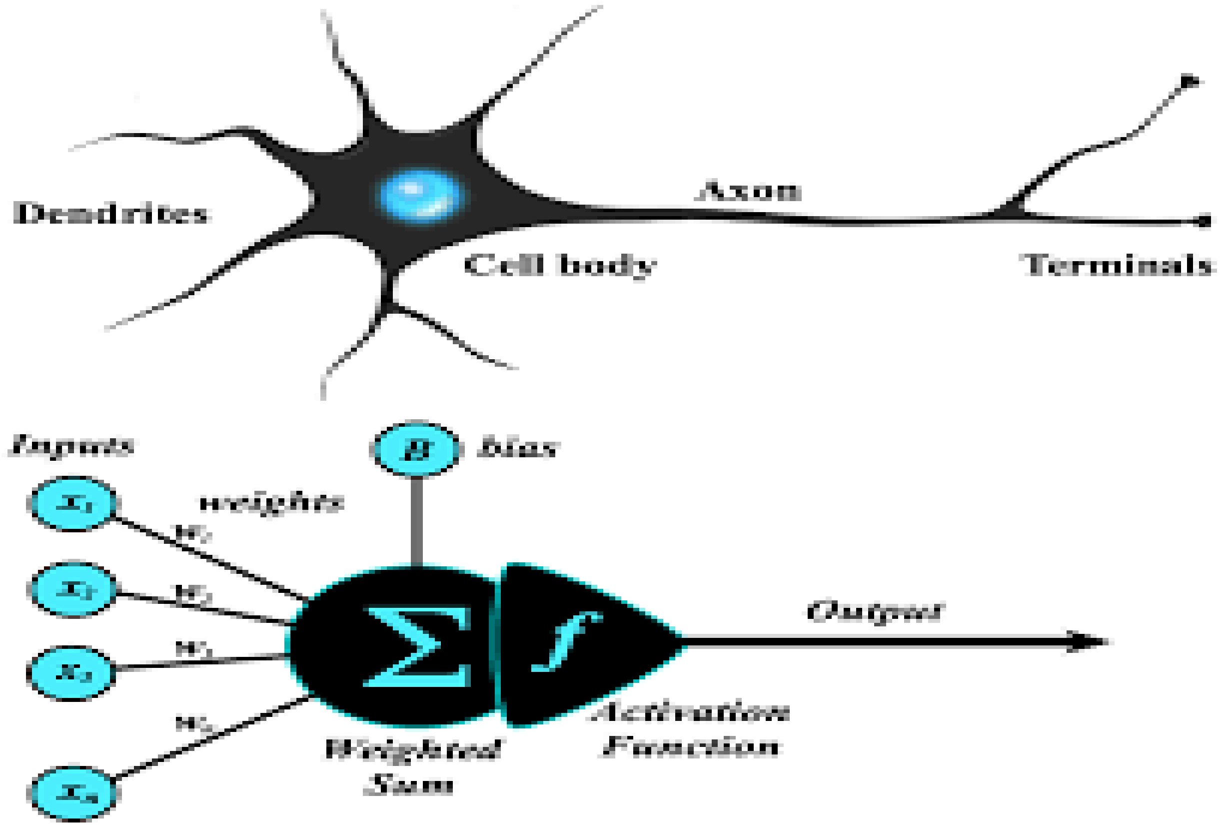
Machine Learning

- 1st Term, 2025/2026
- December 2025

- **Prof. Mohammed A. Al Ghamdi**

Neural Networks

A neural network is a machine learning program, or model, that makes decisions in a manner similar to the human brain, by using processes that mimic the way biological neurons work together to identify phenomena, weigh options and arrive at conclusions.



Neural Networks:

- To understand how the brain works.
- To understand a style of parallel computation inspired by neurons and their adaptive connections.
- To solve practical problems by using novel learning algorithms inspired by the brain.

