Now, there is something called I think I let me just go through this quickly what is called a Perceptron Learning Algorithm. What is called PLA, Perceptron Learning Algorithm before we step into other things. So, for example, you might actually wonder right I mean then in the earlier case I just actually wrote down the weights directly and the biases directly right. So, you might wonder is there an organized way to actually do this right and the Perceptron Learning Algorithm actually does that. So, imagine that you know again let us go to write down the W is W1, W2 whatever W0, W1 write up to Wn and then the X is X0, X1 write up to Xn and write this is our unknown. So, when you say that we want a learning algorithm what we mean is that somebody right gives me a bunch of examples and says that these are all positive examples and these are all negative examples or in a sense right those carry label 1 and these carry label 0 then you just give me the correct weight such that I can actually do the classification and this learning algorithm says that no.

So, it gives the convergence guarantee provided these examples are linearly separable. Now this P, now this P are input with labels 1. Let me call this input with labels 1 and then input with labels 0 and those I mean which means that for which the output label is 1 and those for which the and then initialize you can initialize W randomly. If you do not have any idea then initialize W randomly.

So, while not convergence that means while you have not yet achieved convergence do a pick random X belonging to the set P union n by which mean that from the entire set of examples pick 1 if X belongs to P and summation Wi xi. Now this i will go from 0 to n because we have included the bias and all into it. So, summation goes from 0 to n if this is less than 0 which means that ideally right we should have liked it to be greater than or equal to 0 in which case we would not have done anything. But if it turns out that actually for that weight choice of weights that you have and for this X that is come that you know to be a positive example. If this turns out to be less than 0 then of course then you need to change your W.

So, what you do is then you change W and change W as W plus X and we will just show with some intuition as to why this makes sense and if X is X comes from the set of negative examples and if what happens is otherwise in the sense that summation Wi xi is actually greater than or equal to 0 in which case again you have to act you have to do something because you would have liked it to be less than 0. So, then do something else of course you do not have to do anything then W is equal to W minus X I may not or may write sometime the underscore, but remember that it is all vectors and it is not scalars. So, you can ask why does this kind of it make sense. So, the way to kind of write think about it is like this I mean so the way to write you know think about it is like on I mean if you want to kind of graphically see it right. So, what you can think of is the simplest case right that you can think of is you are actually you are actually right we are kind of thinking of getting at a weight vector I mean think of it as some it could be a 3D space it could be an N dimensional space or whatever let us say on a kind of you know a 2D grid.

So, what you are sort of saying is that I have got I have got examples of this kind let us say I have got all these examples which are actually which I mark with let us say a circle and then I mark let us say these other guys right maybe there are some again it depends upon how these examples are, but let us say I mark these as the positives I think we have been using cross of positive and these are maybe right my this are negative examples. So, what might happen is you know your kind of initial guess right could be could be could be could be could be you know kind of a W vector like that right which is not able to able to kind of do a classification the way right you would ideally like it to be. So, what you would like is you know a W such that such that right I mean such that if you take such that right I mean you know if you such that right it makes it makes an acute angle right I mean you know with respect to acute well no it should be an angle less than kind of 90 degrees with respect to all the positive examples and then all the right this one the negative examples which are on this side right they come I mean I will show that right why that makes sense. So, kind of geometrically right that is what it means. So, you are trying to so trying to align this is a W.

So, I can think of this line this line right that I have drawn this line is that line for which you see W transpose X is 0 and this line right if you think about think about a W right that is wrong then what could happen is then your W transpose X equal to 0 right could be could be like that right in which case in which case you are not really classifying things the way right you would like it then your W right would be well right in this case well I have not drawn it that correctly but then right imagine that that you chose something like that and then a W like that then right you would not be correct right because then you would have examples that are that are that are right mixed up. So, what you really want is you know ideally you would want a W transpose X like that whereas where on one side of the line you get all the positive examples and on the other side of the line you get all your all your say this one negative examples and this kind of a simple thing right that you are doing actually helps you do that. So, what it does is you know it tries to it tries to change the angle right of this of this is a W the W that angle the W the angle that W is making with respect to these examples such that you can actually end up doing the classification the correct way. I will write I will just show you why that makes sense okay this is just a just to give you a feel for what probably you are seeking right a graphical feel for that and what you really want is this this one. So right so if you think about the angle between the angle between between W and X between W and X right that we know is given by let us say a cos alpha right which is equal to W transpose X by norm W norm X right and so so if W transpose so right you can have situations like this right three situations if W transpose X is equal to 0 then we know that theta is 9 alpha this is alpha cos alpha right.

