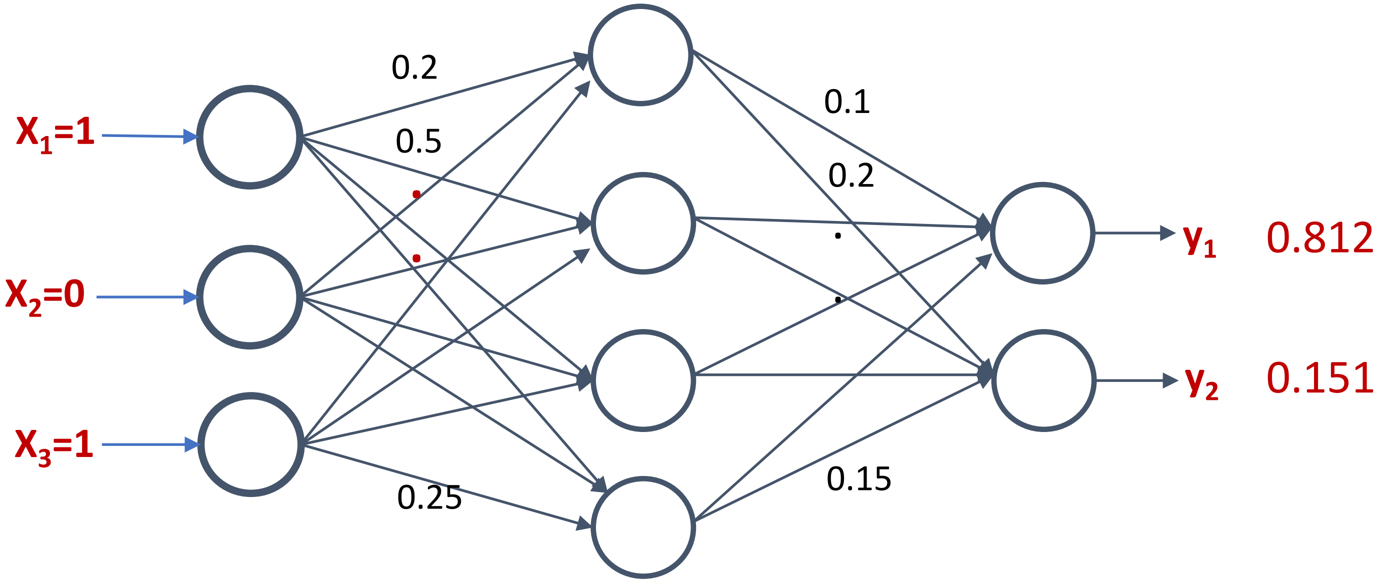
Bias in Neural Networks

Without Bias



$$\bar{y} = Sigmoid(\, Sigmoid(\bar{x}^T W).V\,)$$

[The Multi-layer perceptron models that we have discussed so far do not have bias

What is a Bias?

[It is easier to realize from a linear function]

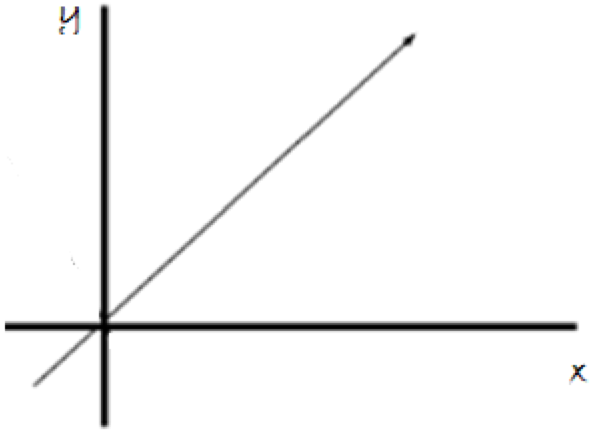
What is a Bias?

$$\boldsymbol{y = mx}$$

[Let us consider a linear function]

What is a Bias?

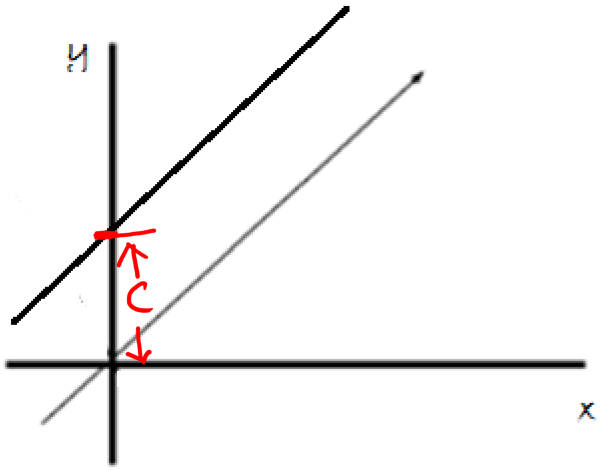
$$y = mx$$



[It represents a line passing through origin. With different values of m, we get different lines passing through origin.]

