Course Name: Machine Learning and Deep learning - Fundamentals and Applications

Professor Name: Prof. M. K. Bhuyan

Department Name: Electronics and Electrical Engineering

Institute Name: Indian Institute of Technology, Guwahati

Week-10

Lecture-34

Welcome to NPTEL online course on machine learning and deep learning fundamentals and applications. Up till now I have been discussing about the concept of statistical machine learning techniques. Now I will be explaining the principles of artificial neural networks and how artificial neural networks can be employed for pattern classification. In my discussion, I will be considering both supervised and unsupervised artificial neural networks. And in the beginning I will be discussing the fundamental concept of the artificial neural networks. And after this I will be giving some examples how artificial neural networks can be employed for pattern classification.

So now let us begin this class the concept of the artificial neural networks. So the objective is to build an electrical system which is capable of performing tasks through reasoning as much as a human. So that is the motivation or that is the objective of the artificial neural network.

So which can be employed for pattern classification, pattern recognition and also for predictions.

So there are many applications of artificial neural network. So I will be explaining the concept of pattern recognition and how we can employ the artificial neural networks for pattern classification, pattern recognition. So that concept I am going to explain. So before explaining the concept of the artificial neural network, I am showing the structure of the biological nervous system because the artificial neural network the inspiration is coming from the biological nervous system. So the structure is very similar.

So in the figure I have shown the concept of the biological nervous system. So here you can see the cell body is available. So then right collect signal from the cell body and all the signals are sum up and if the sum signal is greater than a particular threshold then I will be getting a signal and it is transmitted via axon. So I have shown axon here. So I am repeating this concept the concept of the biological nervous system.

The signals are collected by dendrites and all the signals are sum up and if the sum signal is greater than a particular threshold then a particular signal will be transmitted via axon. So that is the concept of the biological nervous system.

So same thing here I am showing in this figure also. So here you can see the signals are collected by the dendrites. So the signals are collected by the dendrites.

So these are the dendrites and if the sum signal is greater than a particular value then the signal is transmitted via axon. So in the second figure in this figure and that concept I have shown.

So all the signals are collected by the dendrites and these signals are sum up. So here you can see the summation is taking place here and if the sum signal is greater than a particular threshold then the signal is transmitted via axon.

So this is the concept of the biological nervous system.

So same thing can be implemented in the artificial neural network. So let us discuss about the artificial neural network.

So the same model that is the model of the artificial neural network. So in the figure I have shown the signals S_1, S_2, S_3, S_4, S_5. So these are the signals the input signals.

The signal is multiplied with the weight the weight is w_1, w_2, w_3, w_4, w_5. So these are the weights.

So we are multiplying the signal value with the weight. So S_1 is multiplied with w_1, S_2 is multiplied with w_2 like this we are multiplying the signals. So we are getting the sum value.

So we are getting the sum value and this sum value is compared with a threshold. So threshold is determined by the squashing function. So $\sigma(.)$ is the squashing function and if the sum signal is greater than a particular threshold then I will be getting the signal S that is transmitted. So the sum signal is calculated like this. This is nothing but the weighted inputs.

The input signal is weighted by the weights the weights are w_1, w_2, w_3 these are the weights. So this is nothing but the weighted inputs and I can determine the sum the sum signal I can determine and we are considering the squashing function. So with the help of the squashing function I can compare the sum signal with this squashing function and if the sum signal is greater than a particular threshold then I will be getting the signal the signal is the S, S is the signal. So this is the concept of the artificial neural network. So the inspirations from biological nervous system.

So this is one example of the squashing function. So this is called the sigmoid function we are considering the sigmoid function as a squashing function $\sigma(x) = \frac{1}{1+e^{-x}}$. So suppose if I consider the sum value is here because sum value is not greater than the threshold then I will be getting the output 0. But if I consider the sum signal is here suppose in the position number 2 and suppose the position number 1. So if I consider the sum signal in the position number 2.

So that is greater or that is equal to the threshold value then I will be getting the signal the signal in the output. So the sum signal is compared with the squashing function and if it is greater than or equal to the value of the squashing function then I will be getting the output signal in the artificial neural network. So that is the concept of the squashing function. So in case of the artificial neural networks we can consider two types of networks one is the fixed network another one is the adaptive network. So already I have shown in the previous slide the weights weights of the artificial neural networks.