So alpha is 90 degrees and if W transpose X is less than 0 then what would you think your alpha will be less than what greater than 90 degrees alpha is greater than 90 degrees and W transpose X is greater than 0 then it will mean alpha is less than 90 degrees right. So now so this operation right so what you what you ideally want is for all the positive examples right for example for all the the positive examples you would want you would want something like this to get satisfied and for all your negative examples right you want something like that to be satisfied because for those you want your W transpose X to be less than 0 and for these you want your W transpose X to be greater than or equal to 0 this is what you want right. Now this is exactly what that update rule will do okay so if you go and check that update rule right so when we said that you know for X belonging to P so we said that if this condition is not satisfied right so let us say for X belonging to a positive example and suppose we find that we find that right we have to update update W then what we can do is we will take a new W right which is W new and let us say that right that we did we said that we will do we will do a W plus X okay and W plus X right which then which then means that if you try to do W new transpose with X right with that example then you will have something like W plus X of course the whole transpose into X or this like W transpose X plus X transpose X right which then means that and since rate W transpose X is what you already had right and now you know now right you have you have a term X transpose X right and this X transpose X is always a number right you know greater than or equal to 0 which then which which actually means that means that right I mean which then which then means that right you are able to inch closer to so right so this is what helps get this is what helps you get to the this is what this is what helps you right change the angle of the line right the line to to to kind of classify the positive example correctly to classify to classify the positive example of course to classify the positive example this is okay and similarly right what will happen I mean if you if you had on the contrary right if you had for X belonging to belonging to to see negative right you will have right so you will have again right what you will do is you will do W new is equal to W minus X and therefore right you can now show that W new transpose X right that is that is you mean right new new angle okay which you will get which now will now be will now will now depend upon W minus X transpose X just like W transpose X minus X transpose X which then is now say smaller than the than the right earlier value and this is always greater than or equal to 0 and and of course you know so it one might wonder whether the whether it actually converges but right we are not going into the proof of this but there is a convergence proof for this okay so if you repeatedly do this will actually end up right getting that getting that vector W in the right manner right like I said right ideally you would like it to be such that all these positive examples are over one side and then all these right negative examples are on the other side okay. So the one thing right that I just wanted to also mention is there is there is you know there is this right universal approximation theorem I will just state this today okay so this is called the universal approximation theorem approximation theorem which actually right again I mean that we are not we are not going to go to the proof of that but what it says is a multi-layer network of neurons a multi-layer network of neurons with a single hidden layer with a single hidden layer the single hidden layer can be used to approximate can be used to approximate any continuous function any continuous function see up till now we saw what are called a Boolean functions right now this is going way above that approximate any continuous function to any desired accuracy or to any desired any desired preposition that okay so right what this means is that what it means is that if ideally right what you would like to do is do is you know get a mapping G of X see I mean what exactly are you doing right using a network you are saying that X goes as input and then and then right you need a Y and you are sort of saying that you are trying to get that mapping let us say G of X right which is what you ideally want so you want Y to be equal to some G of X and what it is saying is if you are trying to attempt to get to the G of X which we do not know right this these functions we do not know a priori right this is all this could be heavily complicated we do not even know what the G of X is but then right I mean so what this means is if you had an analytical you can show it if you do not have it then still right what it is saying is if if you are if you are trying to if you try to approximate it right with some with with this one a network right then then that I mean network rate will give you give you an F of X and you can arrive at an F of X such that G of X minus X of X is arbitrarily small I mean you can come as close as possible to this G of X which is what you ideally probably are seeking but then you may actually end up getting an F of X but that F of X can be arbitrarily close to your G right that you want and you know this is a very sort of a powerful thing right so this I think was actually shown by two people by Saibenko in 1989 and I think there was an there was a further improvement over this by by Hornick in 1991 this is just to give give a timeline okay and yeah there is a there are there are a few few kind of very tricky things about this which I think which you know which you know I will not kind of right I mean you know you know enter into for the time being just I mean if you have any doubts you feel ask and you know you can ask and then maybe write I will clarify but I do not want to want to stir the pot okay but but but then the whether the point is this right I mean this by itself right is actually a is actually a powerful result right and which then means that whether you are solving a regression problem or or right whether you are solving a classification problem irrespective of what you are doing right this particular universal approximation theorem gives you the guarantee that whatever whatever it is right that you are trying to do you can get as close as possible to to the to the function right that you are that you are trying to trying to what you call I mean trying to yeah trying to right emulate or trying to arrive at right and most of the time we do not know G okay we should also realize that okay we just know the task we do not know what G will take you there and the whole idea behind the developing a neural network is to be able to approximate G right as closely as possible even though we do not have an analytical form for G okay I think I will stop here.