**Perceptrons:**

* In machine learning, the Perceptron is an algorithm for supervised learning of binary classifiers.
* A binary classifier is a function which can decide whether or not an input, represented by a vector of numbers, belongs to some specific class.
* It helps to classify the given input data.
* Perceptron is a single layer neural network and a multi-layer perceptron is called Neural Networks.

**The perceptron consists of 4 parts:**

* Input values or One input layer
* Weights and Bias
* Net sum
* Activation Function

**But how does it work?**

The perceptron works on these simple steps

1. All the inputs x are multiplied with their weights w. Let’s call it k.
2. Add all the multiplied values and call them Weighted Sum.
3. Apply that weighted sum to the correct Activation Function.

**Why do we need Weights and Bias?**

* Weights show the strength of the particular node.
* A bias value allows you to shift the activation function curve up or down.

**Why do we need the Activation Function?**

* In short, the activation functions are used to map the input between the required values like (0, 1) or (-1, 1).

**Where we use Perceptron?**

Perceptron is usually used to classify the data into two parts. Therefore, it is also known as a Linear Binary Classifier.

One of the limitations of the perceptron model is that the learning algorithm works only if the data is linearly separable. It is a characteristic of the perceptron neuron itself which behaves like a step function. To overcome the problem with Perceptron, They introduced a new type of artificial neuron called a sigmoid neuron.

**Deep Neural Network:**

A deep neural network is a neural network with a certain level of complexity, a neural network with more than two layers. Deep neural networks use sophisticated mathematical modelling to process data in complex ways.

**Sigmoid Neuron — Building Block of Deep Neural Networks:**

In the sigmoid neuron, a small change in the input only causes a small change in the output as opposed to the stepped output.

There are many functions with the characteristic of an “S” shaped curve known as sigmoid functions. The most commonly used function is the logistic function.

* The inputs to the sigmoid neuron can be real numbers
* sigmoid neuron model uses the gradient descent algorithm.

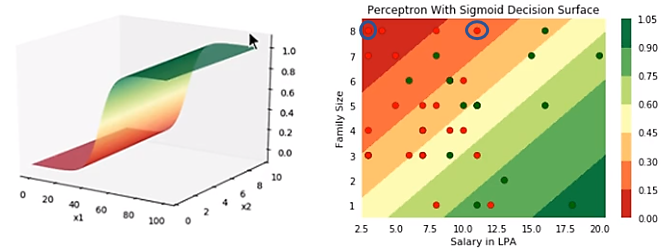
**Gradient Descent Algorithm:**

The objective of the learning algorithm is to determine the best possible values for the parameters, such that the overall loss (squared error loss) of the model is minimized as much as possible.

We will keep doing the update operation until we are satisfied. Till satisfied could mean any of the following:

* The overall loss of the model becomes zero.
* The overall loss of the model becomes a very small value closer to zero.
* Iterating for a fixed number of passes based on computational capacity.

Let’s see how sigmoid neuron will handle this non-linearly separable data. Once I fit our two-dimensional data using the sigmoid neuron, I will be able to generate the 3D contour plot to represent the decision boundary for all the observations.



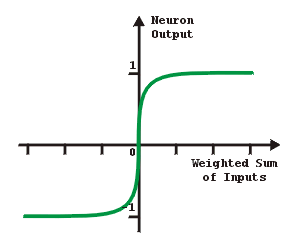
**The architecture of a Neural Network:**

* Artificial neural networks are composed of a set of neurons, joined together by synapses.
* Neurons perform a simple computational task, generally a basic yes/no decision.
* The structure of any neural network is therefore defined by the way in which various neurons and synapses are linked together.
* **Inputs and Outputs:**
  + Neurons in the network are generally arranged in layers – groups of neurons which are linked only to similar groups on their left and right.
  + The input neurons form the first such layer.
  + Values from the input layer are fed from left to right through various hidden layers.
  + The last layer of neurons in the network is known as the output layer.
  + This layered architecture is inspired directly by the design of the brain, in which layers of neuron cells are arranged in an onion-skin pattern.
* **Computation:**

Both neurons and synapses output a value which is dependant on the values presented to them at their input. The neuron’s inputs are summed to give a “Weighted Sum of Inputs” (WSI) value. Weighting is the multiplication of synaptic weights.

So we use the sigmoid function in neural networks which is almost like a boolean gives(0 or 1).

The equation for the sigmoid function looks like this 

This is the sigmoid graph.

Various different types of neuron exist, each of which generates its output based on a different process. Some common neuron types are listed here:

* + **Input neurons:** These provide a place where input values can be presented to the network. The output of an input neuron is equal to the input value assigned to it.
  + **Hidden (sigmoid) neurons:** These provide the network with the ability to perform calculations. The output of a sigmoid neuron is equal to the sum of its inputs multiplied by a sigmoid activation function.
  + **Output neurons:** Output neurons provide a place from which network output values can be read.
  + **Bias neurons:** Bias neurons output a steady value of 1. Bias neurons can be linked to other neurons to provide a constant bias