So if you remember this one in my previous slide I have shown the signals are multiplied with the weights S_1 is multiplied with w_1 S_2 is multiplied with w_2 S_3 is multiplied with w_3. So these are multiplied and we are getting the signal the sum signal we are getting. So in case of the fixed network these weights are fixed. So networks in which the weights cannot be changed that means the $\frac{dw}{dt} = 0$. But in case of the adaptive networks we can change the weights of the artificial neural networks that means $\frac{dw}{dt} \neq 0$ in case of the adaptive network.

In case of the fixed network $\frac{dw}{dt} = 0$. So mainly I will be considering the concept of the adaptive network because during the training of the artificial neural networks I have to adjust the weights of the artificial neural networks. So that is the training actually. So during the training we are changing the weights of the artificial neural networks. So every neural network has knowledge which is contained in the values of the connection weights that means the knowledge of the artificial neural network is available in the weights.

So during the training I can modify the weights that means the modifying the knowledge stored in the network. So that is the concept of the learning. So during the training or during the learning I can modify the weights of the artificial neural networks and that is nothing but modifying the knowledge stored in the network. So the information is stored in the weight matrix. So I have a matrix that is the weight matrix W of the artificial neural network and learning is the determination of the weights.

So that is the concept of the training. So during the training we can determine the weights of the artificial neural networks. So like in case of the machine learning that I discussed about the concept of the supervised learning and the unsupervised learning in this artificial neural network also I have supervised learning and also unsupervised learning. In case of the supervised learning the concept is like this. Suppose if I consider one simple artificial neural network so input is suppose S_1 another input is S_2 and suppose the weights w_1 and w_2 I am considering and we are getting the output of the network.

So I can say this is the desired output we have this information the desired output. So this information is available corresponding to a particular training sample we know what is the desired output and in the network I can determine what is the actual output. The difference between the desired output and the actual output that is called the error. So error is nothing but the desired output minus actual output. So the error we can determine like this and what is the response of the what is the output of the network.

So output of the network I can determine like this that is $S_1 w_1 + S_2 w_2$. So like this I can determine the output of the network. So we know what is the desired output corresponding to a particular training sample and also we can determine the actual output of the neural network. The difference between the desired output and the actual output that is called the error. The error is back propagated to the input.

So now error is back propagated to the input to adjust the weights of the artificial neural network. So I am adjusting the weights of the artificial neural network so that I can reduce the error and this is called the training of the artificial neural network. So we can apply some techniques like LMS convergence technique we can apply or the gradient descent algorithm that algorithm also we can apply for determining the determining the weights of the artificial neural network.

So the objective is to minimize the error and error is nothing but the difference between the desired output and the actual output. So output of the network we can determine like this.

So this is about the supervised learning. In case of the unsupervised learning that means no external teacher is available that concept already I have explained in my previous classes. Now in this case it is nothing but the grouping of the feature vectors. So based on the similarity I can do the groupings and this supervised learning is performed offline because we have to do the training that is during the training we are considering the offline training but the unsupervised learning is performed online. So that means in the online process we are grouping the feature vectors based on the similarity.

So this is the concept of the unsupervised learning that means the class information is not available but based on the similarity between the feature vectors we can group the feature vectors and we can employ the artificial neural networks. So some examples like the competitive learning. So or another example is the Kohonen neural network. So these are the examples of unsupervised learning. So unsupervised artificial neural networks.

So these concepts I will be explaining later on. So in this case I am showing one simple artificial neural network and I am showing the activation function that is the sigmoid activation function. So I am showing the inputs the inputs are -0.06 -2.5 and 1.4 and you can see the connecting weights. So the weights are w1 w2 w3 so these are the connecting weights and we can determine the x the signal x is the sum signal so we can determine the signal x we can determine. So how to determine the signal x you can see here x is it is the multiplication of the input signal with the corresponding weights x is nothing but the multiplication of the input signal with the corresponding weights that is the weighted inputs. So minus 0.06 is multiplied with the weight weight is 2.7 2.5 is multiplied with -8.6 1.4 is multiplied with a 0.002 and corresponding to this we can determine the x the x we can determine. And after this we can determine the response of the neural network that is we have to consider the activation function. So that is the sigmoid activation function we can consider. The x is compared with the sigmoid function and based on this I can get the output so this is the activation function that is the squashing function and that is the sigmoid activation function.

