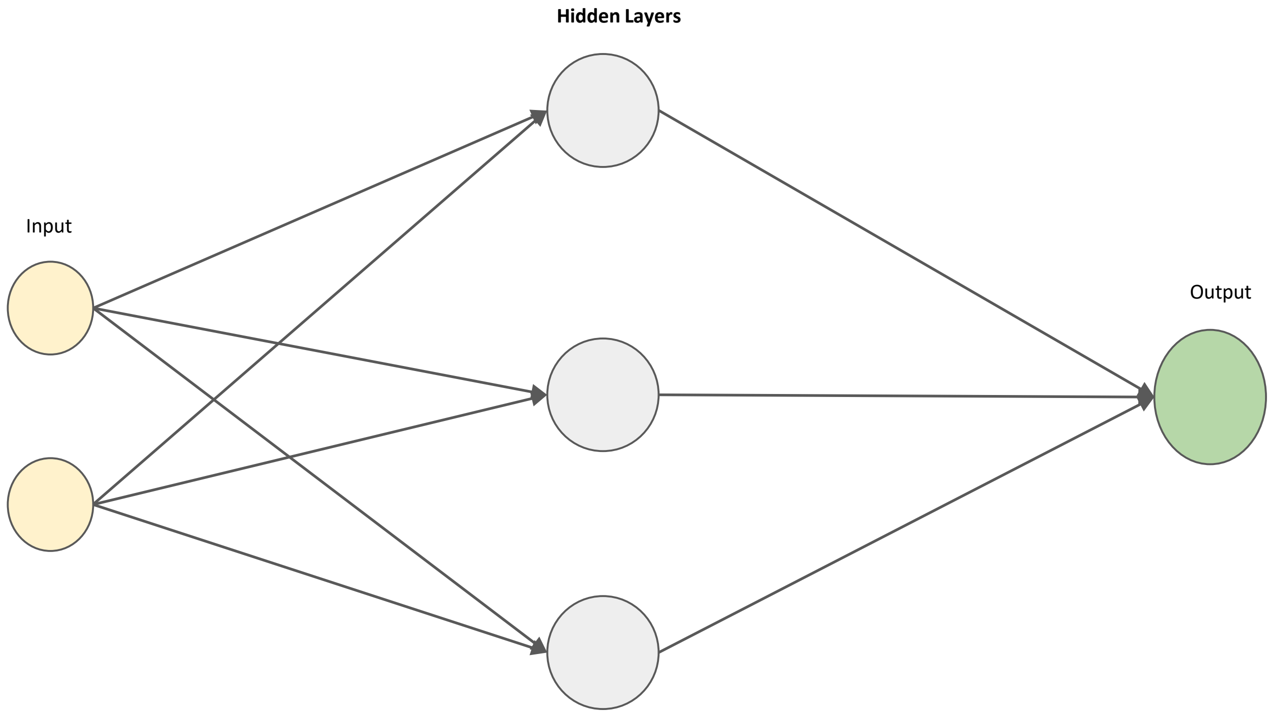
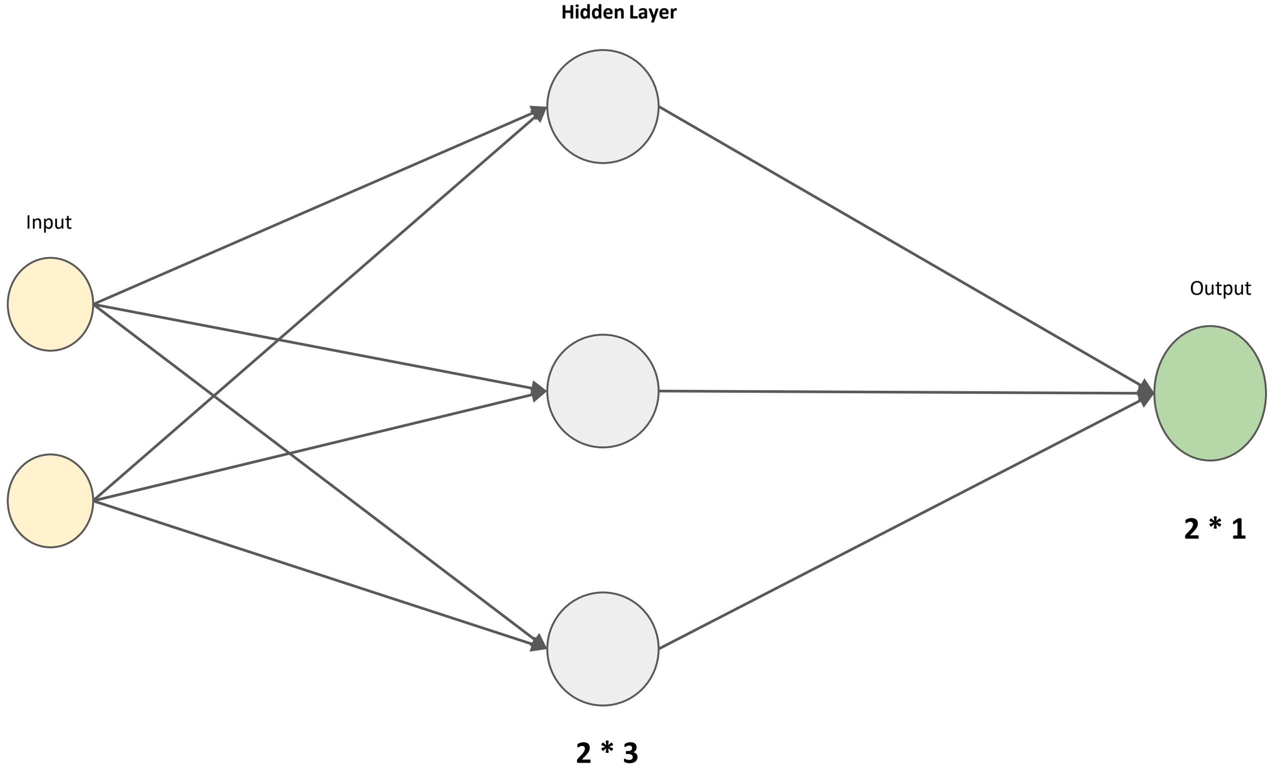
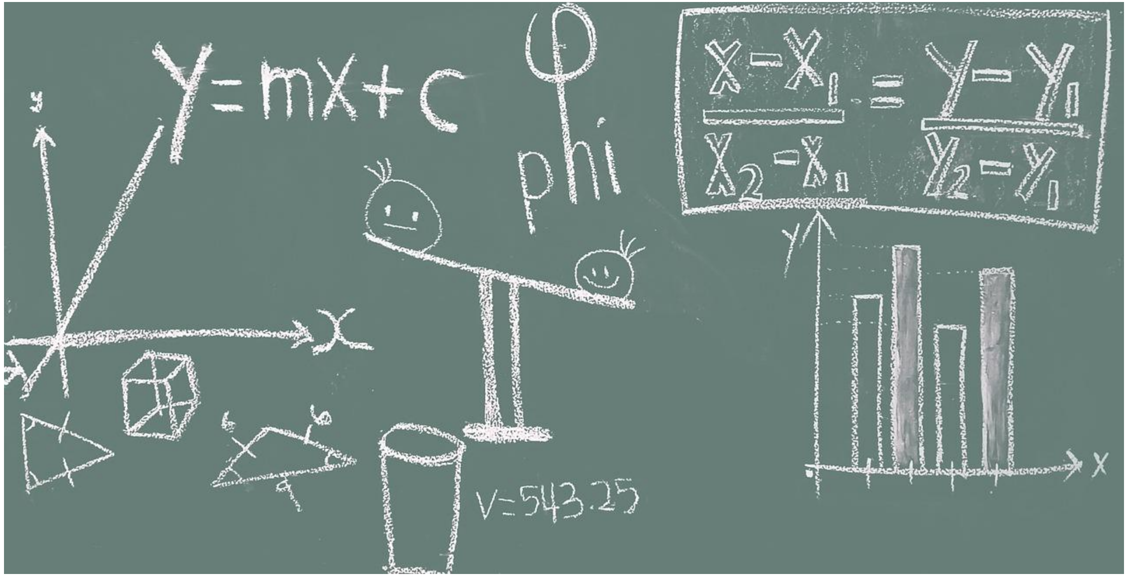
Working of
Neural Networks