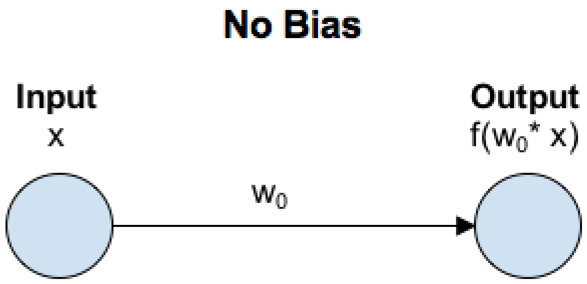
What is a Bias?

$$y = mx + c$$



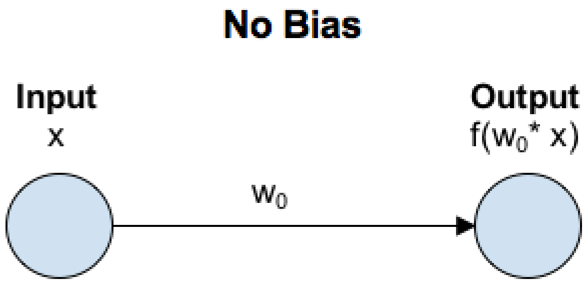
[If I want to move parallelly, I would need to add an intercept c. Here, c is a bias, which allows to move the line flexibly. So, the idea of adding a bias to a model is to make the model more flexible to fit into the problem better]

MLP with Bias



[Let us see this simple MLP with no hidden layer.]

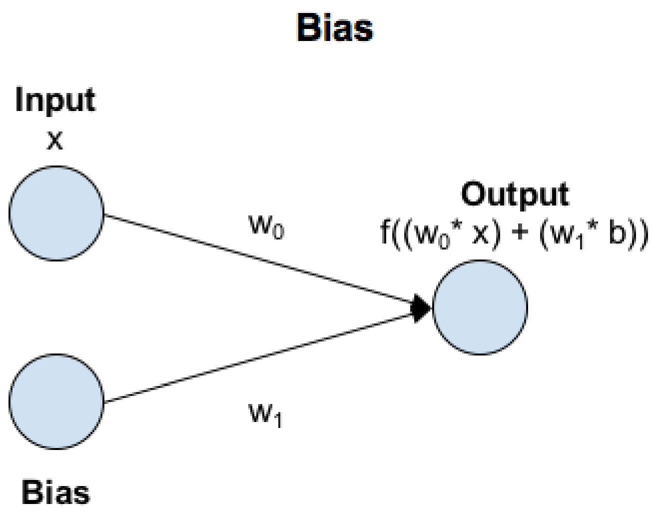
MLP with Bias



$$y = mx$$

[Let us see this simple MLP with no hidden layer. It is equivalent to linear function y = mx without bias]