So I can get the output of the network based on this comparison. So x is compared with the sigmoid function and if the x is greater than or equal to a particular threshold that is defined by the sigmoid function then I will be getting the output of the artificial neural network. So now let us discuss the concept of the training how actually we do the training in case of the artificial neural networks. In this case I am showing some training samples and I am showing the outputs of two classes. So we are considering two classes and the outputs corresponding to these two classes.

So for the first class the output is 0 and for the second class the output is 1. So two classes we are considering. So now how actually we do the training. So let us first consider first input the first input is 1.4, 2.7 and 1.9. So corresponding to this 1.4, 2.7 and 1.9 that is the first input the output should be 0 corresponding to that class, class 1 or class 2. So for class 1 the output is 0 for class 2 suppose it is 1.

So output is 0 corresponding to the input 1.4, 2.7 and 1.9. So first I have to initialize with random weights. So that is the first principle. So I have to select the random weights and after this I am applying the inputs the inputs are 1.4, 2.7 and 1.9 and output is the desired output is the 0 the desired output is the 0. But in this case we are getting the actual output that is 0.8. So 0.8 is the actual output. So this is the actual output and what is the desired output the desired output is 0. So 0 is the desired output and the difference between the actual output and the desired output. So difference between these two is the error. So already I mentioned the error is back propagated to the input to adjust the weights of the artificial neural network to reduce the error. This error is back propagated to adjust the weights of the artificial neural network.

So I have shown that all the connecting weights and this error is back propagated to the input to adjust the weights of the artificial neural network so that the error can be minimized.

So this is corresponding to the input 1.4, 2.7 and 1.9. So let us consider the second input 6.4, 2.8 and 1.7. So corresponding to the second input the actual output we have to determine and the desired output is 1.

So the desired output is already mentioned it is 1. But we are getting the actual output that is 0.9 and the desired output is 1. So error we can determine so error is minus 0.1 and that is the difference between the desired output and the actual output. And this error is back propagated to the input so that we can adjust the weights of the artificial neural networks.

So we are doing the adjustment of the weights of the artificial neural networks. And this is the process of the learning of the weights that means we are doing the training training of the artificial neural networks. This is the concept of the supervised learning which is performed offline. So we have to do this process repeatedly and until some convergence condition is not satisfied. And based on this we can adjust the weights of the artificial neural networks.

And this is nothing but the training or the learning of the supervised artificial neural networks. So this pictorially this training process I can show like this. So here you can see I am showing two classes the samples of two classes one is the blue class that is the samples corresponding to one class that is a blue samples the blue colored samples and another one is the yellow colored samples corresponding to the second class. So if I consider random weights corresponding to the random weights this is the non-linear decision boundary between the classes. So, this is the non-linear decision boundary between the classes corresponding to the random weights.

After this I am doing the training that means I am adjusting the weights of the artificial neural network and corresponding to this you can see the position of the decision boundary between the classes. So this is the training that means I am adjusting the weights of the artificial neural network and the corresponding decision boundary you can see. So like this I have to do the iterations and you can see the corresponding decision boundaries between the classes like this. And finally I am getting this decision boundary between these two classes. One is the yellow colored samples corresponding to a particular class and the blue colored samples corresponding to the another class.

So this is the non-linear decision boundary I am getting because of the training of the artificial neural network. So in this case we are considering f(x) is a non-linear function that is the sigmoid function we are considering that is the squashing function. So what is the importance of the non-linear squashing function? So if I consider a network with one hidden layer can learn perfectly any classification problem. As per the theory a network with one hidden layer can learn perfectly any classification problems. So now this fxy actually we are considering non-linear during the training you can see here that we have obtained a non-linear decision boundary between the classes.

Suppose the squashing function is linear then in this case I can only draw the linear decision boundary between the classes if f(x) is linear that is the activation function is linear. But we are considering non-linear activation function and because of this we are getting non-linear decision boundary between the classes. So this neural network use non-linear activation function that is the fx and we can consider complex decision boundary between the classes. But in case of the support vector machine I have explained in my previous class that we can obtain the linear decision boundary between the classes. In support vector machine what we do first we transform the input data so that they will be linearly separable.

