So, last class right, we were, I just started a perceptron, it is like a modelling of you SUB_TEXT know a neuron, right and you know that, then you know that neurons apparently even in our biological system there are, I do not know right, several of them right, you know in terms of hundreds of millions and so on, billions maybe and they all interact with each other, right. And the idea is that you know all of this started around 1958, okay, this model of a neuron which is a perceptron, even before that there was something else, but you know we will not worry about that, let us go ahead with this. So a perceptron right, as I was saying it is a very simple model, but it actually helps explain a lot of things and in fact all the things that come later they are all, they have all evolved from this model, okay. So what it mean, what I said last time was that you have a neuron that is modeled as follows, so right it receives inputs right, I mean you know even within our own system right that is what happens, so these neurons right get inputs from say other this one neurons, these by themselves could be outputs of other neurons, but for the time being just imagine that right these are the actual inputs, and you have like X1, X2 right all the way up to some Xn right as I said and these are all kind of right you know coming as input to the neuron and these could by as I said these could themselves be the output of some other neuron or may be a bunch of neurons. And you have a threshold right which is like a theta, so this nonlinearity right that is where it comes, but then right we will see that one is just having a single neuron right may only be help, may only help you to solve a certain set of problems right, you still need to take it to a different level, but then for the time being rate suffices to say that you get some Y and where Y where you say whether a neuron fires or not right, so that 1 0 is to say whether this guy is going to fire or not right. And right now right now we will take it to be a binary value, but later on right I mean this Y can also be a real number okay, but then right for the time being let us say that the model is such that you know Y, so what was it that I had written last time, so Y is equal to 1 right, 1 if summation okay and all these are weighted right as I said last time W1 to Wn, so you take a weighted sum I going from 1 to n and if this is greater than or equal to theta then you say you know the output the neuron will fire, if not right I mean otherwise right I mean if this is like less than theta strictly less than theta then you say that then you say that it will not fire okay.

Now using such a simple thing right you might wonder what can you do okay, so what I will do is I will take an example to actually illustrate what you can do with just a single neuron right. Let us take for example a binary valued case okay not that a perceptron is actually limited to that, but just for simplicity right. Let us say that you know you have an OR gate right means you know a Boolean function, let us take a Boolean function right which we want to sort of implement and we can think of it as a kind of a classification problem okay. So let us take the simplest of cases, so what I have here is something like X1, X2 and the output is Y, so instead of n number of input let us take 2 input the situation, well let us say 0, 0 right as we know should produce an output that we ideally seek as 0, 0, 1 I think it is 1 right I mean I have not done this for a long time, but 1, 1, 1 and then 1, 1 is also 1 right that is how you call in that is the way an OR gate works.

Now another way to see it is to kind of draw it on a sort of a plane and then think of you know these two axes is X1 and X2 and mark these points right. Let us say that here is our 0, 0 yeah it will take a while to get adjusted to this and then here let us say is 1, 0 and here we will say 0, 1 and let us say here it is 1, 1 right and I am actually right deliberately indicating the 0, 0 by circle and then the rest by a cross and what you are seeking is some kind of a classifier now, what you are saying is right you need it is some kind of a classification this simple whatever you can do it in so many ways, but then I think of a line you know that is let us say I mean right that is not through the origin I mean the way I have drawn it let me just know what I meant was this okay. So let us say right so this 0, 1 and what you are looking at is something like that right simple sort of a classifier it can even be a line okay through these two points for example it could also have been this right through the through the two crosses that are at 0, 1, 1, 0 because your condition is greater than or equal to theta. Now as long as you fix a suitable theta right you can have all these points lying on one side of the line the crosses right 0, 1, 1, 0 two of them could also be on the line okay they could also be here the other line that I have drawn so you have so many options by the way and then 1, 1 right they will all come on one side and then your 0, 0 right which is supposed to be a label so you can think of these as labels right 0, 0 gets a label 0 and then all the other 3 right get a label 1 okay. Now this is a very, very simple problem right and right this is how it started so they started showing that you can actually implement things like this but in order to understand right okay what this actually means is we can actually we will let us try to arrive at so you so right so eventually okay even when you go later right as you go higher and higher it will all boil down to computing weights okay and this and right and this unknown which is the theta will be later on model we will call it a bias and so on for the time being right we will just call them as weights and another parameter theta right.

