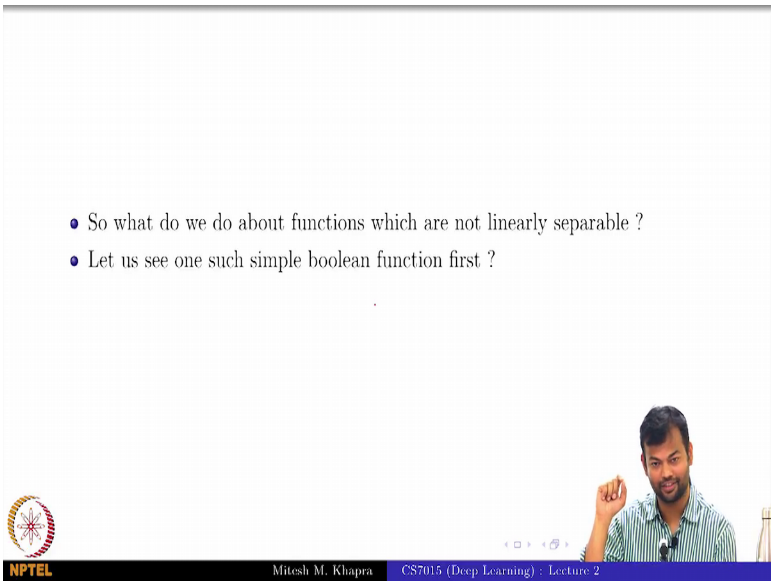
Deep Learning
Prof. Mitesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - 2.7
Lecture - 02
Linearly Separable Boolean Functions

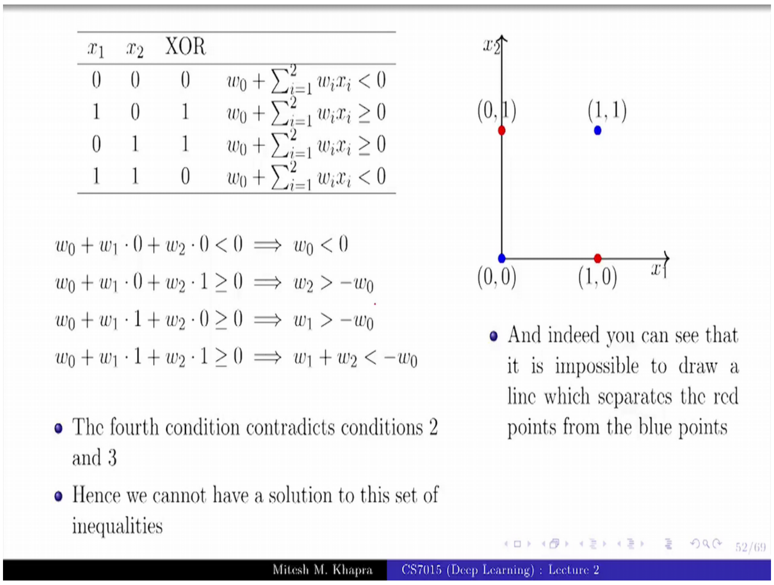
So, in this module, we look at Linearly Separable Boolean Functions again and we will try to make some more statements about them, ok.

(Refer Slide Time: 00:20)



So, what do we have do; so, the guiding question that we have is what do we do about functions which are not linearly separable and let us see one such very simple function. Can you guess what function I am going to talk about? All of you are paying attention in the first lecture, good right.