So now I will discuss the concept of the artificial neural network which can be employed for pattern classification. So we can employ artificial neural networks for pattern classification. So let us move to the next slide. So the ANN for pattern classification. So the problem is suppose I have two classes there is a two class pattern classification problem and these are the samples corresponding to the class 1 and suppose I have another class.

So these are the samples corresponding to the another class the class is class 2 and we have a decision boundary between this class. So this is the decision boundary. So for this two class problem I have to design the artificial neural network. So corresponding to this decision boundary I can write the equation of the decision boundary something like this $w_0 + w_1 x_1 + w_2 x_2 = 0$.

So this is the equation of the decision boundary. So the problem is the feature vector x is assigned to the class 1 and corresponding to this the response of the network response of the network is positive. That means the output should be +1 and x is assigned to the class

2. So corresponding to this the response of the network response is negative. So that means the output should be -1.

So that means the output is either + or -. So maybe we can consider that a step activation function in this case. Step activation function may be considered because output is plus 1 and minus 1. So we may consider the step activation function. So corresponding to this problem this decision boundary already I have mentioned the decision boundary is the equation of the decision boundary is $w_0 + w_1x_1 + w_2x_2 = 0$.

So this is the equation of the decision boundary. So if $w \neq 0$ then I can write this equation $x_2 = -\frac{w_1}{w_2}x_1 + \frac{w_0}{w_2}$. So this is the equation of the decision boundary. So it is very similar to y = mx + c. So in case of the artificial neural network what actually we considered the weighted sum of the inputs is calculated by the output node.

So this is the first step. So the weighted sum of the inputs is calculated by the output node. After this the output node then compares the weighted sum to a particular threshold and a classification decision can be taken based on this. So I am repeating this. The weighted sum of the inputs is calculated by the output node and the output node then compares the weighted sum to a particular threshold and a classification decision can be taken based on this.

So corresponding to this I can draw the neural network. So the neural network will be suppose very simple network. It has three inputs. So we are considering one bias input. So this is a bias input and corresponding weight is w_0. One input is x_1 that is the weight is w_1 and another input is x_2 and the corresponding weight is w_2 and the output is suppose y.

y is the output. So corresponding to this neural network the response will be either +1 or -1. So we may consider a step activation function for this. So the condition is if $w_0 + w_1x_1 + w_2x_2 > 0$ then I will be getting the positive response. Otherwise I will be getting the negative response.

So based on this I can do the classification. So in one case I will be getting the positive response in another case I will be getting the negative response. So this weights w_0 w_1 w_2 so all these weights I can determine during the training process. Now what is the importance of the bias that point I am going to explain. So I am here considering one bias input if you see the network.

So this is the bias input. So what is the importance of the bias that I will explain after some time. So this is the simple network for this problem. The problem is the two class pattern classification problem. So this can be also implemented by another equation.

So in this case we are considering the equation $w_0 + w_1x_1 + w_2x_2 = 0$. So we may consider another equation. So let me move to the next slide. So maybe we can consider this equation $x_1w_1 + x_2w_2 = T$. So in this case the T is the threshold so this equation I can consider and if $w_2 \neq 0$ then $x_2 = -\frac{w_1}{w_2}x_1 + \frac{T}{w_2}$.

So this is the equation. So how to take the classification decision if $x_1w_1 + x_2w_2 > T$ then corresponding to this I will be getting the positive response and if $x_1w_1 + x_2w_2 < T$ then I will be getting the negative response. So we can also consider the positive response and the negative response corresponding to this equation. Equation is the $x_1w_1 + x_2w_2 = T$. Now what is the importance of the bias input so I can explain now. So in my previous class I have explained the activation function that is the sigmoid activation function $f(x) = \frac{1}{1+e^{-x}}$.

So this is the expression for the sigmoid activation function and activation function is something like this. This is the sigmoid activation function. So now let us consider a simple network and first I will explain what is the importance of the bias for this. So suppose this simple network I am considering only one input x is the input and weight is suppose w_0 and output is in this case the sigmoid function is $sig(x.w_0)$ and after this I am considering the sigmoid function.

