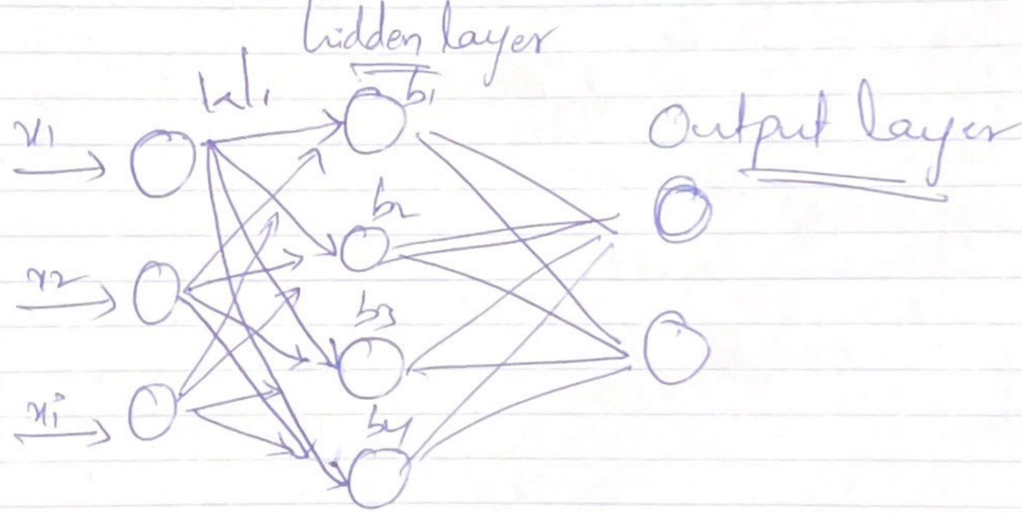
For a 2-layer Neural Network,
with 1 hidden layer and 1 output layer.

Neural Network image



Forward Propagation:-

Value for each output neuron
can be calculated as

$$y_j = \sum_i x_i w_{ij} + b_j$$

Dot product for forward propagation

$X = [x_1 \cdots\cdots x_i]$ $W = \begin{bmatrix} w_{11} & \cdots & w_{1j} \\ \vdots & & \\ w_{i1} & \cdots & w_{ij} \end{bmatrix}$

$B = [b_1 \cdots\cdots b_j]$.

Gradient Descent

$$w \leftarrow w - \alpha \frac{\partial E}{\partial w}$$

Backpropagation derivation:-

We need to give derivative of its ~~error~~ ERROR of the output which is $\partial E / \partial y$ and the backpropagation algorithm will give derivative of error of the input $\partial E / \partial x$

$$\frac{\partial E}{\partial x} \leftarrow \text{layer} \leftarrow \frac{\partial E}{\partial y}.$$

where

$$\frac{\partial E}{\partial x} = \left[\frac{\partial E}{\partial x_1} \frac{\partial E}{\partial x_2} \cdots\cdots \frac{\partial E}{\partial x_i}\right]$$

$$\frac{\partial E}{\partial y} = \left[\frac{\partial E}{\partial y_1} \frac{\partial E}{\partial y_2} \cdots\cdots \frac{\partial E}{\partial y_i}\right]$$ and E is

scalar of x any y matrices.

$\Rightarrow$ so to calculate $\frac{\partial E}{\partial x}$ we need to update the weights by using update rule

$$\cancel{\partial w} \quad \frac{\partial E}{\partial w} = \sum_j \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial w}.$$

Projection of Backpropagation.

$$\frac{\partial E}{\partial x} \leftarrow \frac{\partial H_1}{\partial x} \overset{\times}{\leftarrow} \frac{\partial E}{\partial H_1} \leftarrow \frac{\partial H_2}{\partial H_1} \overset{\times}{\leftarrow} \frac{\partial E}{\partial H_2}$$

Now, we need to add activation function for every derivative of backpropagation

$$\frac{\partial E}{\partial x} = \frac{\partial E}{\partial y} f'(x)$$ where $f'(x)$ is derived from every neuron

$$Y = [f(x_1)\, f(x)_2 \cdots\cdots f(x_i)]$$
$$= f(x).$$

Now after updating weights for $\frac{\partial E}{\partial x}$ we need to calculate error for each output which helps to reduce the loss function.

which is

$$E = 1/n \sum_i^n (y_i^* - y_i)^2$$

after getting the error, we need to calculate mean squar error where it is the mean of square of error of actual value and predicted value.

Sigmoid activation function

$$\sigma(x) = \frac{1}{1+e^{-x}}.$$

after derivating this function

$$\sigma'(x) = (1+e^{-x})(1 - 1+e^{-x}),$$

which can be termed as

$$\sigma'(x) = x(1-x).$$

If we use sigmoid, this will give the log values of the output ranging from 0 to 1.

But using update rule on binary classification using log loss, the model cannot interpret the binary output which should be either 0 or 1.