This is, in fact referred to as the confidence interval and why is it called as confidence because, if you add the confidence to the training error, some confidence measure to the training error then that tells you, that what is going to be your bound on the test error. So, at least if the confidence measure is known, you can confidently say that you tester is going to be less this much, so that is why the name confidence measure or confidence interval that has originated.

So, two terms the training error and confidence interval, in fact both of these are definitely functions of h. Now, R train in the expression for the training error we did not explicitly tell about any dependence on h, but you think about it, very surely R train also will depend upon h is in it very clear. You see you take a network, a two dimensional network let us say with two I mean synaptic weight connections or single neuron to synaptic weight connections, three free parameters, your VC dimension is equal to three for that.

And now you take another network, where you have three inputs, one biased, so four free parameters here the VC dimension is four. Now, which one will lead to lesser training error obviously the one, which is having more number of connections, so with h, with the increase of h definitely that leads to a lowering of the training error. In fact, if we try to plot the training error versus the h so on the x axis if we plot h, so this is increasing direction of h and the y axis we called as the error axis, where we are going to plot the error.

So, if we are plotting the first term that is the training error then the training error will be of this nature, something like an explanation, so this is the training error or the R train expression, R train as function of W. Now, this is the dependence of the training error on h, definitely we all understand that as h increases the training error definitely decreases. So, that leads to thinking that definitely, we should have as large an h as possible as per as our budget permits, we should keep on increasing the h, we should keep on increasing the dimension of the network.

And then we should have a better training error, but there is another factor we have to consider the generalization aspect. Now, just think that, what is going to be this expression, what is going to be the dependence of the epsilon on h, does it increase or does it decrease or unpredictable. Definitely, in this case N is much higher as compared to h that is bound to happen because you are considering N number of training patterns, so which will be much more then what the VC dimension tells you.

So, definitely in this case what happens is that with the increase of h, what happens is that this epsilon expression that monotonically increases, so with the increase of h epsilon monotonically increases, so the epsilon characteristic with h are rather the confidence interval should monotonically rise in the case of training error it was monotonically decreasing, where as in the case of confidence interval it should be monotonically increasing of this nature.

Now, comes the vital question because not that we are very seriously bothered about the training error, not that we must reach exactly at this spot or at h training to infinity, we are ultimately going to use the network, where the test error is going to be minimized. Now, what is the test error or what is the R guaranteed, R guaranteed will be R train plus this epsilon, so this is epsilon N h alpha, so the addition of these two things and what

kind of a characteristic could you expect, for this R train plus epsilon that is the guaranteed risk.

Constant no, do not do not think that this epsilon is exactly balancing the training error, it is not like in equalizer that it exactly balances. In fact, if you calculate the characteristics yes, there will be a minimum, in fact the characteristic of the R test, the risk on the test would be something of this nature and at a point it reaches the minimum and then again it rises, so this is something that we have to note, so this is what, this is the R guaranteed or the limiting factor on R test.

Now, R test is going to be better than this, but at least we know that in the worst case R test is going to be like this, so this is R test W under the worst case, yes this is R guaranteed. So, in the limit I say that this R test is equal to R guaranteed, yes I mean this is the limiting case of R test or rather this is the R guaranteed. So, this is R test in the limit, so the actual R test may be somewhere here it cannot go to this area, it has to be lower than this.

But, definitely when we consider the bound, we have to be very careful about this characteristic that at least this is the place, where we find that there is a minima and this is the minima that we should look for why, because that gives the best generalization capability. So, the corresponding value of h should be somewhat the optimum value of the VC dimension, which we must consider.

If, we consider unnecessarily a larger VC dimension we may feel happy that the training error is much less, but in the process of making the training error better, we will be rather making the characteristic more vulnerable to the training error, I mean to the test error. So, in fact this characteristic has got two very important regions to note, in fact in this region somebody can try to operate it in this region, naturally not very advisable because here you are getting quite high training errors.