So this is the one case. In the second case what we are considering we are considering the network something like this. So xw_0 and we are considering the bias input. So this bias input is 1, 1 is the bias input and corresponding to this the network output will be $sig(w_0x + w_1)$. So this is the output $sig(w_0x + w_1)$ is the output.

This is the output corresponding to the second network. Suppose I need this output so output suppose should be 0 I need this output corresponding to x=2. So what I can consider in this original f(x) that is the sigmoid function corresponding to x=2 the output is not 0 output is 1 because this value is 1 so this value is 1. So corresponding to x=2 output is not 0 in this case but my requirement is corresponding to x=2 the output should be 0. So this is my requirement. So for this what I can do I can shift the activation function maybe this side I can just shifting the activation function.

So if I do the shifting then corresponding to x=2 I may get output 0. So suppose this is 2 so corresponding to this 2 I am getting the output 0 but in this case if I consider it is 2 then I may not get 0 it is 1. So I can shift the activation function so I can shift the activation function maybe this side or maybe this side towards left or the towards right and this shifting can be done based on the weight the weight is w_1 that is the bias weight. Suppose

$w_1 = -5$ I can shift the activation function towards right or whenever we need to shift the activation function towards left I can do based on the weight w_1. So that is the bias weight so w_1 is the bias weight so based on the bias weight I can shift the activation function towards right or towards left.

So that is the importance of the bias input. So if you consider the previous example you can see I have shown the decision boundary so based on the bias I can shift the decision boundary towards a particular class but the orientation I cannot change. So that is the importance of the bias and here I have shown based on the bias I can shift the activation function towards right or towards left and corresponding to this corresponding to a particular input I can get the desired output. So that is the objective of the bias terminal. So what actually we have considered we have determined the sum of the network $sum = w_0 + \sum_{i=0}^{n} x_i w_i$.

So weighted sum of the inputs we have considered. So this is actually which reference to the previous network. So suppose I have n number of inputs x_1 up to x_n and we have a bias input the bias input is 1 and this weight is w_0 and this is w_1 this is w_n and this is the output I am getting y. So corresponding to this one we have considered the two class pattern classification problem. So already I have shown these are the samples corresponding to the class 1 and these are the samples corresponding to the class 2.

So this is the decision boundary between the classes decision boundary between the classes. So this is actually I can consider as region R_1 and this region I can consider as region R_2. So if the $Sum > 0$ then my region will be R_1 that is actually the corresponding to the class 1 and the response of the network should be positive that is the positive response that is the 1 and if the $Sum < 0$ the corresponding region will be R_2 and the response of the network will be negative corresponding to the class 2. So response of the network should be -1 that is the response of the network. So already I told you that is we are considering the weighted sum of the inputs. So we are calculating the weighted sum of the inputs and that is calculated by the output node and the output node compares the weighted sum to a particular threshold and the pattern classification decision is taken.

So this is the fundamental concept of artificial neural network and how it can be employed for pattern classification. Suppose I am considering another example so let us move to the next slide. So we are considering some logical operations. So first suppose I am considering the AND operation AND logical operation. So corresponding to this AND logical operations I have two inputs one is x_1 another one is x_2 and y is the output.

So if x_1 is 0 x_2 is 0 the output is 0 this is the AND logic. If x_1 is 0 x_2 is 1 the output is 0 if it is 1 it is 0 the output is 0 and if both are 1 and 1 the output is 1. So this is a simple pattern classification problem. So maybe this 0 0 0 that corresponds to the class 1 and this corresponds to class 2. So it is a two class pattern classification problem. So in the feature space how to plot this so suppose if I consider this is a feature space so you see the first we are considering these two x_1 and x_2 and corresponding to 0 and 0 I have 0.

So these are inputs 0 0 0 1 1 0 and 1 1. So these are my inputs so this is 0 0 this is 1 0 this is 0 1 and this is 1 1. So what is the decision boundary between these classes. So the decision boundary will be something like this because this belongs to class 2 and all these three inputs corresponding to these three inputs the class is 1 the class 1.