Neural Networks:

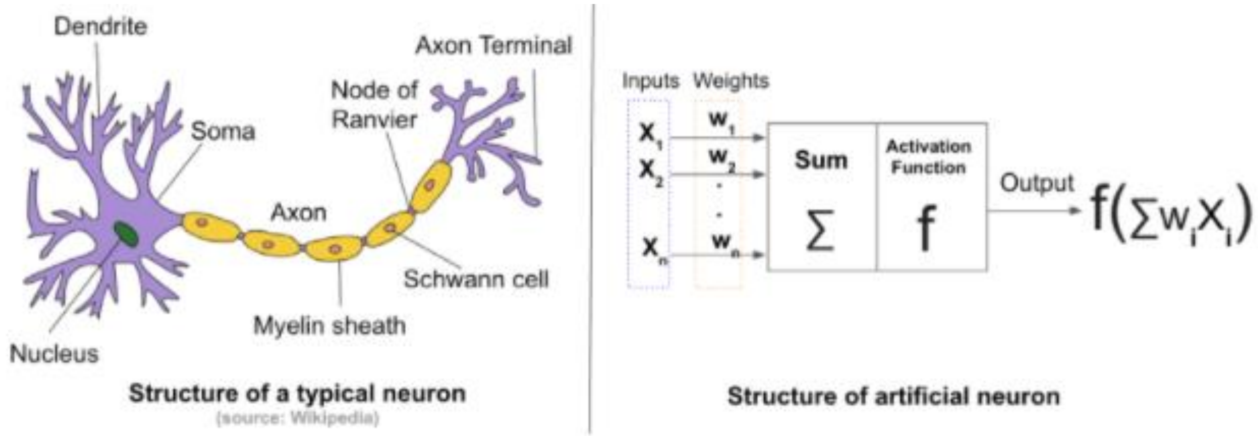
- **Generalization:**

  It can be applied over pure new input.
- **Fault Tolerant:**

  It provides correct output with noisy input.
- **Fast:**

  With multiple way of providing outcomes.

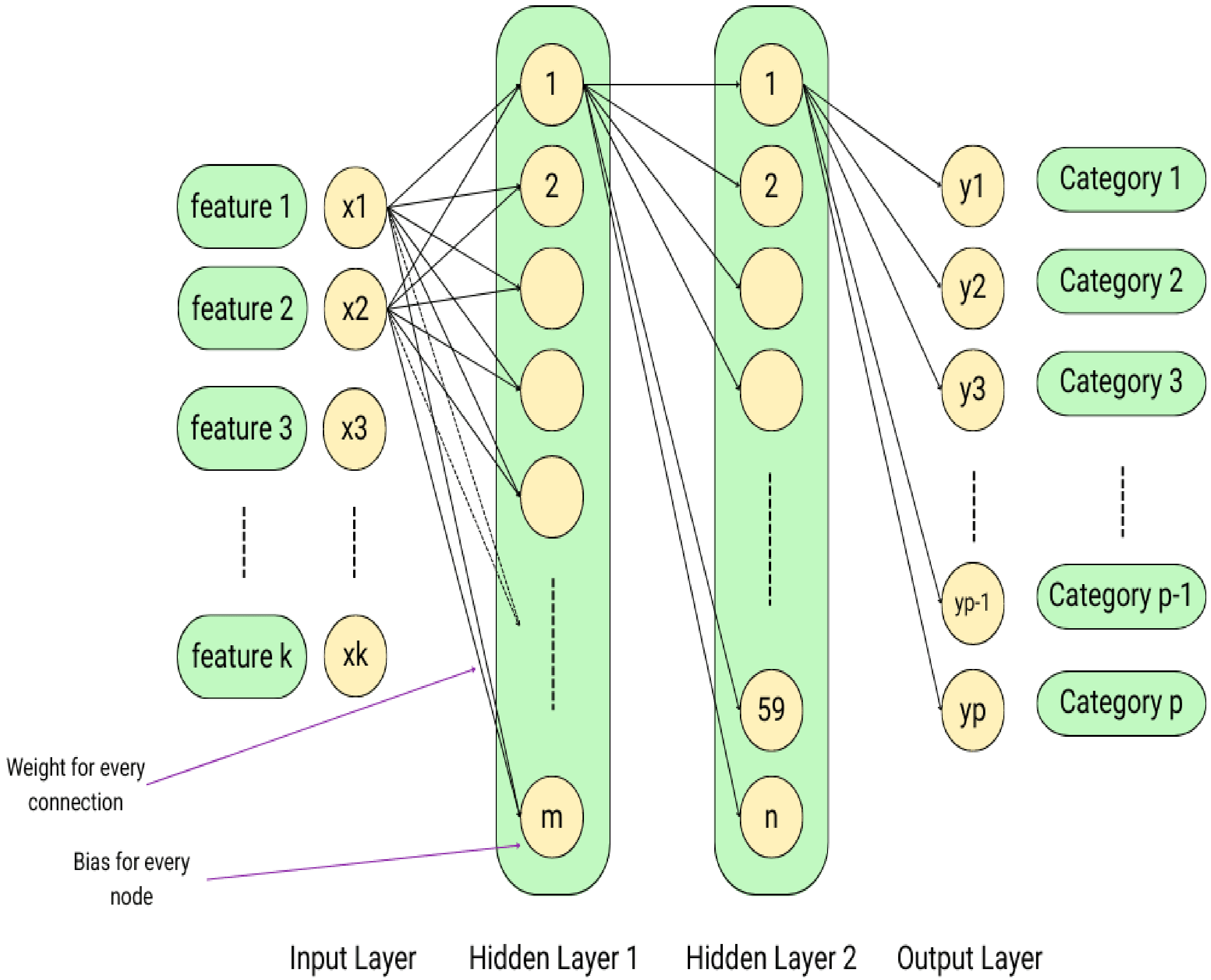
How our brain works:



- **Each neuron we have receives the inputs from other neurons.**
  - A few neurons also connect to receptors.
  - Cortical neurons use spikes to communicate.
- **The effect of each input line on the neuron is controlled by a synaptic weight.**
  - The weights can be positive or negative.
- **The synaptic weights adapt so that the whole network learns to perform useful computations.**
  - Recognizing objects, understanding language, making plans, controlling the body.
- **You have about 10^{11} neurons each with about 10^4 weights.**
  - A huge number of weights can affect the computation in a very short time. Much better bandwidth than a workstation.

NN Components:

- **Layers**
- **Neurons**
- **Weight**
- **Bias**
- **Activation Function**



NN Components (Layers):

- **Input Layer:**
  - The input layer is the first layer of any Neural Network.
  - It represents the input data to the network.
- **Output Layer:**
  - The output layer of a Neural Network represents the final predictions generated by the network.
  - The number of neurons in this layer corresponds to the number of output desired for a given input.
  - In regression problem, it is expected to have a single value, so there will be one neuron in the output layer.
  - In Classification tasks, where multiple classes are possible, there will be multiple neurons (one for each class).

NN Components (Layers & Neurons):

- **Hidden Layer:**
  - The hidden layer exist between the input and output layers.
  - It is to decode the relations between the input and output layers.
  - In NN, we can have more than one hidden layer to achieve more accuracy (but think about the cost).
  - we can have more than one neurons to be more sufficient to achieve our target accuracy).
- **Neurons:**
  - Neurons constitute every layer in the NN, including input, output, and hidden layers.
  - It is like the nucleus of brain cells. Where each neuron contains a bias parameter that the NN learns and adjust during the training process.