(Refer Slide Time: 00:36)

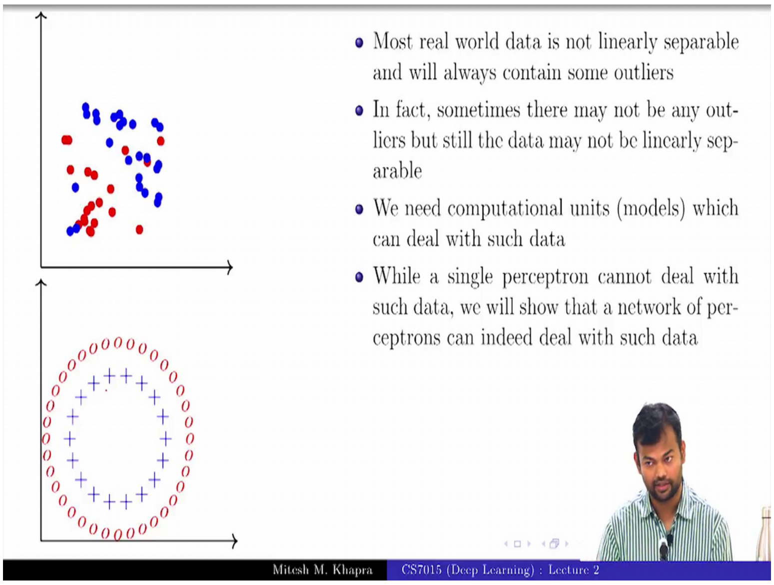


So, here is the XOR function. Now, these are the set of inequalities that result from XOR function. I hope right. Now, let us see the first condition implies that w naught should be less than 0, second condition implies this, third condition implies this, fourth condition implies this. Just looking at this can you tell me, can you find a configuration for w naught w 1 w 2, such that these inequalities can be satisfied together. No, right because 2 and 3 want it to be greater than minus 1 minus w naught and when you take an addition of that, it has to be less than minus 1.

So, that is not going to operate. So, you see a contradiction. So, this is a simple Boolean function which the perceptron cannot handle because it is not linearly separable. It is not linearly separable. There does not exist a line. If there does not exist a line, you cannot find the line, right. In fact, you can look at it visually, right.

So, these are the red points for which the output should be 0 or 1 and the blue points are the points for which the output should be 0. If we need to change this, I think we were using blue as positive and red as negative and you cannot just draw a line. There is no way you can draw a line such that the blue points lie on one side and the red points lie on the other side, right. So, it is a simple two input function, right. So, it is not that I have taken a very contrived example, ok.

(Refer Slide Time: 01:57)

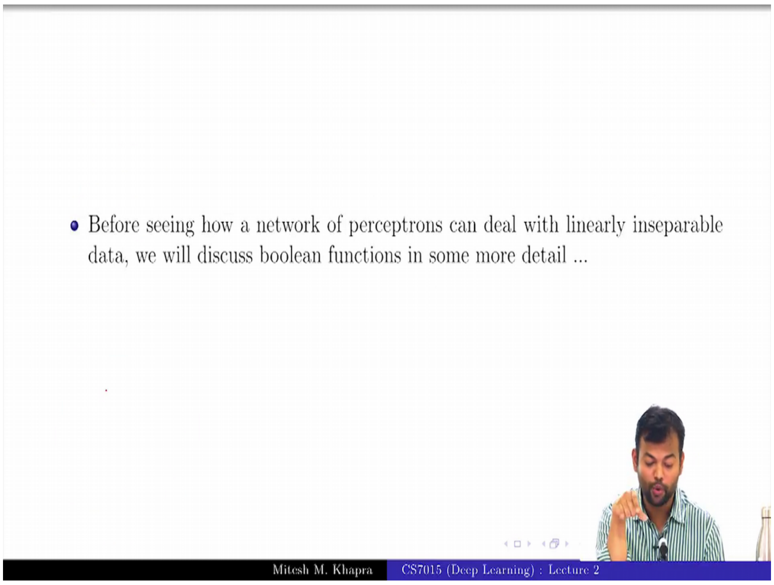


Most real world data is not linearly separable and it always contain some outliers, right. So, here maybe you have some data where you are trying to say that people which live in this part of the world belong to a certain or maybe people who live or work here have a certain qualification, people who work in this company may have a certain different qualification, right and there might be some outliers, right. It is not that is always going be very clean. So, now what do I mean? And it is not necessary that the points will only be outliers.