So this is a class 1. So corresponding to this I have to design the neural network. So how to design the neural network. So I can consider the same equation $w_1x_1 + w_2x_2 + w_0 = 0$ this equation we are considering corresponding to the decision boundary that already I have explained and suppose I am selecting $w_1 = w_2 = 1$. So suppose it is 1 and $w_0 = -1.5$ these I am considering. So corresponding to this my network I can draw the network. So this is the network. So bias is 1 the bias input is 1 and corresponding weight is 1.5 this is the output and one input is x_1 and the corresponding weight is 1 this is x_2 the corresponding weight is 1. So I am getting y.

So this is the network for the AND logic. So this is the network for the AND classification problem. So now we are considering another problem logical problem that is the OR logic. So move to the next slide. So suppose if I consider the OR logic so again we have two inputs x_1 x_2 and y so you know if it is 0 and 0 output is 0 if it is 0 and 1 output is 1 if it is 1 and 0 it is 1 and 1 and 1 it is 1.

So this is the OR logic. So corresponding to this OR logic again it is a two class problem. So this is corresponding to the class 1 and these outputs corresponding to the class 2. So in the feature space I can show the feature space. So this x_1 and x_2 and I have the inputs so 0 0 0 1 1 0 and 1 1. So these are the points and what is the decision boundary the decision boundary I can draw like this.

This is a decision boundary. So if I consider the same if I consider this input this input and this input three inputs that corresponds to the class 2 and this input that corresponds to class 1. So it is a two class classification problem and we have a linear decision boundary between the classes. So this is the linear decision boundary that is the equation is $w_1x_1 + w_2x_2 + w_0 = 0$. So it is a two class problem and it is a linearly separable classes.

So it is a very simple problem and the samples are linearly separable. So linearly separable classification problem. Corresponding to this also I can design the artificial neural network.

Now come to another logic that is the exclusive OR XOR. So corresponding to this logic the input is again x_1 x_2 y. So if you see the the exclusive OR logic so it is 0 0 is 0 0 0 1 is 1 1 0 is 1 and 1 and 1 is 0.

So this will be in one class the class 1 and these two will be in class 2. So this is the XOR logic actually it is a comparator if it is you can see the input is 0 0 the output is 0 but when there is a difference between these two input one input is 0 another one is 1 or it is one input is 1 and another input is 0 then corresponding output will be 1 that is the comparator. So corresponding to this XOR logic it is not possible to simply draw the decision boundary between the classes. So because it is not a linearly separable classification problem so if I draw the if I draw the this feature space in the feature space x_1 x_2 like this so this point is 0 0 already I told you it is 0 1 1 0 and this one so it is it is not a linearly separable pattern classification problem. So the decision boundary will be something like this I can show the decision boundary so one boundary is not sufficient I have to draw these two decision boundaries. So if you see the samples or if you see the inputs this input and this input that is actually the class 1 and if you see the inputs this input and this input that corresponds to the class 2 that corresponds to the class 2.

So it is not a linearly separable pattern classification problem because you have to consider two decision boundaries in the AND and the OR logic we are having the linear decision boundary between the classes. So in the AND or OR logic I have shown the linear decision boundary between the samples of two classes but in this case it is not possible to draw a single decision boundary between the samples of two classes that is it is a non-linearly separable pattern classification problem. So what is actually this problem so for this actually we need the hidden layers in case of the AND logic or in case of the OR logic we have not considered the hidden layer we have only one input layer and the output node but in case of this problem that is the xor logic that is the exclusive that is the exclusive OR logic it is a non-linearly separable pattern classification problem. So for this we need hidden layers.

So a two layer network can classify data samples into two classes which are separated by a hyperplane. However a network having three layers is required when the problem is to classify samples into two decision regions where one class is convex and another class is the complement of the first class. So that means we need the hidden layers for this complex problem for simple problems which are linearly separable we may consider only input layer and the output layer but in this case the problem is complex because it is not linearly separable we have to consider a hidden layer between the input layer and the output layer and in this case the problem is we have to classify samples into two decision regions where one class is convex and another class is the complement of the first class. So that already I have shown in the previous figure for the xor logic. So the convex set can be approximated by the intersection of a finite number of the half planes.