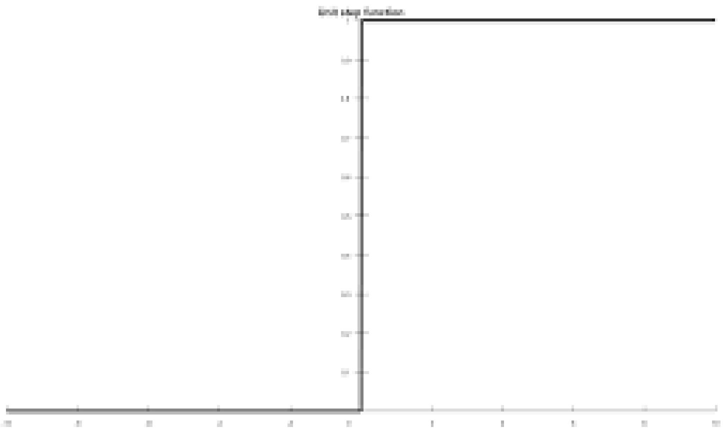
NN Components (Bias & Weight):

- **Bias:**
  - The bias parameter values are typically initialized with random numbers.
  - The NN fine-tunes them to minimize the differences between the computed and actual output.
- **Weight:**
  - Each neuron in one layer is connected to every neuron in the adjacent layers.
  - These connections are represented by a weight value, which determines the importance of that connection.
  - The weight values are the trainable parameters that the NN learns by iterating over the training datasets.

NN Components (Activation Function):

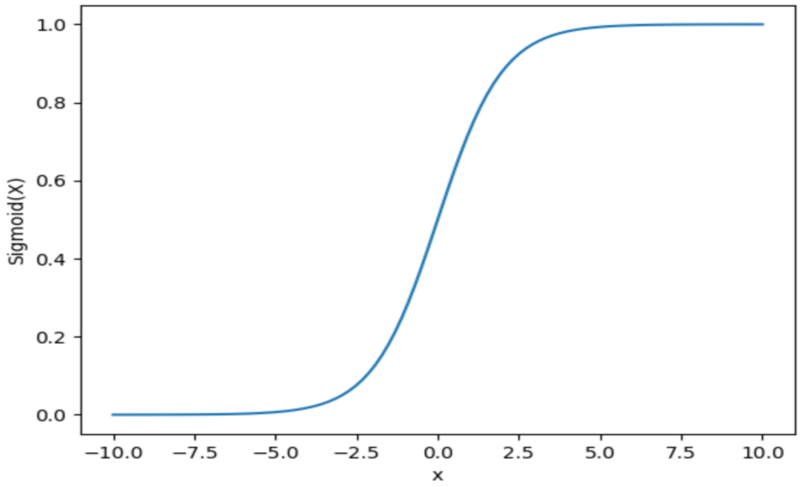
- An activation function in the context of neural networks is a mathematical function applied to the output of a neuron.
- The purpose of an activation function is to introduce non-linearity into the model, allowing the network to learn and represent complex patterns in the data.
- *Without non-linearity, a neural network would essentially behave like a linear regression model, regardless of the number of layers it has.*
- A neural network without an activation function is essentially just a linear regression model.
- *The activation function decides whether a neuron should be activated or not by calculating the weighted sum and further adding bias to it.*

Activation Function:



Binary Step Function

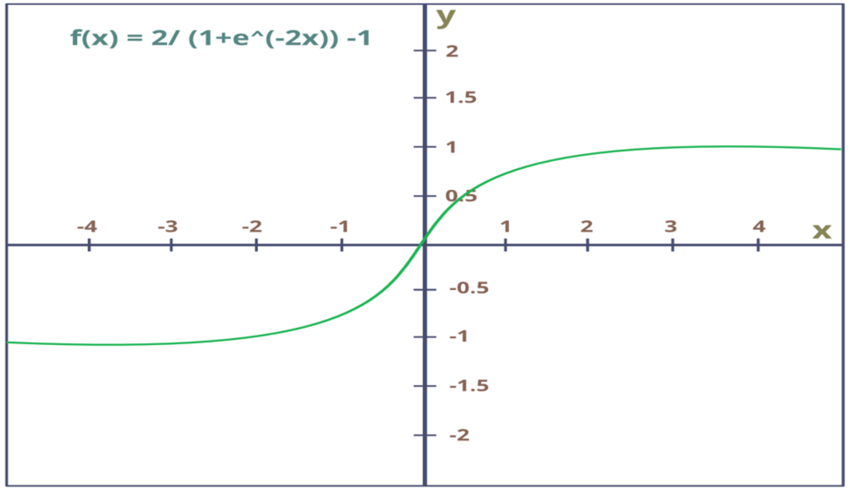
$$f(x) = \begin{cases} 0 & for\ x < 0 \\ 1 & for\ x \geq 0 \end{cases}$$