In fact, there could be a clear case where there are no outliers, but still you cannot find a line such that you separate the positive from the negative. Can you think of such an example? Good right. This is clear data. There is no outliers here as well. I mean it is just saying that everyone who lies within this boundary has a certain characteristic and outside that boundary people have a different characteristic, right and there is no outlier here, but you cannot separate this data with a line. So, all functions that you deal with will not go or are not going to be linearly separable.

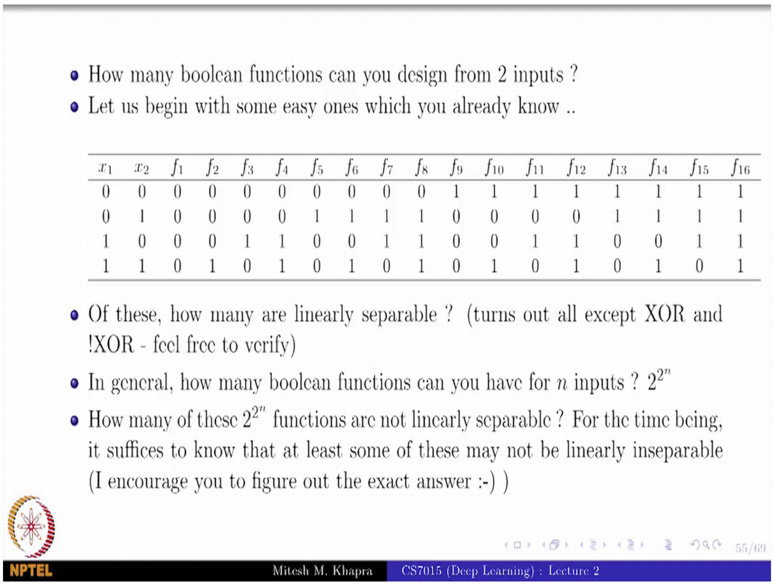
So, we have to work around those, right and while a single perceptron cannot deal with this, we will show that a network of perceptrons can indeed deal with such data. So, that is where we are headed, ok.

(Refer Slide Time: 03:16)



So, before going there we will discuss some more Boolean functions in more detail. And I will try to see what kind of non-linearly separable Boolean functions are there, right.

(Refer Slide Time: 03:26)



So, first of all how many Boolean functions can you design from two inputs? How many can you design? 16 looks like a good number from 3 inputs 256. How many if you understand this? Let us see. So, let us begin with some easy ones that you already know, right. So, these are two inputs x 1 x 2. What is this function always off? The other extreme is always on and I have already given you the answer f 16, right.

So, then you have function and then, some other functions, right. So, why did you reach 16? Actually because with two inputs we will have these four values to take care of and each of these are again binary. So, you actually have 2 raise to 2 raise to n, right. So, for three inputs 2 raise to 2 raise to 3 would be 256, right ok. Now, that is the easy part of these. How many are linearly separable? I will have to do any actually stare it in and seriously try to find the answer when you cannot really do that, right.

So, turns out all of them except XOT and in, not of XOR, ok; so for the two input cases, there are two functions which are not linearly separable, ok. For n inputs how many functions would be not linearly separable? It is an arbitrary, ok. N is not the answer. You are not going to disappoint me, not n ok, but what is the answer, ok. So, for n inputs, we will have two ratio 2 n functions of these we do not know how many are going to be not linearly separable. That is not a solved problem, although I encourage you to go and find the answer.

I am looking for a good will hunting kind of a moment, but all it suffices to know is that there exists some which are not linearly separable, right and that everyone agrees that there exists some, right and as n grows probably that number will increase and so on, but it is not known exactly. You cannot write it as a function, ok. So, what we have done so far is looked at Boolean functions, how many Boolean functions can exist and of that we just have concluded that there would be some which are not linearly separable.

So, we will end this module here.