Number of weights


Hidden Layer
Input
Output
2 * 1
2 * 3

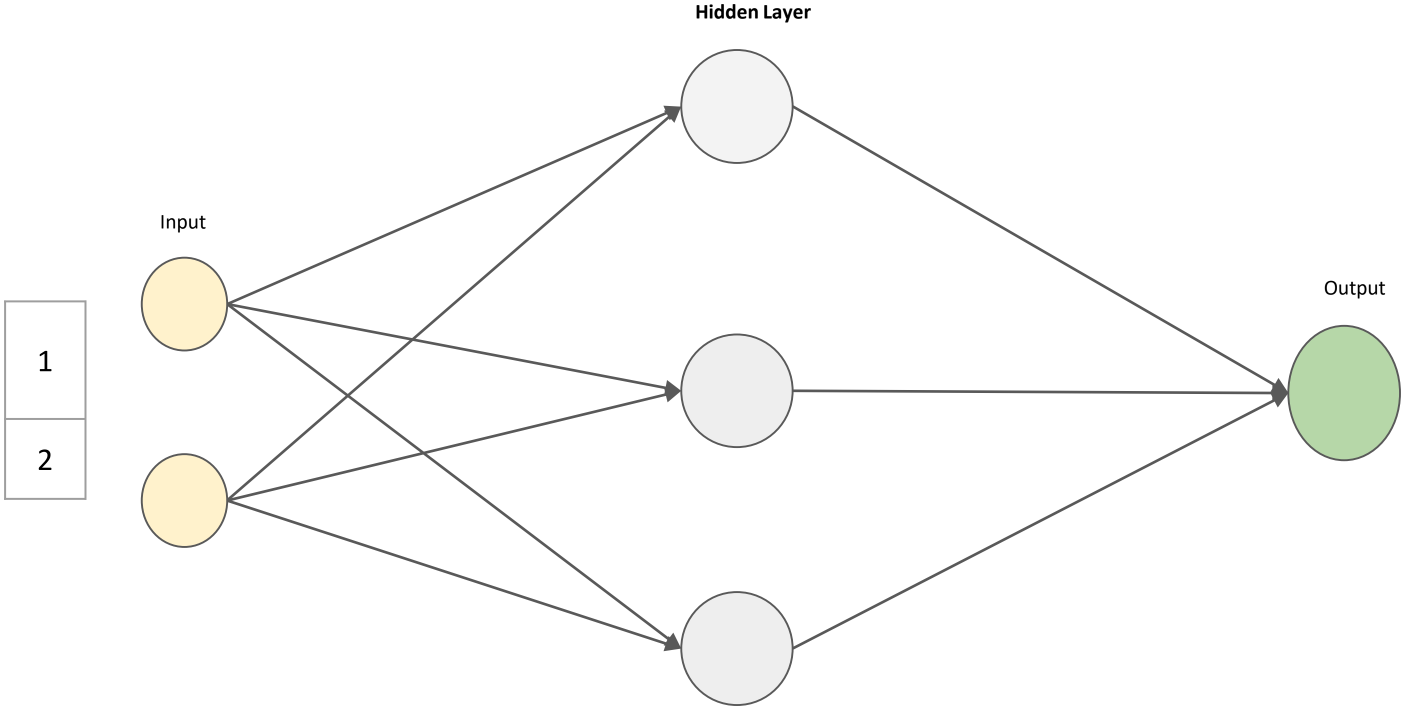


Math of Neural Networks

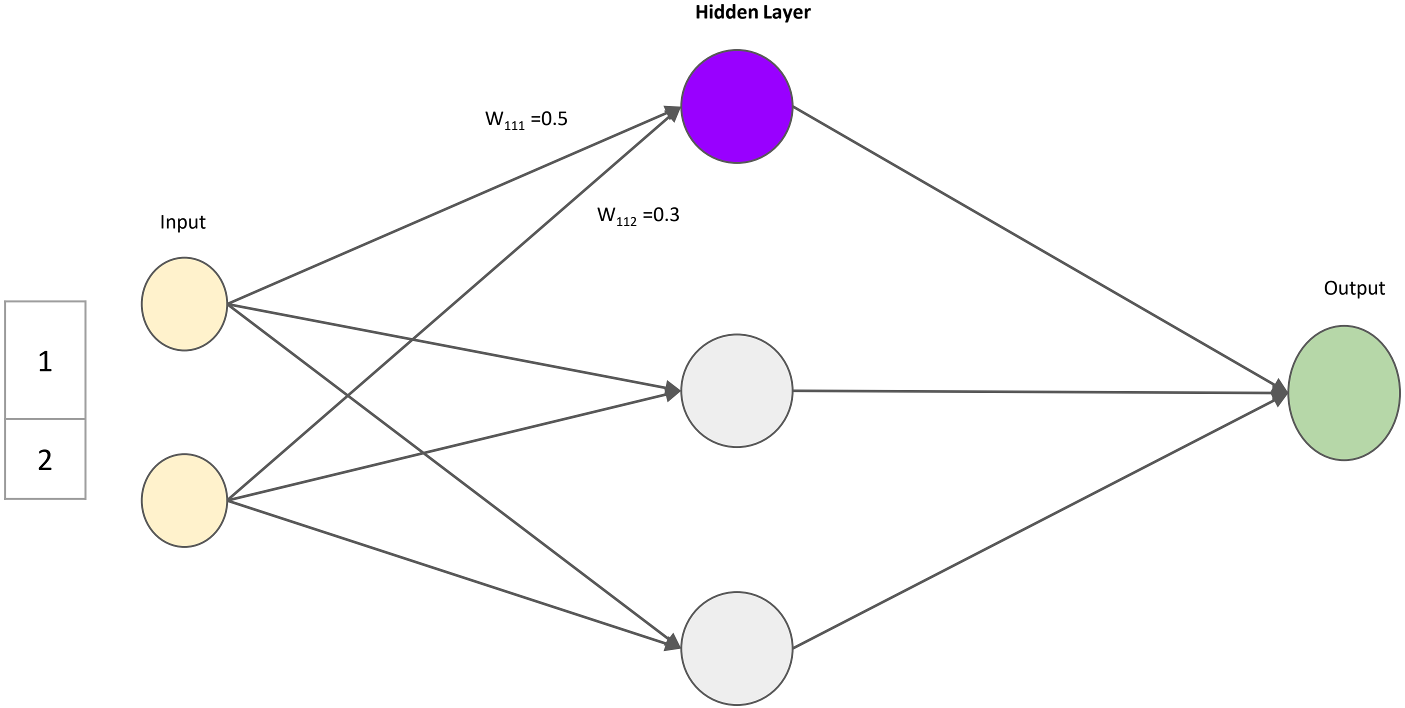
Requirement Data

$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{y}$
1	2	3

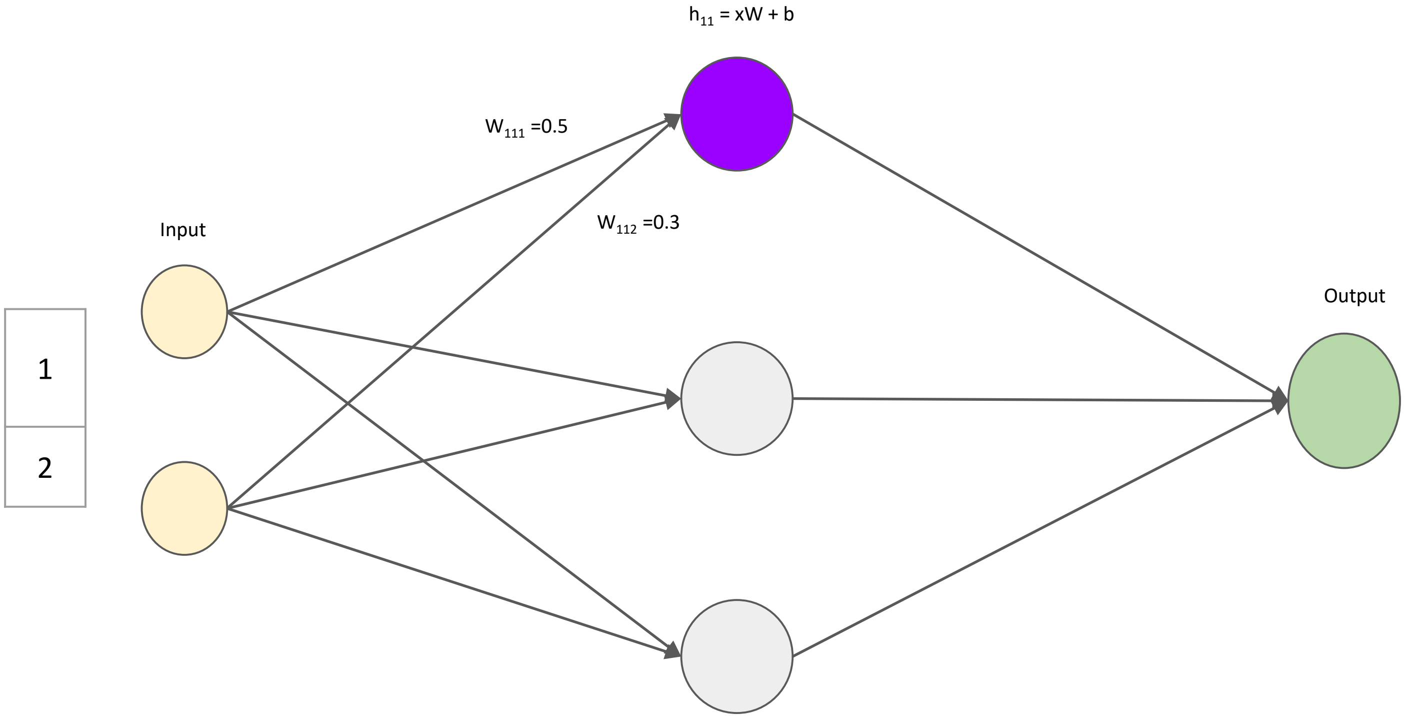
Let's understand how Model will use input features (x_1, x_2) and learn to predict y