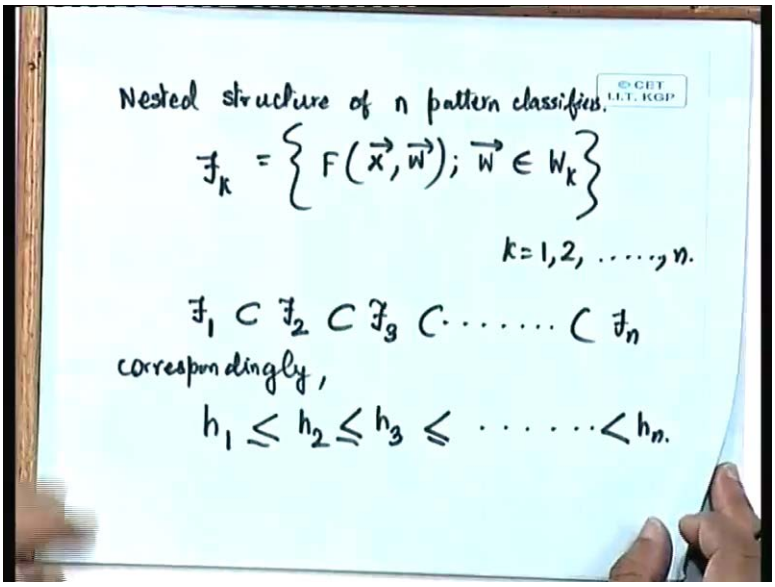
You are also getting quite high test errors, but the test error is decreasing, but you are yet to reach the minima. In fact, this is a region, which we should call as the under determined, so this is under determined and beyond this minimum, this leads I mean let us not call under determined, the better term to say will be under fitment. So, here this is under fitment problem is happening in this region, because as is to say the network is inadequate.

The h is very low and h very low means, that the network is inadequate to learn this things or rather to say that we have not exploited the full capabilities of the network, where as this one leads to the problem of over fitment. So, this is over fitment problem, this one is under fitment problem and here, we have the point of minimum, so we must look for the point of minimum and how exactly are we going to do it, so that something that we should consider.

Now, from the practical aspect really speaking one has to experiment with it, if you ask me the question that from a practical point of view how am I going to say that this is going to be the best type of h. So, in fact we have to experiment upon and then come to a conclusion, so in order to just make an expression of this, let us say that we have got a nested structure of classifiers, by nested structure of classifiers I mean to say that I am going to consider a classifier that can classify everything.

Then, I am going to consider another classifier, which is contained in that classifier, but cannot classify everything that the earlier one could do. And then another classifier, which is contained in its previous one, like that I am considering a nested class of classifiers, so I am going to just represent this nested class of classifier as….

(Refer Slide Time: 36:37)

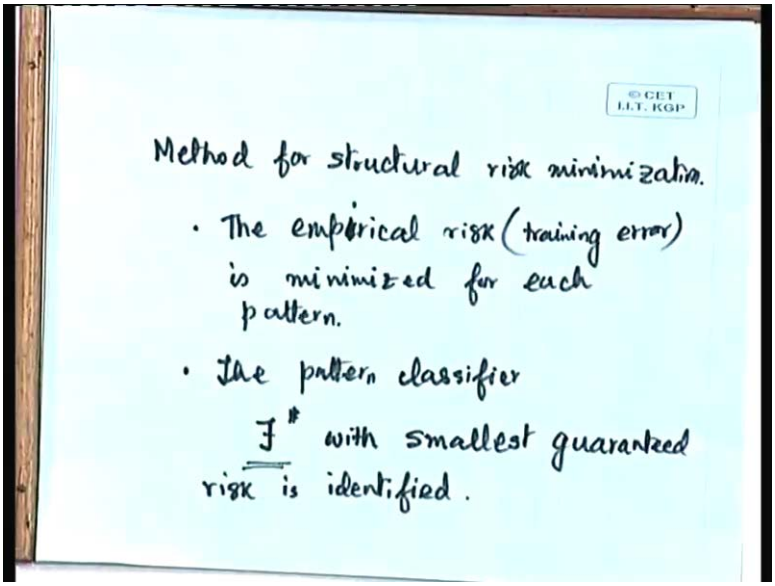


Set F K, where the set will be defined as the set of F X vector W vector, X vector W vector means, it is neural network realization X being the X vector being the input, W vector being its free parameters and in this case the W s that will lie in the region WK. And here, K will be 1, 2, up to n, so; that means, to say that we are considering a nested

structure of N pattern classifiers and how we define the nesting, the nesting is done such that F 1 is contained in F 2 that is contained in F 3 and So, on and the last class is F n, which includes everything.

And if we are having, this sort of nested pattern classifier then correspondingly we must have h 1 which is less than or equal to h 2 less than or equal to h 3 up to h n. Now, what is the methodology that we have to follow for the structural risk minimization, so if we consider a nested structure of pattern classifiers like this.