Sigmoid Function

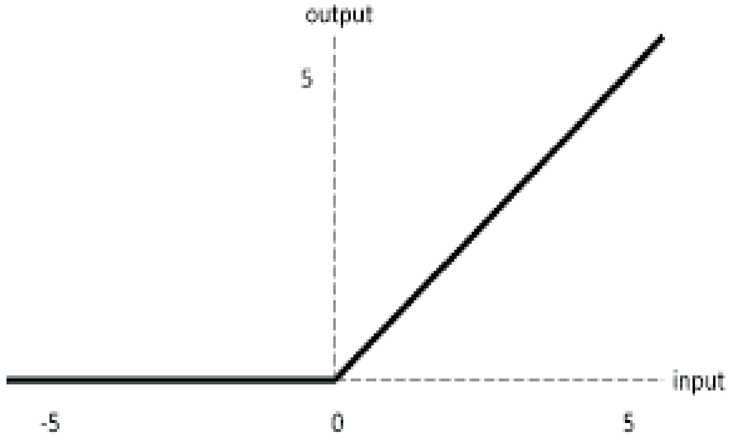
$$f(x) = \frac{1}{1+e^{-x}}$$

Activation Function:



TanH Function

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

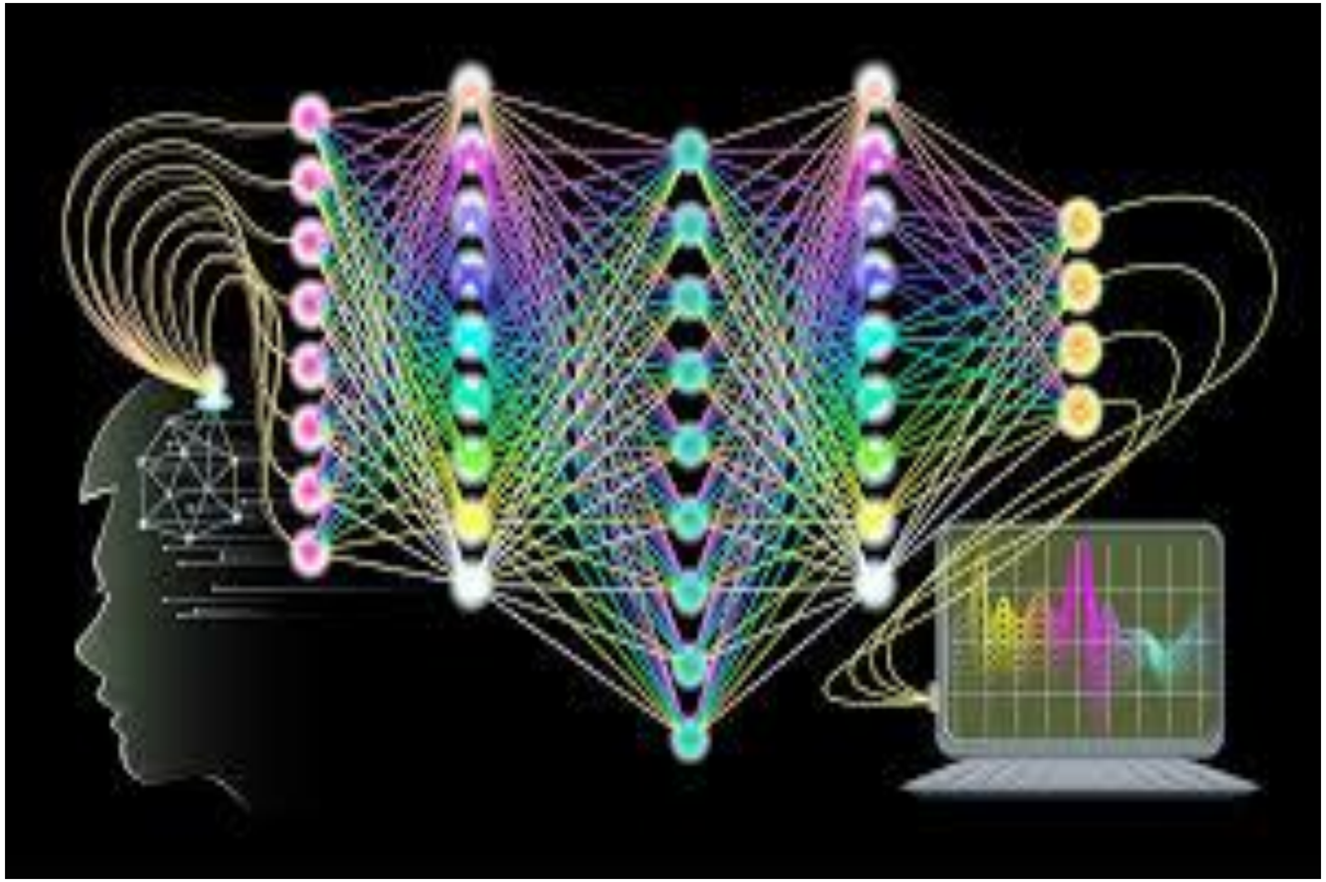


ReLU (rectified linear unit)