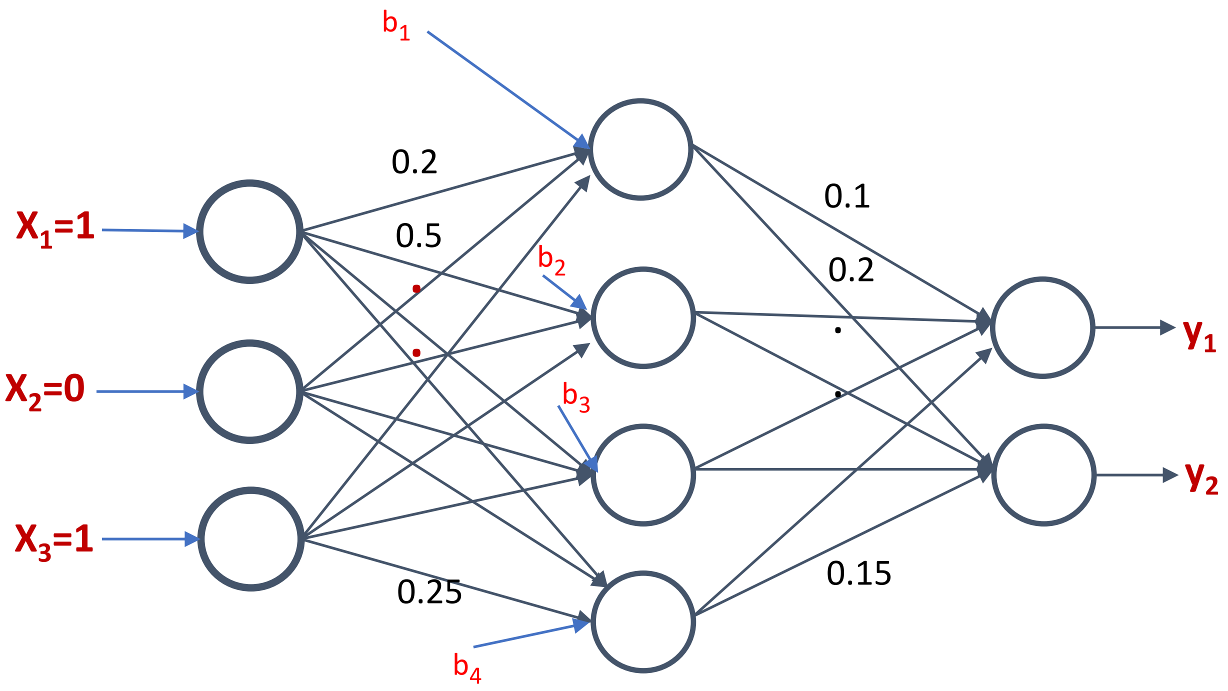
MLP with Bias



$$y = mx + c$$

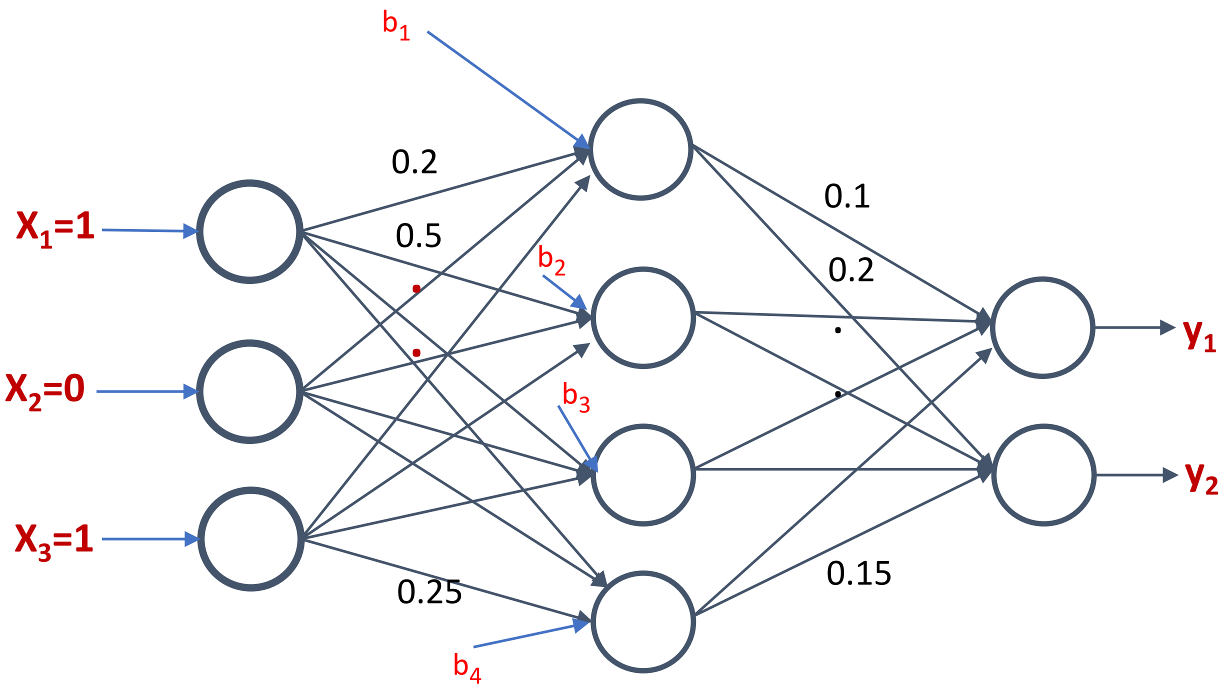
[Now, let us add bias. It is equivalent to the linear function y = mx + c]

