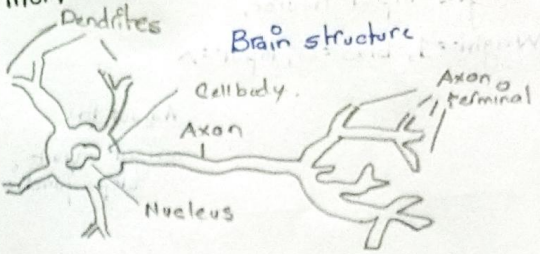
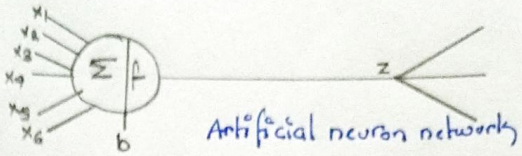
01:1 NEURAL NETWORK FROM SCRATCH IN PYTHON

CHAPTER 1 – Introduction to Neural Network

What is neural network?

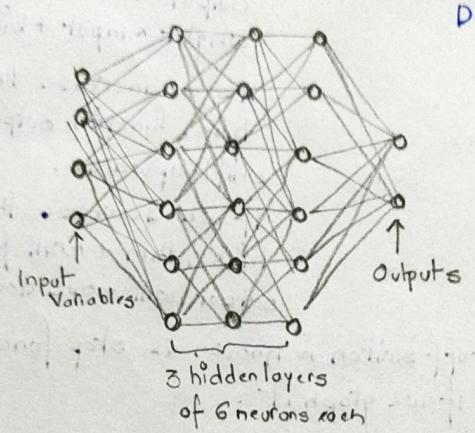


- It is inspired from a organic brain.
- Neural network also called Artificial Neural Network (in recent times, we dropped the "artificial" part)



Artificial neuron network

Sample example of a neural network with 3 hidden layers of 06 neuron each.



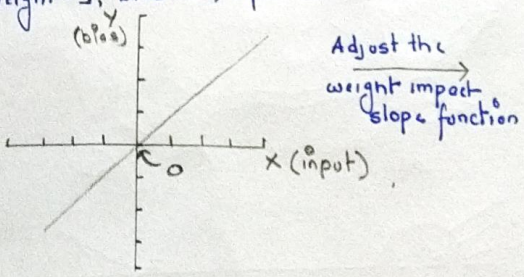
Dense layers – The most common layers, consist of interconnected neurons.

- In dense layer, each neuron is connected to every neuron of next layer, which means that its output value becomes an input for next neuron.
- Each input / connection between neurons has a weight associated with it, which is trainable factor of how much of this input to use and this weight get multiplied by the input value.
- Once all the input weight flow into our neuron, they are summed and a bias another trainable parameter is added.
- The purpose of the bias is to offset the output positively or negatively, which can help us map more real-world types of dynamic data.
- Concept of weight and biases can be thought of "knob" that we can tune to fit our model to data.
- In neural network, we have thousand or even million of parameters tuned by the optimizer during training.

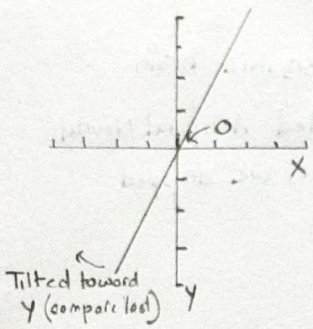
Why we cannot choose one either have biases or just weights?

- Both impact neuron in a different way.
- Since weights are multiplied, they will only change magnitude or even completely flip the sign from positive to negative or vice versa.
- **Output = Weight × input + bias.**

Single input neuron
Weight = 1, bias = 0, input = x



Single input neuron
weight = 2, bias = 0,

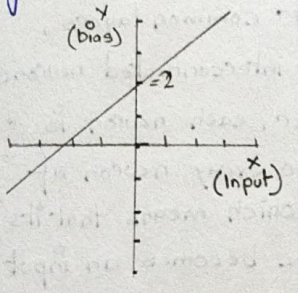


output = weight × input + bias

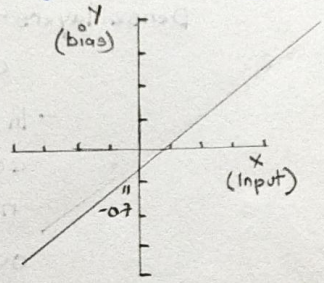
As we increase the weight, the slope will get steeper.

If we decrease the weight, slope will also decrease.

Weight = 1, bias = 2.



weight = 1, bias = -0.70
-0.70

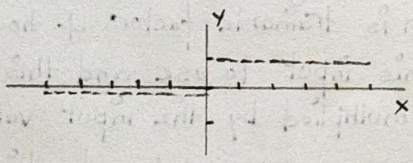


Output = weight × input + bias

As we increase the bias, function output shift upwards.

If we decrease the bias, then overall function output will move downward.

Step Function → In programming language, on-off switch is known as step function. because it looks like steps if we graph it.



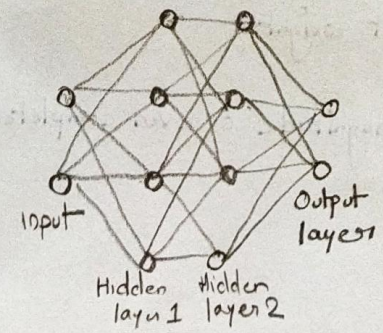
$$y = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

- For a step function, if neuron output value which is calculated by sum(input × weight) + bias is greater than 0 neuron fires output 1 else 0

Formula of single neuron, output = sum (input × weights) + bias.
Usually apply an activation function to this output, output = activation (output)
Here we use step function as activation function.

- Neural network uses different function / activation function such as ReLU (Rectifier Linear), Softmax etc.

Overview of NN → Example of 2 hidden layers and 4 neuron each.



- Along with 2 hidden layer, two more layer - input, output
- Input layer represent actual input data. Preprocess the data through normalization and scaling and you need inputs to be in numeric form.
- Output can be classification

01.3 - End goal for neural networks is to adjust their weights and biases (parameters)

- Common problem in neural network is overfitting, just memorize the training data. So use "in sample" data to train the model and then use "out-of-sample" to validate an algorithim (or neural networks in our case).

  Eg- Dataset consist 100 samples of data, take 10 samples and set them aside as "out of sample" or "validation" data. Train on 90 samples and validate the model on 10 samples. This is called generalization, which means learning to fit the data instead of memorizing it.

Loss → To train these neural networks, we calculate how "wrong" they are using algorithim to calculate the error (called loss) and slowly adjusting their parameters (weight and biases) over many iterations.