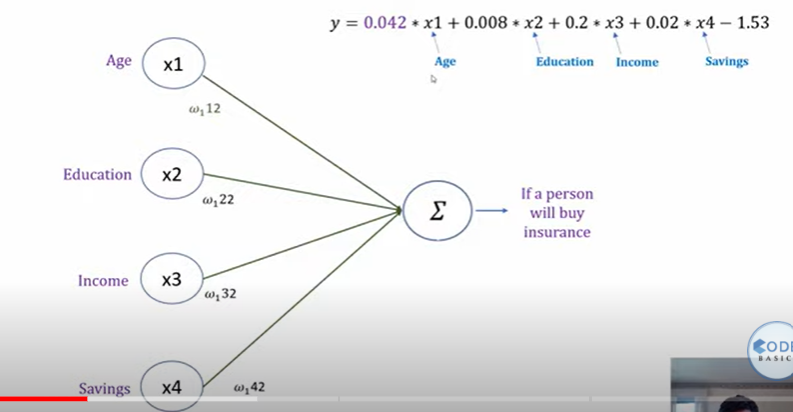
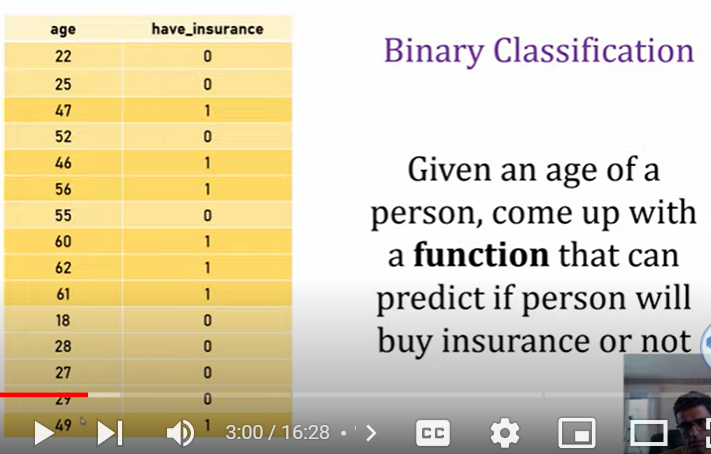
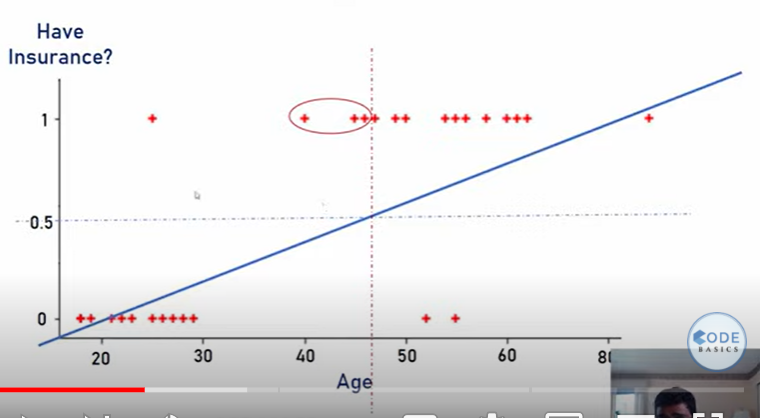
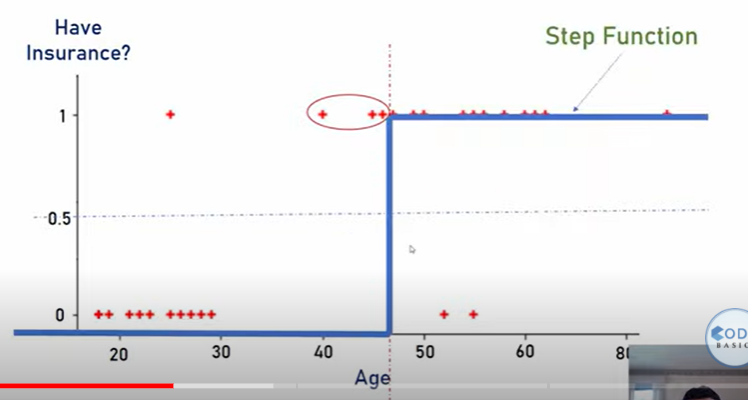
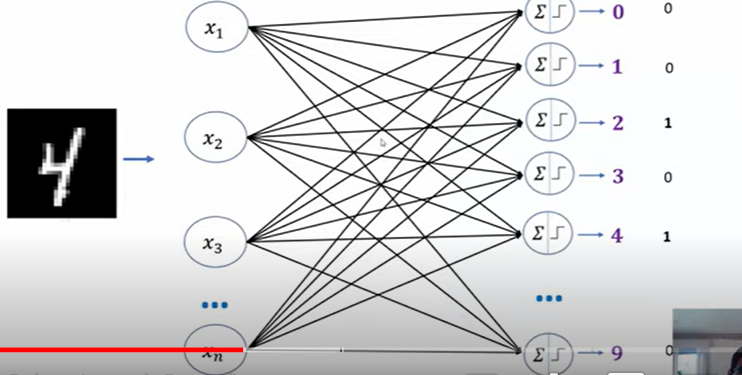
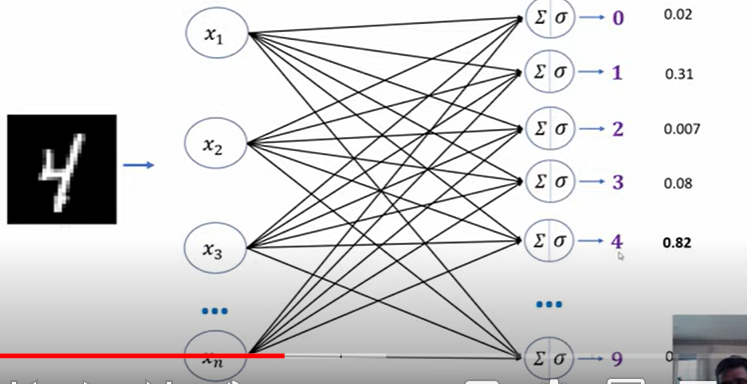
Activation Function:

* It is used to tell whether Neuron is firing or not firing i.e, it tells whether the activation function is activated or not.
* It also means that it will decide whether the neuron’s input to the network is important or not in the process of prediction
* 
* The above image shows that without an activation function, a neural network is just a linear model with linear equation, where even the hidden layer becomes of no use
* And it is common that linear model cannot solve the problems in universe and hence to inject non linearity activation function is used

Let’s take the example of the insurance data set

* 
* Through linear regression, we get the below graph.
* 
* This above image shows that the value greater than or equal to 0.5 tells that people have insurance and others do not have insurance
* And drawing this graph, is known as step function
* 
* The same step function is used in multiclass classification of handwritten digits
* 
* Where there is an image of number four and if we use the step function. And we might end up getting output value as 1 for both 2 and 4 classification classes
* And it becomes hard to make decisions and here is where the use of sigmoid or logit function comes into picture.
* Sigmoid gives a smooth curve instead of just 0 and 1 and because of this we get numbers like 0.02, 0.82 which helps us in decision making
* 

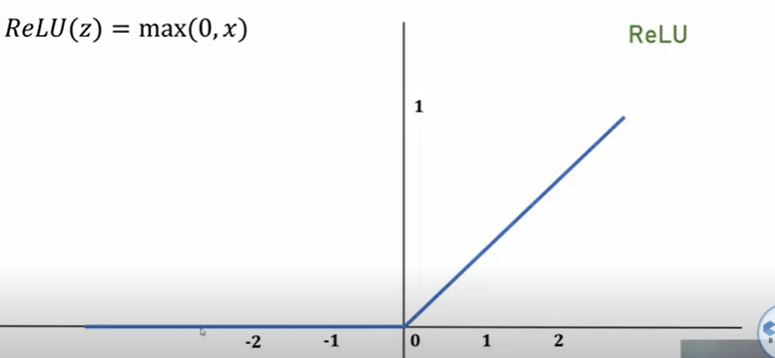
## Tanh activation function

* This works just like sigmoid function, but instead of giving values between 0 and 1, tanh gives value between +1 and -1

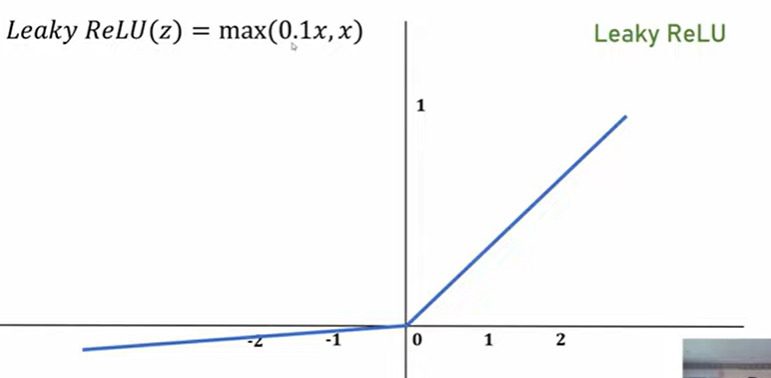
NOTE:

* Use sigmoid in the output layer. All other places try to use tanh
* Issues with sigmoid and tanh is that they have vanishing gradients problem that’ why ***RELU*** was introduced

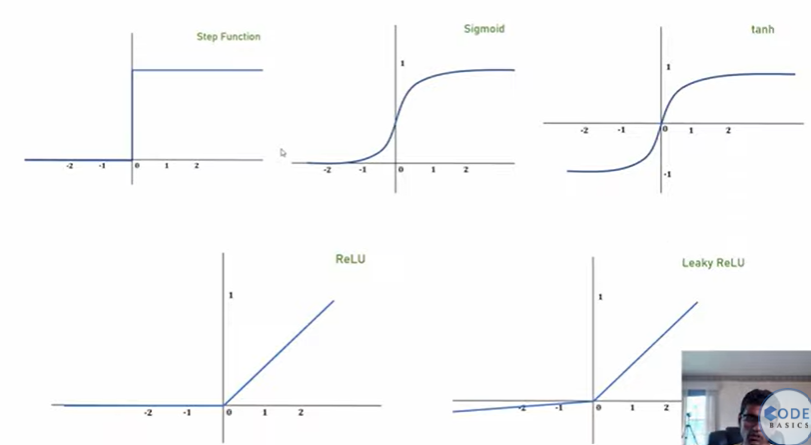
## ReLU Activation function

* If the input value id 0 or less than 0, then the output is 0 else the output is 1.
* 
* ***For hidden layers if you are not sure which activation function to use, just use ReLU as your default choice***
* The computation of ReLU function is very simple and its a light weight function
* ReLU also has vanishing gradient problem and that is because if the input value is 0 or less than 0, then the derivative is 0
* Hence, Leaky ReLU was introduced

## Leaky ReLU Activation Function

* 

## Summary

* 
* Choosing an activation function is all about trial and error
* Implementation Link: [activationFunctions.ipynb](https://colab.research.google.com/drive/1aT3GLboqmgKRyQqzqNRi7Dk4CLVtrBPj?usp=drive_link)