MLP with Bias



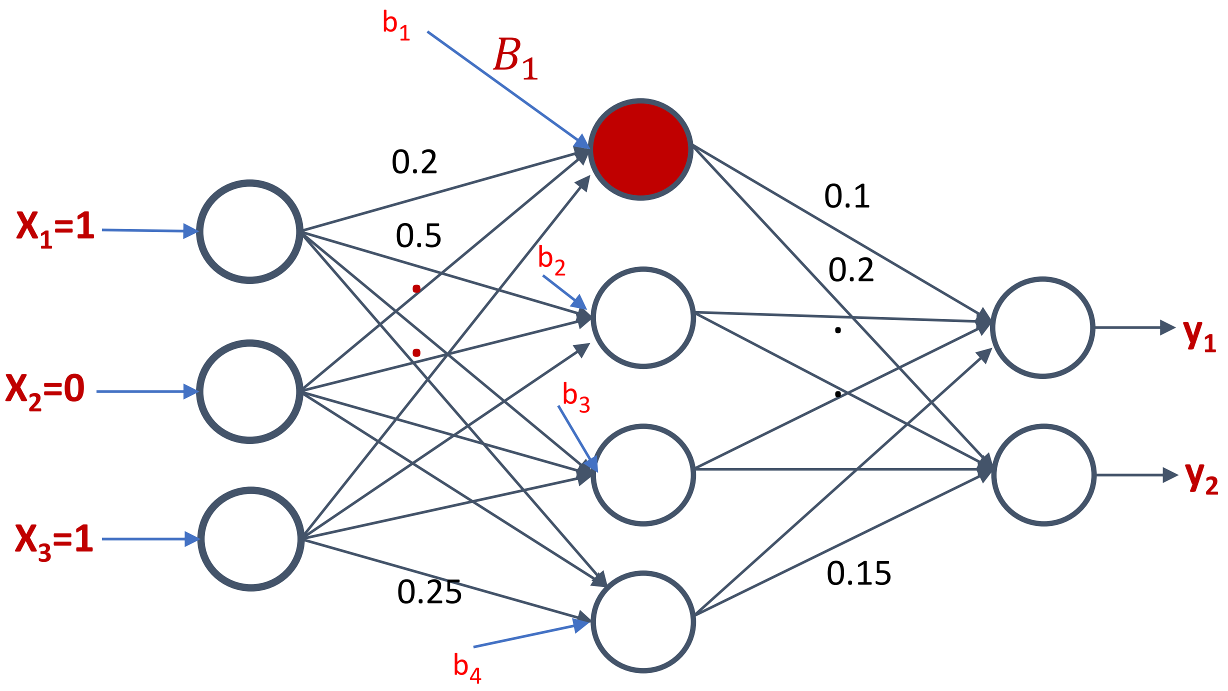
[To generalize, we can add bias to MLP as shown in figure. We are adding bias to hidden layer]

MLP with Bias



[It allows the activation function to shift by a factor defined by the bias and its corresponding weight]