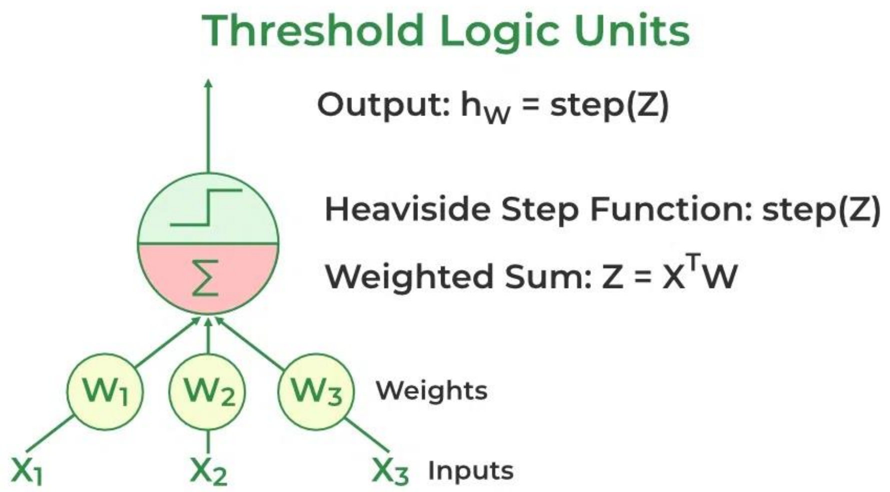
$$f(x) = Max(0,x)$$

NN Classes:

- Perceptron.
- Feedforward neural networks, or multi-layer perceptron (MLPs).
- Convolutional neural networks (CNNs).
- Recurrent neural networks (RNNs).