The nodes in layer 1 can determine whether a particular sample lies in each of the half planes corresponding to the convex region. So that concept I am going to explain in my next slide corresponding to the xor logic. Subsequently the layer 2 of the network performs a logical and operation. So this AND operation is also important. So why it is important I will explain in my next slide.

So the concept is it is a complex problem and that is a non-linearly separable classification problem. So we have to consider hidden layers between input layer and the output layer. So corresponding to the xor logic the problem is the complex problem because it is not possible to draw a single decision boundary between the classes. So that is why we are considering the hidden layers between the input layer and the output layer.

So let us move to the next slide. So already I have shown the feature space corresponding to the xor logic. So we have one input here this is another input another input and another input and this is x_1 and this is x_2 and we have a decision boundary between the classes. So one is this line and another is this line. So corresponding to this input this input and this input the class is class 1 and corresponding to this and this inputs the class is 2.

So corresponding to this the network I can consider like this. So maybe I can consider this equation $-0.5 + x_1 + x_2 \geq 0$ and already I told you we have to consider one AND operation. So AND operation we are considering $1.5 - x_1 - x_2 \geq 0$. So we are considering the AND operation between these two decisions.

Two decisions are taken and after this we are considering the AND operation. So corresponding to this I want to show the artificial neural network for this problem that is the exclusive or classification problem. So we have x_1 we have x_2 and we have a bias terminal.

So I can show the interconnections. So just I am implementing these two equations. This weight is 1, this weight is 1, this weight is 1.

So bias is connected like this. This bias is -0.5 and another connection I am showing here up to this point. So this is 1.5. So this is your -1 and this is -1. So just I am implementing these two equations $-0.5 + x_1 + x_2 \geq 0$ and another one is $1.5 - x_1 - x_2 \geq 0$. These two I am implementing now. After this I have to consider the AND operation. So for AND operation so already you know what is the neural network for the AND operation.

So for AND operation we are considering this one. It is 1. This is -1.5 and the bias this bias is again I am considering another bias. This bias is 1. This bias is also 1 and we are having the output.

Output is y that is the output. So this is the artificial neural network for the XOR logic. So just implementing these two equations. So one thing is that we are finally this step actually we are doing the AND operation. So what is the need of the AND operation that already I have explained in my previous slide. So this is the implementation of the XOR logic so with the help of the artificial neural network. So these are the examples how the artificial neural network can be employed for pattern classification.

After this I am discussing another important network because I have explained the concept of the nearest neighbor classifier. So how actually the nearest neighbor classifier can be considered in case of the artificial neural network. Nearest neighbor classifier. So this is a popular algorithm of pattern classification and how it can be implemented in the artificial neural network. So suppose X is the feature vector.

So it is a d-dimensional feature vector $[x_1 x_2 ... x_d]^T$ and we are considering C number of classes C classes we are considering. So it is a C class classification problem and we are considering the Y_i that is the centroid of the classes centroids. So $i = 1,2, ..., C$, C number of classes we are considering. So C number of centroids we are considering. So in the nearest neighbor classifier we have to find a distance between the feature vector and the centroid.

So we are considering the Euclidean distance the distance between X and the centroid we are determining. So it is nothing but $d^2(x\text{-}, y_i) = (x\text{-} - y_i)^2$. I can write like this or maybe I can write like this $\sum_{j=1}{}^{d}(x_j - y_{ij})^2$. I can write this equation like this. This equation I can expand because it is nothing but $(a - b)^2$. So it is $(x_1^2 + x_2^2 + ... + x_d^2) - 2(x_1 y_{i1} + x_2 y_{i2} + ... + x_d y_{id}) + (y_{i1}^2 + y_{i2}^2 + ... + y_{id}^2)$.

So I can consider like this. So in the minimum distance classifier that is the nearest neighbor classifier I have to find a minimum $d^2(x\text{-}, y_i)$. I have to select which one is the minimum distance because in this case I have to find a distance between the feature vector and the centroids. C number of centroids we are considering because I have C number of classes and I have to determine the distance between X and all the centroids and I have to select the minimum distance. So minimum distance I have to select minimum distance the distance is nothing but the distance between the feature vector and the centroid. So I have C number of classes I have to consider the minimum number of a minimum distance between $x\text{-}$ and y_i.