So the idea is that if you want this particular you know sort of a perceptron to be able to model that or function you can also do it for and you can try it for other cases if you wanted to do it then what it means is you have to arrive at this bunch of weights right in this case you just have to worry about w1 and w2 and then kind of theta right. Now I will just take an example which I have got here so right so the things are right here so what you have is for the first condition right what you have is w1 see this guy right so w1 x1 plus w2 x2 so if you look at w1 x1 is 0 plus w2 times x2 right I mean if you take the first case right w1 0 plus w2 0 right this you want it to be what do you want this to be as a condition less than theta right less than theta because you want the output label to be 0 and then for the next case we take 0 1 you will write it as w1 0 plus w2 1 because x2 is 1 and then this you will write as greater than or equal to c theta because you want the output label to be 1. Then the third case you will write this as w1 into 1 plus w2 into 0 right w2 into 0 this again is greater than or equal to theta and then the last one is w1 into 1 plus w2 into 1 because both inputs are 1 this again is greater than or equal to theta right that is what you would say which then means that from the first one right irrespective of whatever w1 and w2 take it is clear that right theta should be actually should be strictly greater than 0 that is what evolves in the first condition. The second one says that w2 should be greater than or equal to theta then this one says w1 should be greater than or equal to theta then the last one says w1 plus w2

should be greater than or equal to theta okay. Now obviously the last one follows right w2 and w1 are each greater than or equal to theta then some of course will be greater than or equal to theta.

Then what you can think about is some theta right let us choose that I have just chosen arbitrarily something okay I have chosen theta equal to 1 but as I said it does not have to be binary and all okay that is up to you whatever I am just giving it as an example and suppose I choose theta equal to 1 and then I have chosen and then w1 should then be greater than or equal to theta here as an example I have chosen w1 is equal to 2 just arbitrarily okay w2 is equal to arbitrary in the sense that as long as I satisfy these conditions it is okay any number is okay. So w2 is greater than or equal to 1 so I have chosen w2 to be equal to 1 and theta of course is 1 right. So I think so the point is right if you have a condition like this right and then if you go back to this diagram and then you have this, this, let us say this and this right and what do you get so if you kind of if you see the equation right of this line so you will have like w1 x1 so w1 what did I choose as 2 right so you have got like 2 x1 so if you want to draw this line you will have like 2 x1 plus w2 x2 right so w2 is 1 x2 is equal to theta right which is actually theta what did we choose as 1 right. So which means that if I take my x2 to be 0 then my x1 is how much 0.5 right so which means that you are looking at some point here okay 0.

50 and then if I take my x1 to be 0 then x2 is 1 right so then that will be like 0, 1 which is here so you are kind of afraid looking at a line like if you are looking at a line maybe I should change the color so you are looking at a line I think I can choose a color here it looks like okay so right so you are kind of looking at a line like that okay which is of course passing through this okay it is not very evident but right that is how it is and I mean this is not a unique solution right but one of the solutions could be this and right you are able to do a classification right. Now this sort of right this sort of what to say and I am not going to show the other examples leave it to you okay to try something else okay so the idea was that right so this way itself created some excitement right in the Henderson community and this perceptron model by the way was Rosenblatt 1958 okay so that you get some hang of when this happened right so it is like 1958 okay this is a perceptron model and then right Minsky I do not know the exact year I think it was about 4 to 5 years later what they said was right this is all fine but then right think of a situation where let me go to the next page where they said take the case of where right XOR gate I mean we are still looking at a Boolean functions which are still may be much simpler than the ones right that you can have where a function is real valued right and right now we are only looking at a single output you can also think about a multiple output case and so on.