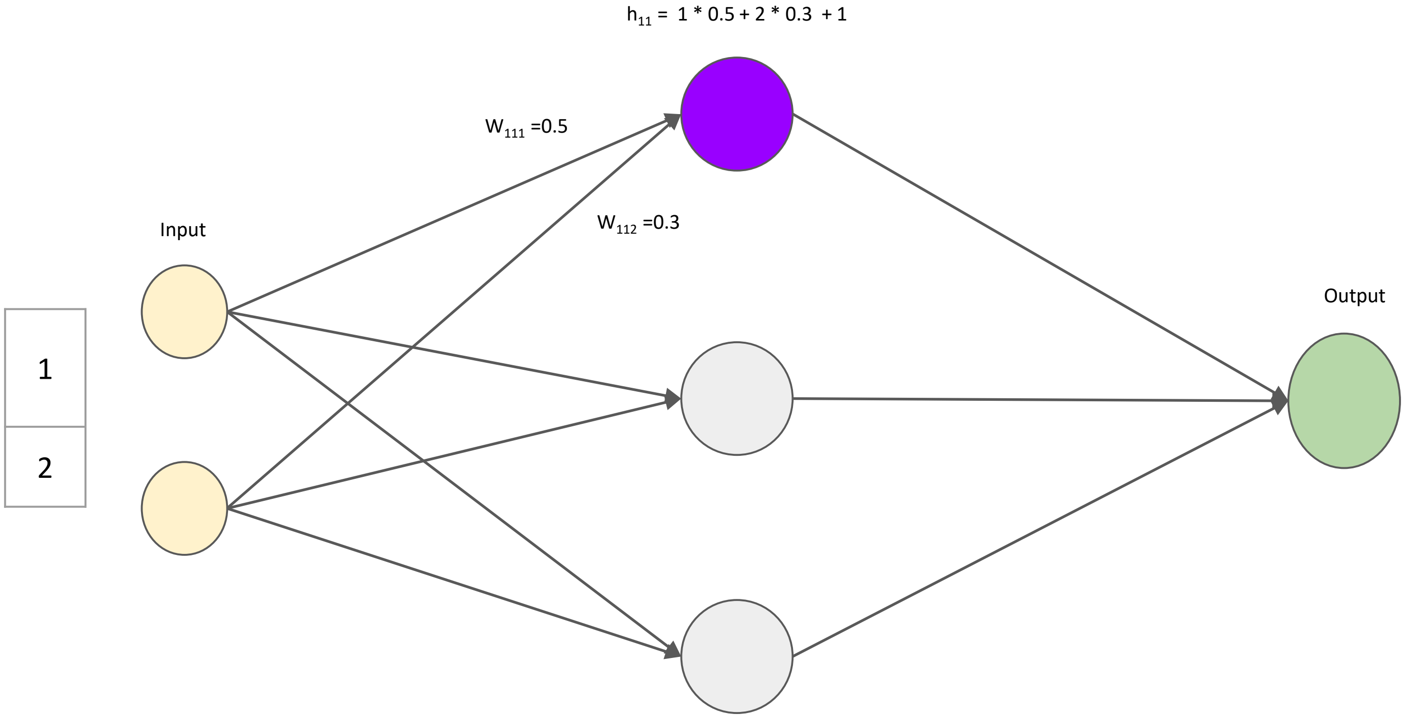
Using a simple neural network with 1 hidden layer



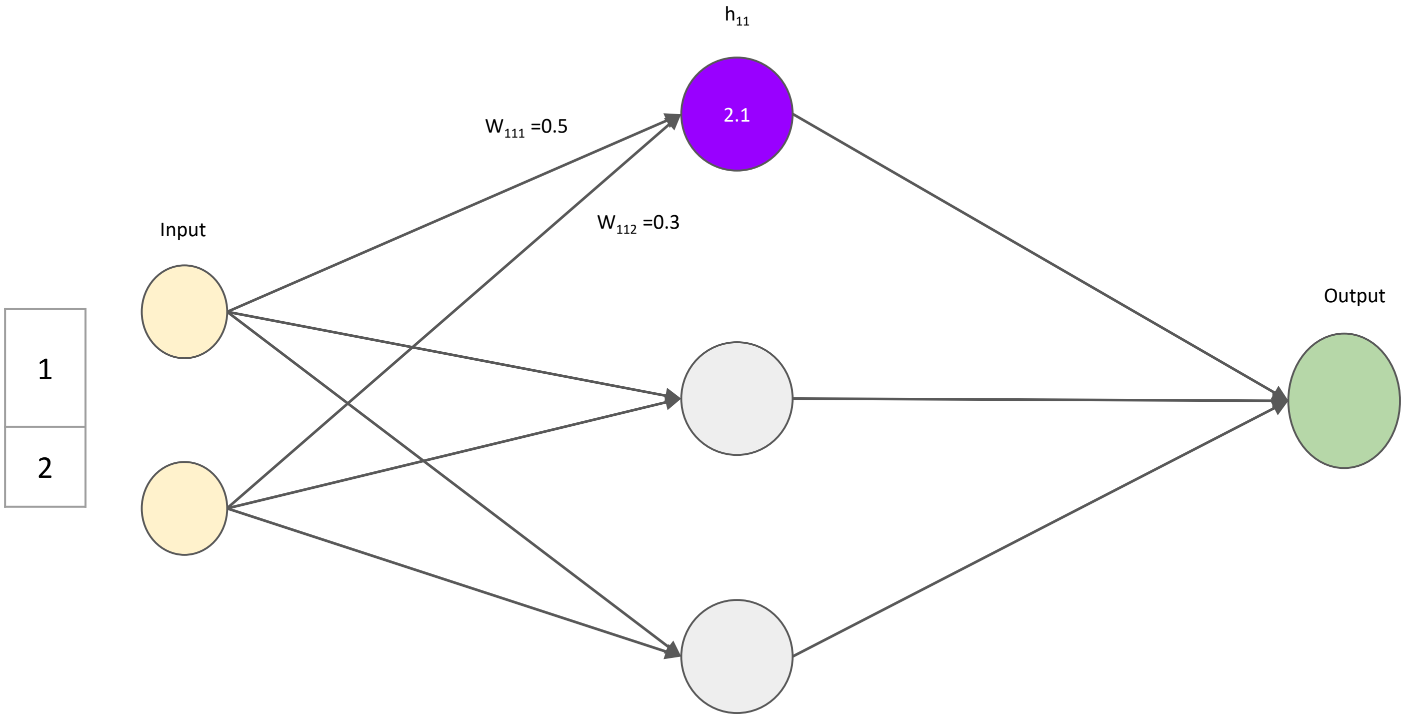
Let’s calculate output for 1st Neuron in hidden layer