(Refer Slide Time: 38:41)



Then, the method for structural risk minimization that we should follow is that the empirical risk, the empirical risk means the training error I mean different books and literature they call it sometimes as empirical risk, sometimes they call as training error because training is after all an empirical data. So, that is why this is empirical risk or training error, so the empirical risk is minimized for each pattern.

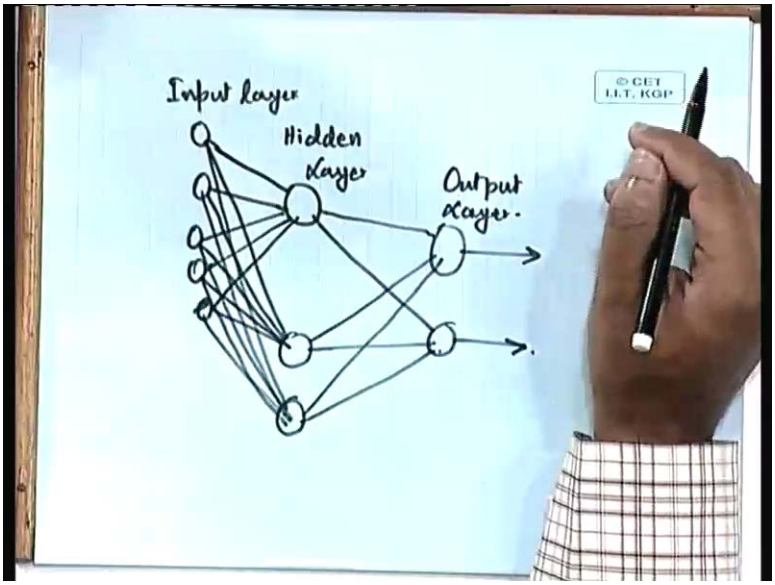
And then you determine the pattern classifier, out of all these pattern classifiers F 1 F 2 up to F n, you determine that which pattern classifier gives you the smallest guaranteed risk, smallest guaranteed risk means this one. So, in fact you are in this case I should say that could be that, the set F 1 will be here, then F 1 will be contained in F 2 will be contained, in F 3 etc, etc F n may lie I mean somewhere over here, so there is point of minimum and let us call that us F star.

So, the pattern classifier F star with smallest guaranteed risk, with smallest guaranteed risk is identified and in fact that is the best solution that we are going to have, so we are

going to consider that pattern classifier which is having F star, F star will correspond to this point of minimum. So, this principal actually is called as the structural risk minimization and this is where, you find that the VC dimension plays a very crucial role.

Now, practically how we are going to do, that is the question that we should address. See normally kind of analysis or the kind of this structural risk minimization is carried out for the case of multilayered feed forward networks, which we will be learning in more details later on. Now, in a multilayered network, typically the configuration is like this that from the set of inputs.

(Refer Slide Time: 42:05)



Supposing this is one layer, let us say that we form an input layer and then so this is connected to one layer and then these will be in turn connected to an output layer, may be that in the output layer we consider these neurons. So, these neurons will form the hidden layer and this will be the output layer and this will be the input layer. Now, naturally, we normally do not want to touch the output layer, because output layer is finally going to gives us the number of classes.

So, we in fact fiddle with the hidden layer, because hidden layer is not playing any direct role in generating the output, it is playing a crucial role, but somewhat indirect. Even if we eliminate one hidden layer node it should be possible to still have a multilayered, feed forward connection, but whether that will have a better generalization or watts generalization is something that we should investigate. So, we can have let us say a large

number of hidden layer, neurons to begin with and large number of hidden neurons means that the VC dimension also will be correspondingly higher.

And we keep on eliminating each hidden layer neuron one by one decreasing the VC dimension, at one point we will be finding that the training error is minimized and that is the point that we have to note, beyond that if we keep on reducing hidden layer notes then it will again increase. So that way, we are going to find the point of optimality, so that is one practical means of selecting the structure, so this VC dimension has got a very big impact on the structural design of the neural networks.

I mean is there any question that I can answer related to the VC dimension and structural risk minimization, so we will be then meeting again and then we will be discussing about the single layer perceptrons and in fact, we already have some elementary idea about the single layer perceptrons, but we will go more in terms of its convergence aspect because that is one point which is very important.

So, we will be beginning with the single layer perceptron chapter in the coming lecture.

Thank you very much.