MLP with Bias

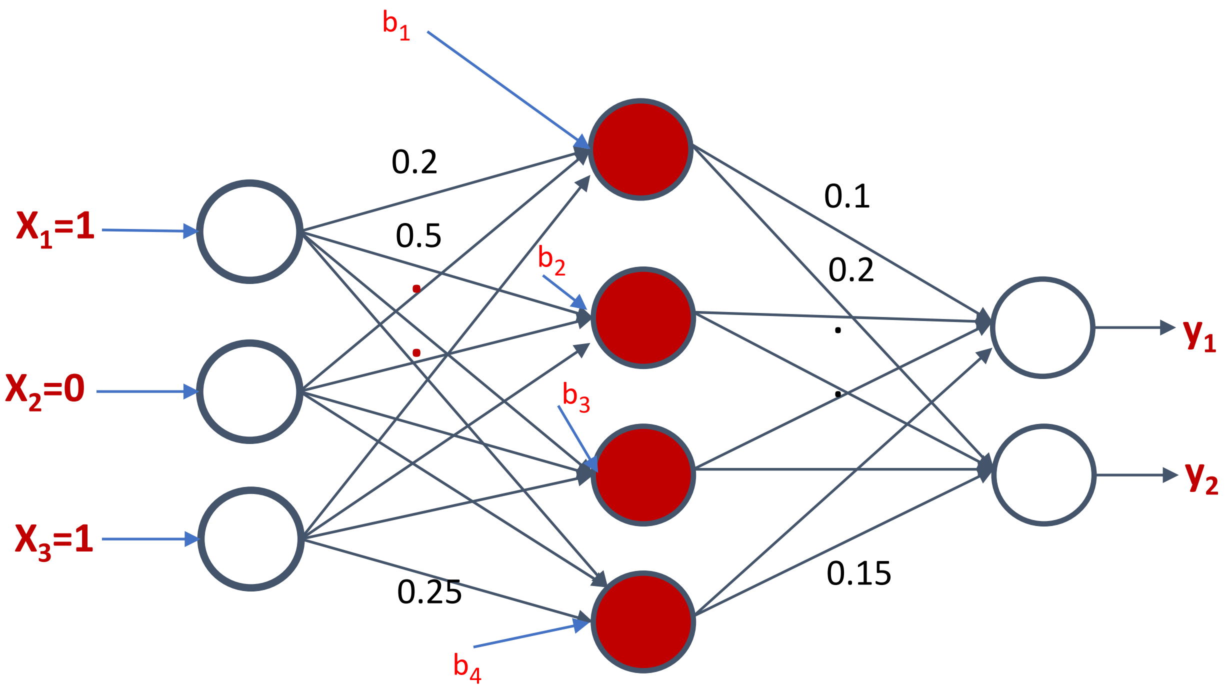


$$h_i = \sigma\left(\sum_j x_j W_{ji} + B_j b_i\right)$$

Where B_j is bias weight of the node j.

[With bias, the output from the hidden node can be defined as this expression]

MLP with Bias

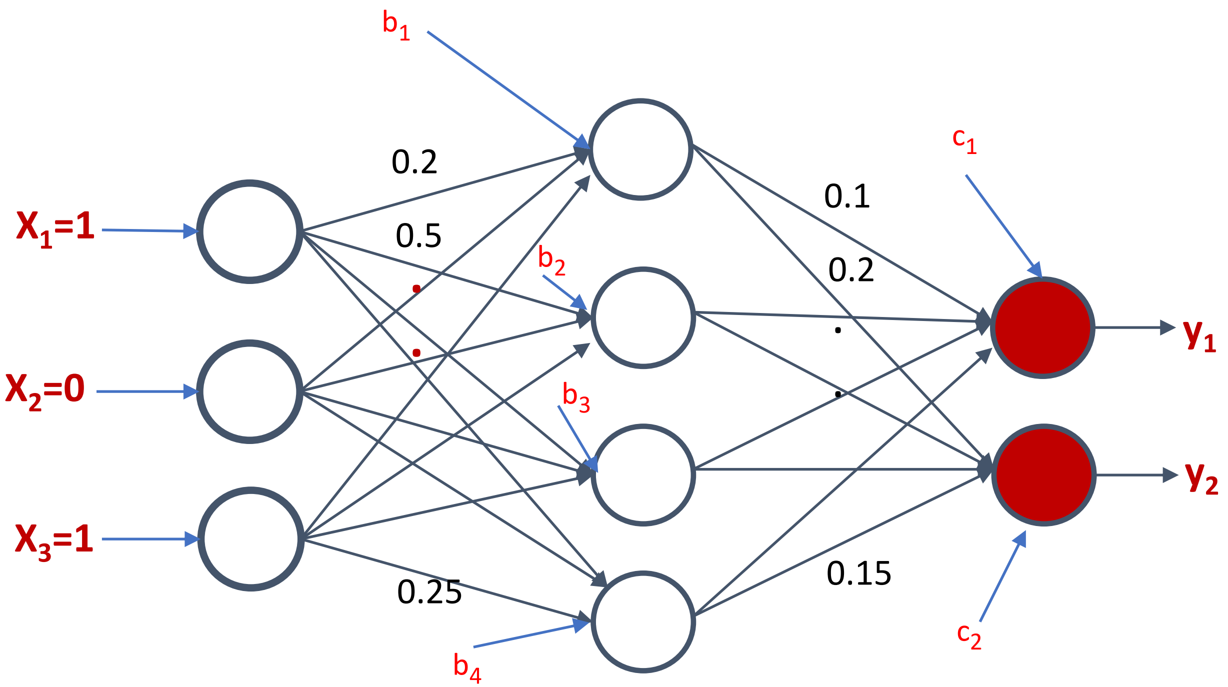


$$h_i = \sigma\left(\sum_j x_j W_{ji} + B_j b_i\right)$$

$$\bar{h}^T = \sigma\left(\bar{x}^T W + B\bar{b}^T\right)$$

[Likewise, the output vector from the hidden layer can be defined as this expression]

MLP with Bias



$$\bar{h}^T = \sigma\left(\bar{x}^T W + \bar{b}^T\right) \qquad \bar{y}^T = \sigma(\sigma\left(\bar{x}^T W + B\bar{b}^T\right)V + Cc^T)$$

[Similarly, bias may also be added to the nodes in the output layer]

Summary

- Bias allows the output vector to shift by a factor defined by the bias and its corresponding weight making the model more flexible.
- Biases are hyperparameter defined by users
- The weights associated with the biases are learnable parameters.