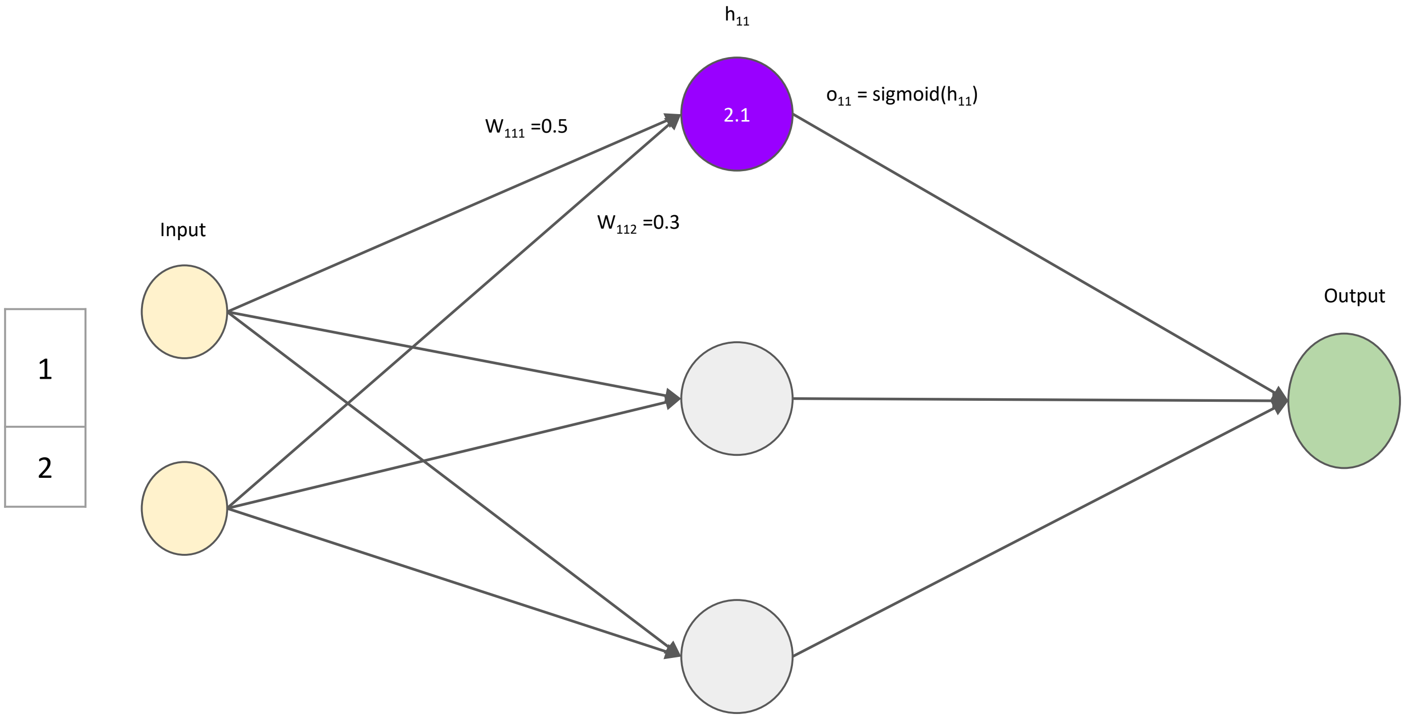
h_{11} = xW + b
W_{111} =0.5
W_{112} =0.3
Input
1
2
Output



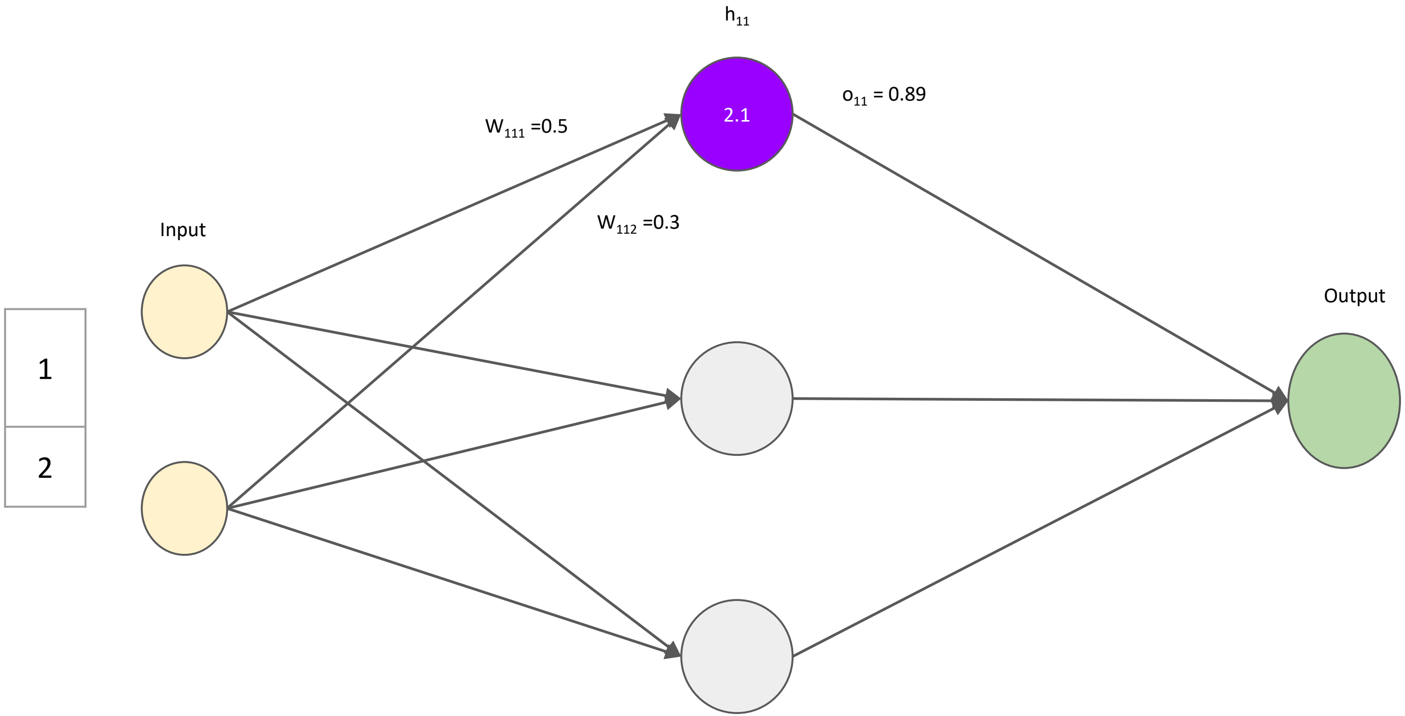
Initializing each weight with random numbers and bias with 1 (can be any number)



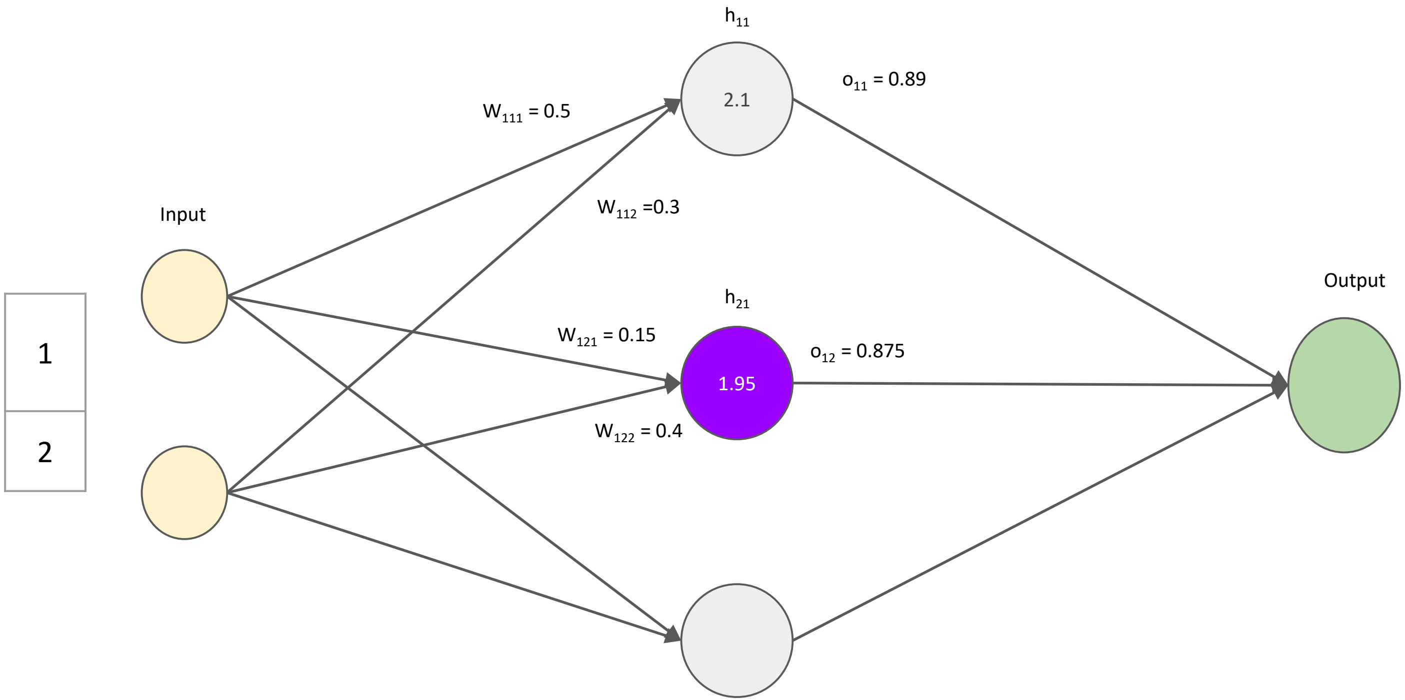
h11
2.1
W111 =0.5
W112 =0.3
Input
1
2
Output