So in this expression if you see in this expression the first term is $(x_1^2 + x_2^2 + \ldots + x_d^2)$. So this is same for all the classes and it has no role in classification. So that means we can cancel this term. So we may not consider this term it is not important. This term is not important.

So we have to consider the other terms and one important point is if I consider the discriminate function I have to determine the maximum discriminate function. The maximum discriminate function corresponds to the minimum distance. In my earlier classes I have explained the concept of the discriminate function for a C class pattern classification problem. I have C number of discriminate functions and I have to select the largest or the maximum discriminate function that corresponds to that particular class.

So I have to pick the largest discriminate function and that gives the corresponding class. So I am repeating this. So I have to select the maximum discriminate function. So the maximum discriminate function corresponds to minimum distance between the Feature vector $\bar{x}$ and the centroid. So the maximum discriminate function corresponds to that minimum distance. And in this case I have to determine the minimum distance that corresponds to the maximum discriminate function. So in terms of discriminate function I can write like this $g(\bar{x}) = (x_1 y_{i1} + x_2 y_{i2} + \ldots + x_d y_{id}) - \frac{1}{2}(y_{i1}^2 + y_{i2}^2 + \ldots + y_{id}^2)$. So this is the expression for the discriminate function. So we have to maximize this discriminate function. So we have to find the maximum $g(\bar{x})$ we have to find because I have the C number of discriminate functions. So move to the next slide. So what we have obtained in my previous slide so $g(\bar{x}) = (x_1 y_{i1} + x_2 y_{i2} + \ldots + x_d y_{id}) - \frac{1}{2}(y_{i1}^2 + y_{i2}^2 + \ldots + y_{id}^2)$. So we have this expression and already I told you that I have to determine the maximum discriminate function.

So we have suppose the discriminate function is $g_1(\bar{x})$ corresponding to the class 1 $g_2(\bar{x})$. So all these discriminate functions we can determine. So I have C number of discriminate function and out of this I have to determine the maximum discriminate function which one is the largest discriminate function we can determine. So corresponding to this equation my artificial neural network I can show. So I am drawing my artificial neural network. So this is x_1 inputs are x_1 up to x_d because we are considering the d dimensional feature vector.

So this is the $\bar{x}$ and we are finding the distance between the feature vector and the centroid. So this is nothing but y_1 we are finding the distance between the $\bar{x}$ and the centroid and you can see the bias this is this part is the bias part. So it is $-\frac{1}{2}$. So this is nothing but $\bar{y}_1^2$. So for all the classes this is for 1 for 2 like this up to C number of classes I have to consider these connections only I am showing the connections for the node 1 for the class 1. For C number of classes you have all these connections and corresponding to this you can determine $g_1(x\text{-})$ you can also determine $g_2(x\text{-})$ this $g_C(x\text{-})$ also you can determine this is $g_2(x\text{-})$ $g_C(x\text{-})$ also you can determine.

So all this you can determine and we have to select the maximum discriminant function we have to pick the largest discriminant function. So for this we are considering a comparator and these are the inputs to the comparator these are the inputs to the comparator and I have to pick the largest discriminant function and based on this I can select the particular class. So classes are y_1 that is the centroid for the class y_1 y_2 y_C so C number of classes we are considering. So if you see this part of the network only I am showing the connections for only one centroid. So for all the centroids C number of centroids we have to do the connections and we have to find the discriminant function $g_1(x\text{-})$ $g_2(x\text{-})$ so all these we have to determine and after this we have to consider one comparator because we have to determine the largest discriminant function and based on this I can take a classification decision.

So this is the concept of the the nearest neighbor classifier. In this class I explained the concept of the artificial neural networks and I explained the importance of the activation function and the bias input. Also I have explained some classification problems so for this I considered XOR logic OR logic and the AND logic. So for AND and OR logic we can consider linear decision boundary between the classes that is a linearly separable classification problem but in case of the XOR XOR is a non-linearly separable pattern classification problem and after this I discussed the concept of nearest neighbor classification technique how artificial neural network can be employed for this problem the problem is the nearest neighbor classification.

So that concept also I have explained. In my next class I will be explaining another important concept that is the radial basis function artificial neural networks. So let me stop here today. Thank you.