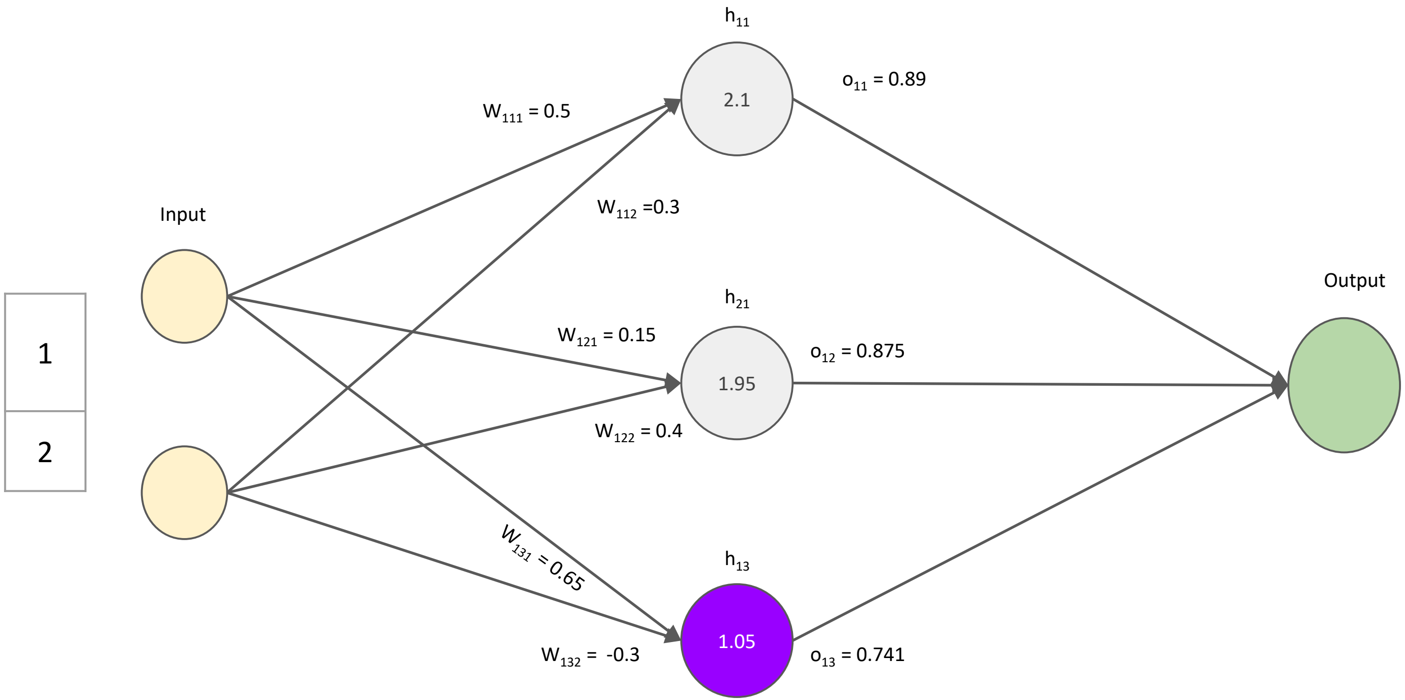
Apply activation (sigmoid) to Neuron’s output



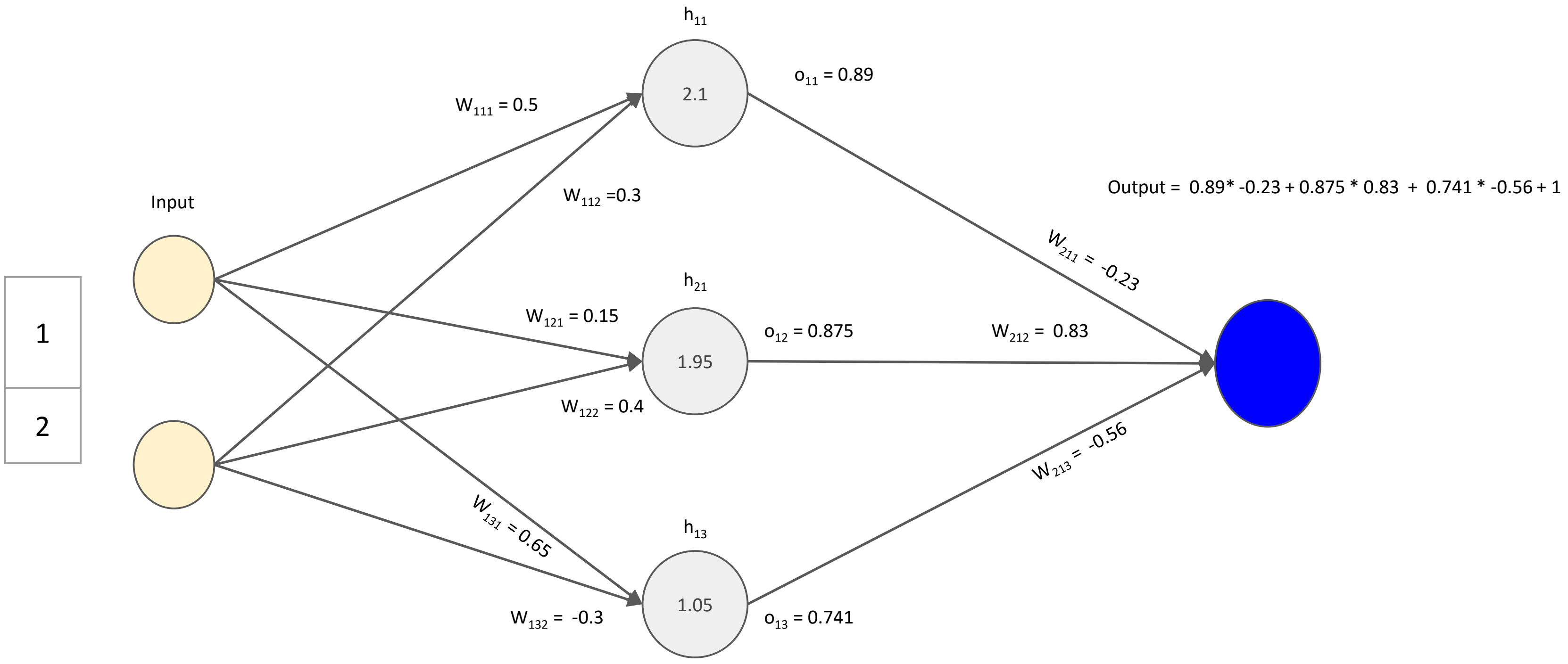
h11
2.1
o11 = 0.89
W111 =0.5
W112 =0.3
Input
1
2
Output



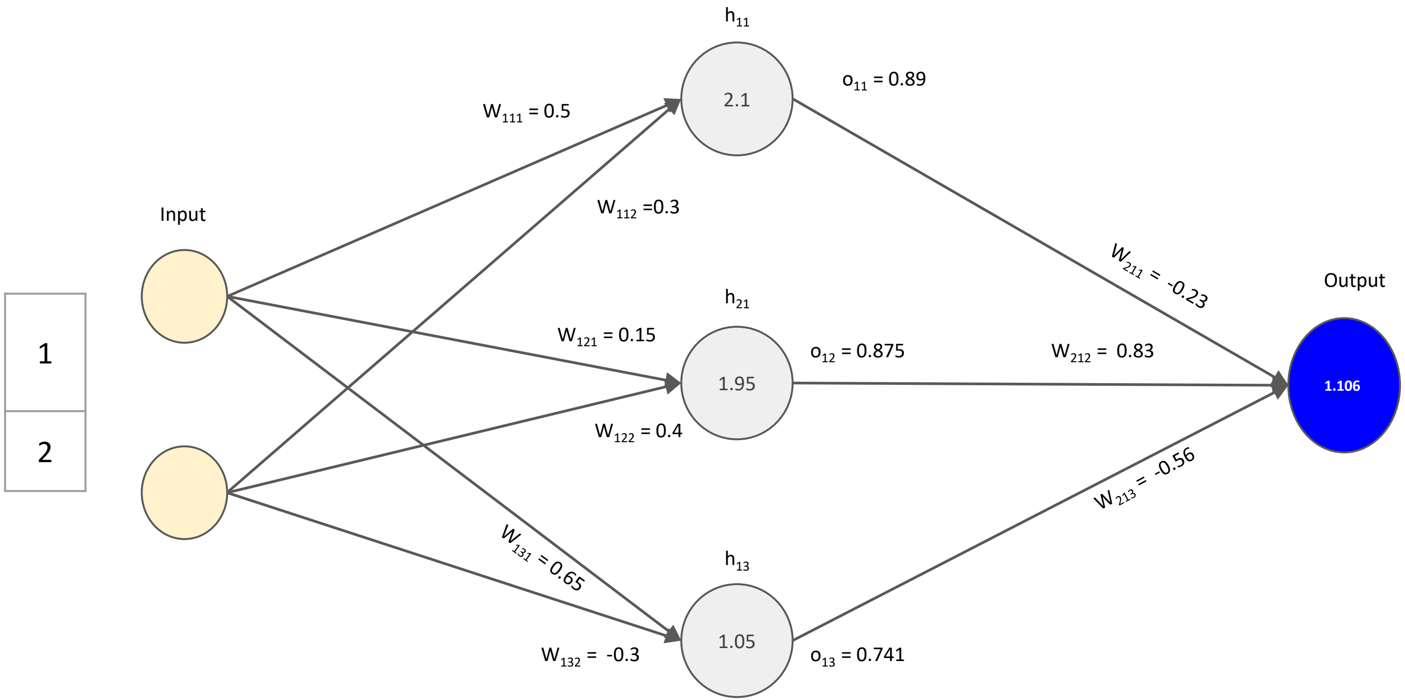
Repeat calculations for 2nd neuron...



...and 3rd Neuron

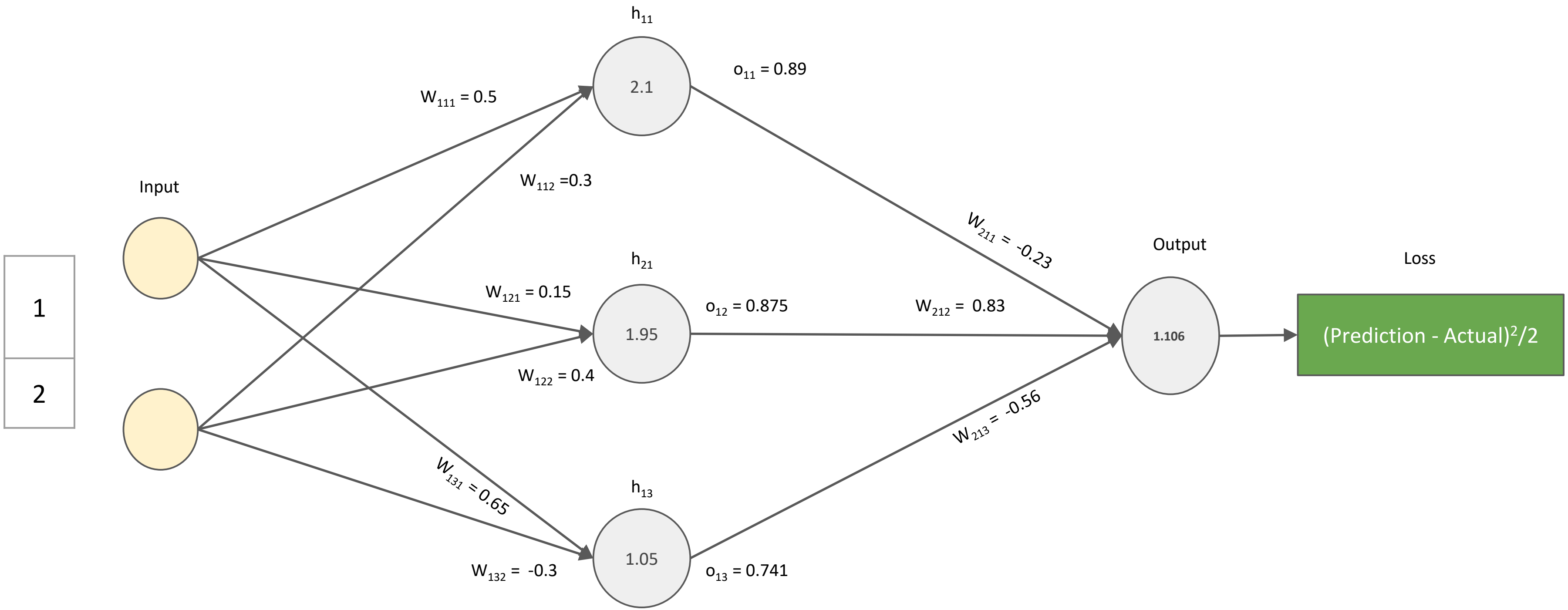


Initialize weights and bias for output layer and calculate output

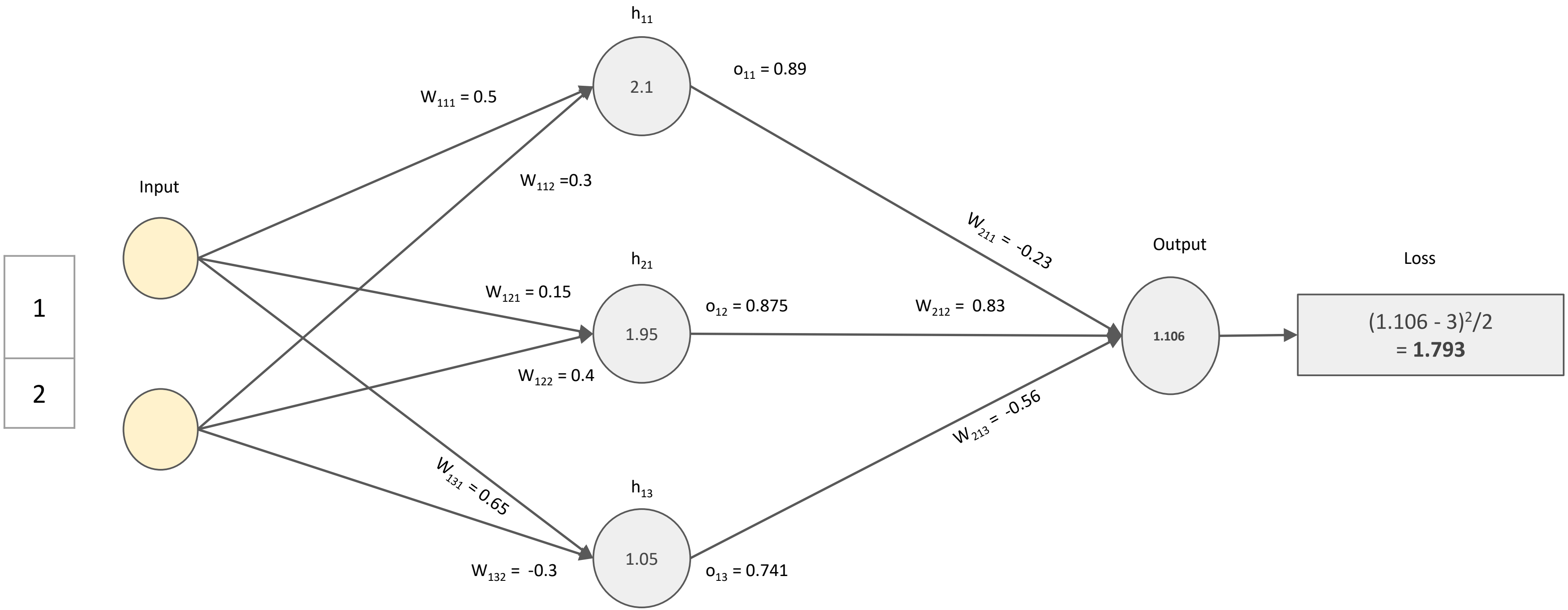


h11
2.1
o11 = 0.89
W111 = 0.5
W112 =0.3
Input
1
2
h21
1.95
W121 = 0.15
o12 = 0.875
W212 = 0.83
W211 = -0.23
Output
1.106
W122 = 0.4
W213 = -0.56
W131 = 0.65
h13
1.05
W132 = -0.3
o13 = 0.741

What’s next?

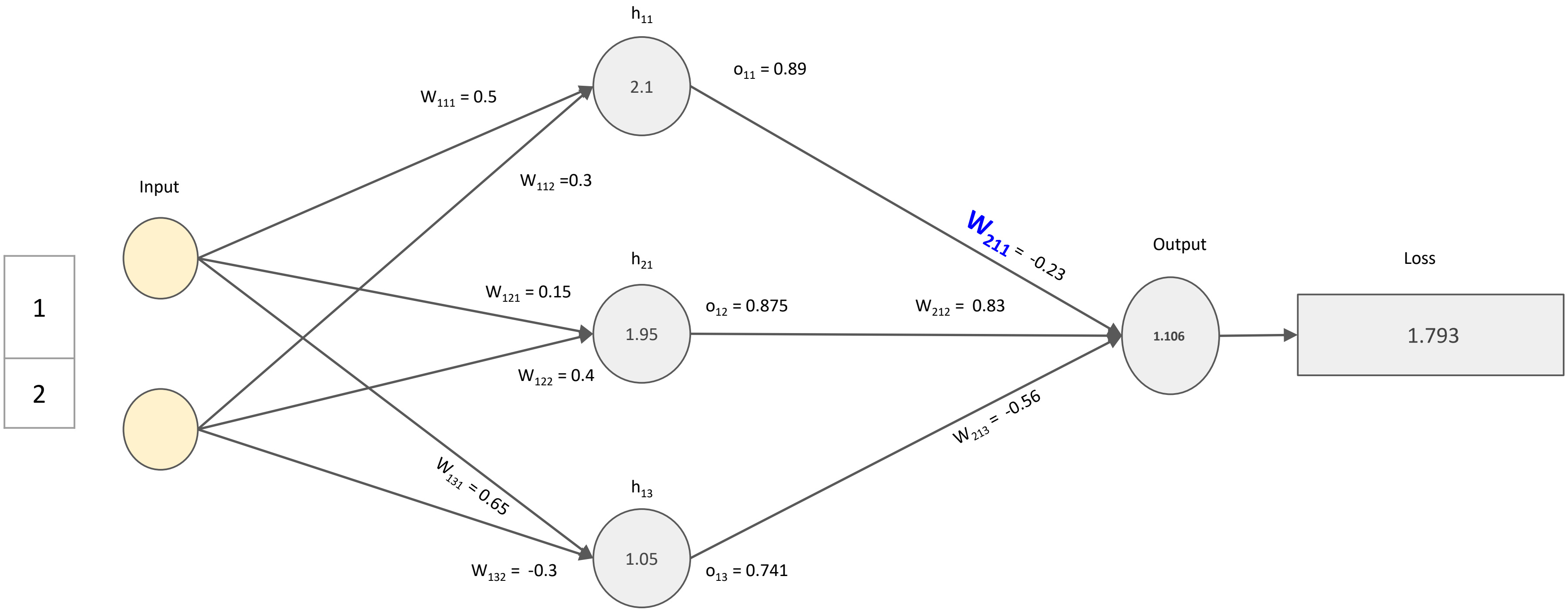


Calculate Loss



Forward Propagation ***i.e*** starting with Input features, calculate Loss

How do we update weights?

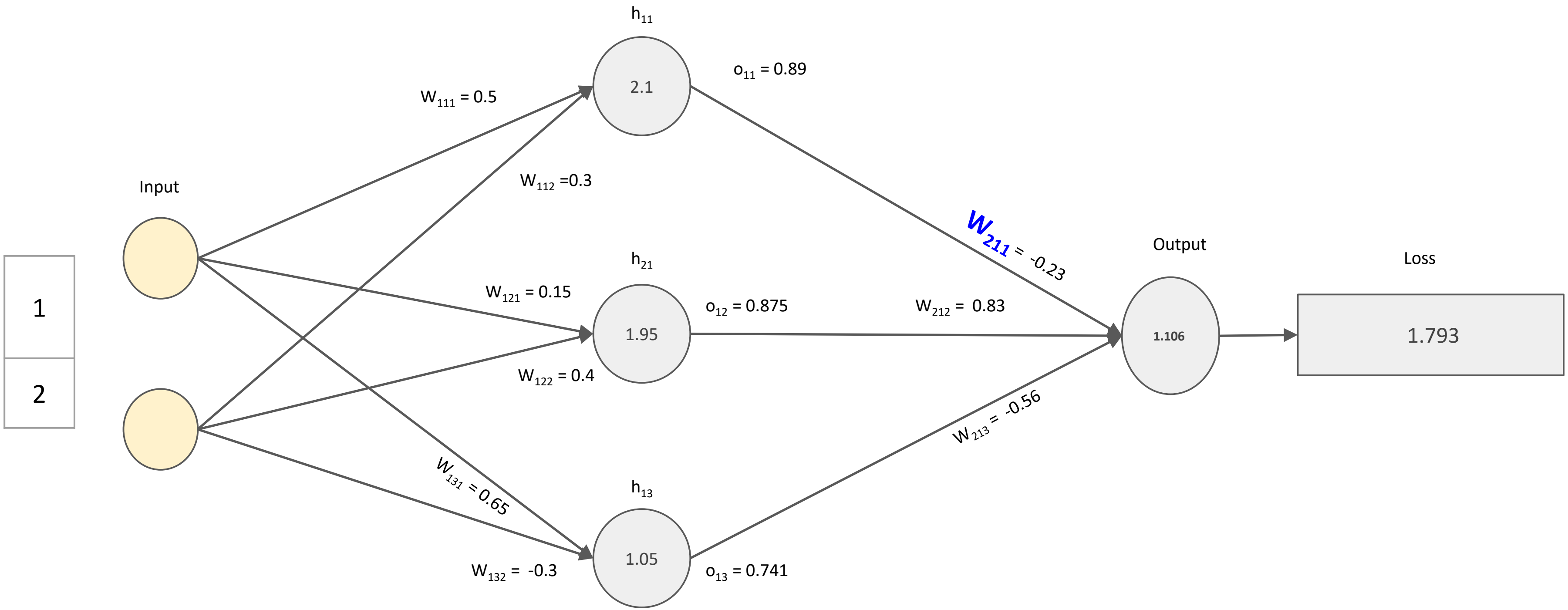


Minimizing Loss using Gradient Descent

Gradient Descent

$$w_{new} = w_{old} - \eta \frac{d}{dw} J(w_{old})$$

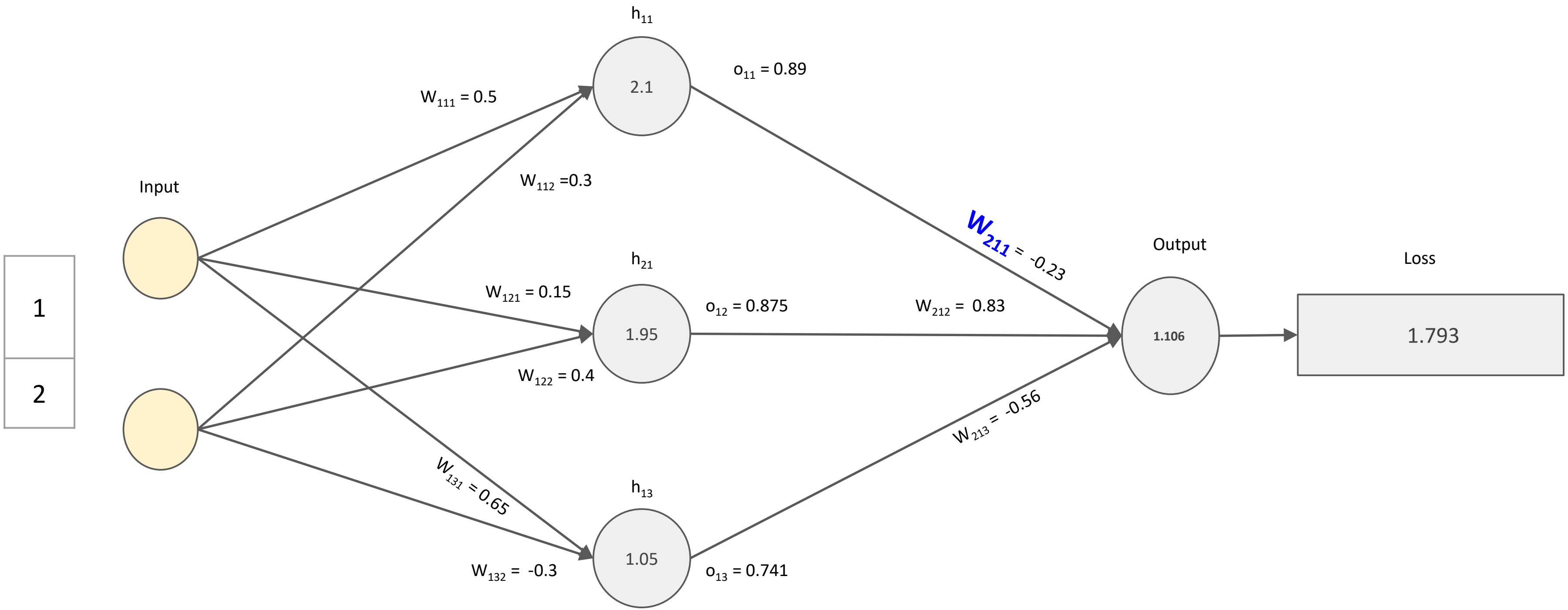
Calculate Gradient for Loss w.r.t each Weight and Bias in the Neural Network



We start with Weights nearest to Output Layer e.g. W_{211}

$$\frac{d\ Loss}{dw_{211}}$$

How is Loss related to w_{211}?



Loss is dependent on Output, Output is dependent on W_{211}

$$Loss = f(output)$$

$$output = g(w_{311})$$

$$Loss(w_{211}) = f(g(w_{211}))$$

$$\textit{Chain rule} \rightarrow \frac{d\,f(g(x))}{dx} = \frac{df}{dg} * \frac{dg}{dx}$$

$$\frac{d\,Loss}{dw_{211}} = \frac{d\,Loss}{d\,Output} * \frac{d\,Output}{dw_{211}}$$

Let’s calculate each Gradient term individually

$$\frac{d\,Loss}{d\,Output}$$

$$Loss = (Actual - Output)^2/2$$

$$\frac{d\,Loss}{d\,Output} = 2 * (Actual - Output) * -1/2$$

$$\frac{d\,Loss}{d\,Output} = 2 * (3 - 1.106) * -1/2 = -1.894$$

$$\frac{d\,Output}{dw_{211}}$$

$$Output = w_{211} * o_{11} + w_{212} * o_{12} + w_{213} * o_{13} + b_{21}$$

$$\frac{d\,Output}{dw_{211}} = o_{11} + 0 + 0 + 0$$

$$\frac{d\,Output}{dw_{211}} = 0.89$$

$$\frac{d\,Loss}{dw_{211}} = \frac{d\,Loss}{d\,Output} * \frac{d\,Output}{dw_{211}}$$

-1.894 * 0.89

= -1.686

What will be new value of w_{211}?

$$w_{211new} = w_{211old} - \eta * \frac{d\,Loss}{dw_{211old}}$$

Learning rate
(assume 0.01 here)

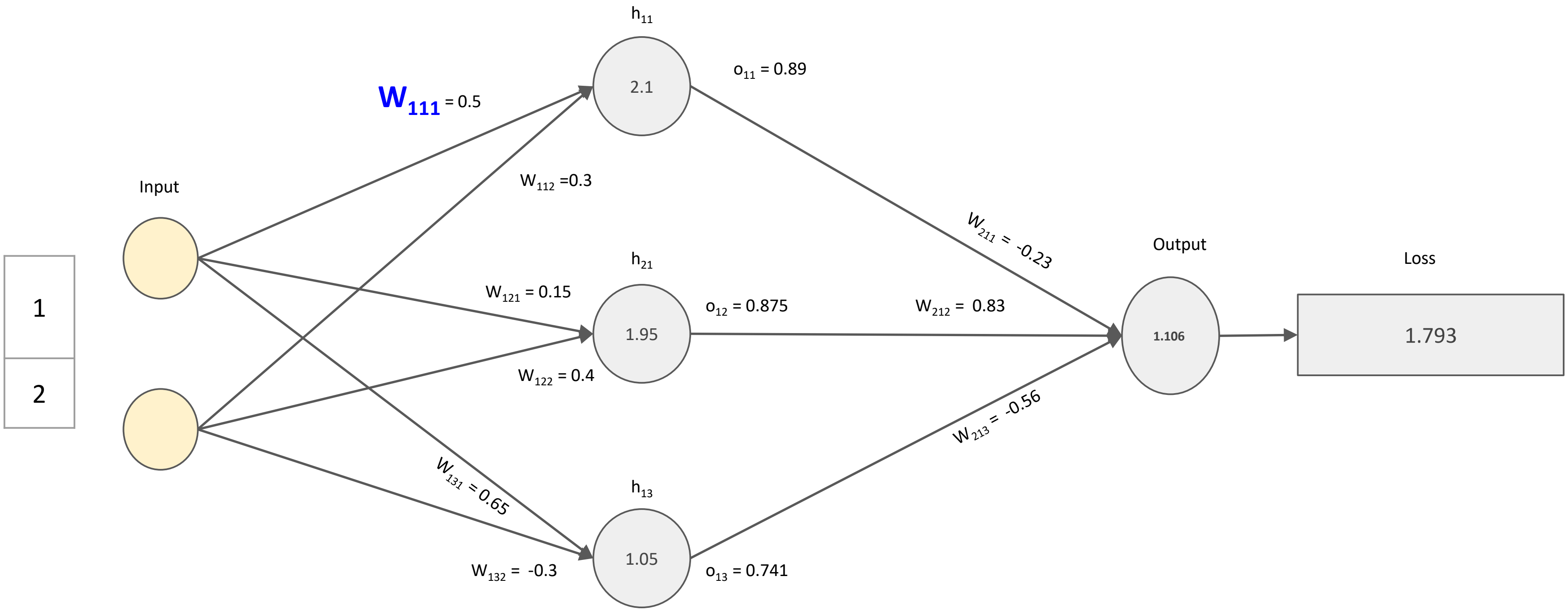
= -0.23 - 0.01 * -1.686

= -0.213

$$w_{212}?$$

$$w_{213}?$$

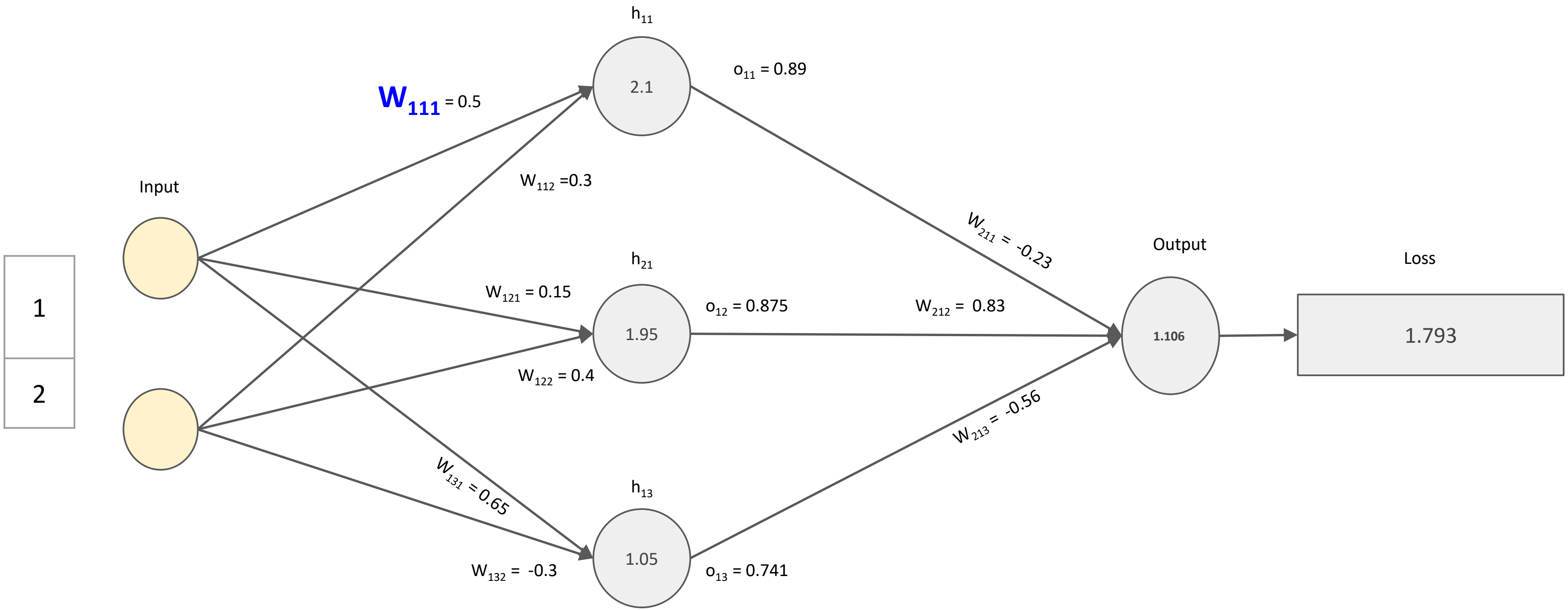
Calculate both in same way as W_{211}



Updating weight for hidden layer e.g W_{111}

$$\frac{d\,Loss}{dw_{111}}$$

How is Loss related to w_{111}?



$W_{111} \rightarrow h_{11} \rightarrow o_{11} \rightarrow \text{Output} \rightarrow \text{Loss}$

$$Loss = f_1(Output)$$

$$Output = f_2(o_{11})$$

$$o_{11} = f_3(h_{11})$$

$$h_{11} = f_4(w_{111})$$

Using Chain Rule

$$\frac{d\,Loss}{dw_{111}} = \frac{d\,Loss}{d\,Output} * \frac{d\,Output}{do_{11}} * \frac{do_{11}}{dh_{11}} * \frac{dh_{11}}{dw_{111}}$$

Already calculated while calculating Gradient for Weights in Output Layer

$$\frac{d\ Output}{do_{11}}$$

$$Output = w_{211} * o_{11} + w_{212} * o_{12} + w_{213} * o_{13} + b_{21}$$

$$\frac{d\ Output}{do_{11}} = w_{211} + 0 + 0 + 0$$

Gradient in Hidden Layer is dependent on Weights in Next Layers

-0.23

$$\frac{do_{11}}{dh_{11}}$$

$$o_{11} = \frac{1}{(1+e^{-h_{11}})}$$

Activation (Sigmoid) function

$$\frac{do_{11}}{dh_{11}} = o_{11}(1 - o_{11})$$

NOTE!

Activation function should be Differentiable *i.e* it should be possible to calculate partial derivative of Activation function

0.89*(1-0.89)

=0.0979

$$\frac{dh_{11}}{dw_{111}}$$

$$h_{11} = w_{111} * ip1 + w_{112} * ip2 + b_{11}$$

$$\frac{dh_{11}}{dw_{111}} = ip1 \ + 0 + 0$$

1

$$\frac{d\,Loss}{dw_{111}} = \frac{d\,Loss}{d\,Output} * \frac{d\,Output}{do_{11}} * \frac{do_{11}}{dh_{11}} * \frac{dh_{11}}{dw_{111}}$$

-1.894 * -0.23 * 0.0979 * 1

= 0.0426

New value of w_{111}

$$= 0.5 - 0.01 * 0.0426$$

$$= 0.499$$

Calculate Gradients and perform Descent for other Weights in Hidden layer

Backpropagation

Propagating Loss or Error to all the layers in the network, starting with Output Layer