Perceptron:



- It is one of the simplest Artificial neural network architectures.
- It is the simplest type of feedforward neural network.
- It consisting of a single layer of input nodes that are fully connected to a layer of output nodes.
- It uses threshold logic units (TLU).
- It has the following components:
  - Input Features, Weights, Summation Function, Activation Function, Output, Bias, Learning Algorithm (Weight Update Rule)

Perceptron Example

- Suppose that we need to apply AND function by using perceptron.
- AND function has two input (x_1 and x_2), they are either 0 or 1.
- The output (y) is always 0, except when the both input are 1s.
- See the schedule.
- Activation Functions:

  if $s_j \geq 0$ *then* $x_j = 1$,

  if $s_j < 0$ *then* $x_j = 0$.
- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Perceptron Example

- We suppose that the used weights are as follow:

  $w_1 = 0.1$

  $w_2 = 0.3$

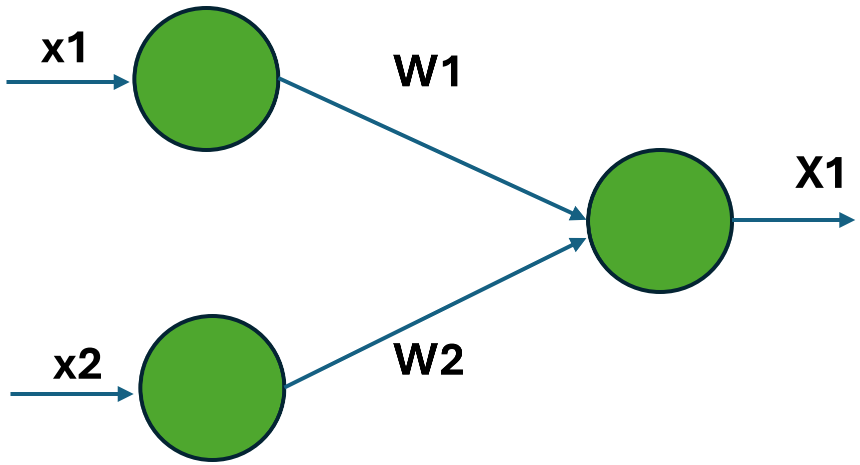
- Now, we calculate the model for the first input (x_1=0 and x_2=0).

  $S_j = ((\sum x_i * wi) + b)$

  $S_j = (0 * 0.1) + (0 * 0.3) + (-0.1) = -0.1$

  Based on the Activation function;

  X = 0.



- Activation Functions:

  if $s_j \geq 0$ *then* $x_j = 1$,

  if $s_j < 0$ *then* $x_j = 0$.

- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Perceptron Example

- Now, we need to check the weights, weather to change it or not.

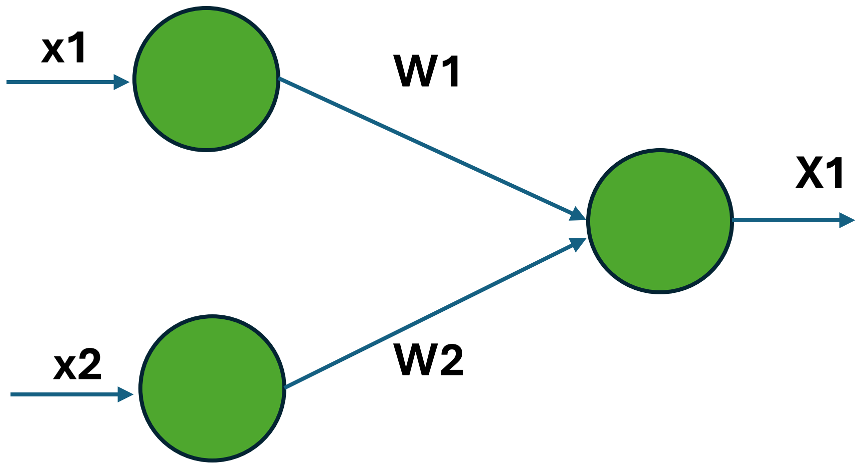
  if the predicted output is the same as the original output => No change.

  if the predicted output is **NOT** the same as the original output => Do Change.
- In this case, there is no need to change.
- Next, calculate the second input (x_1=0, and x_2=1):

  $S_j = (0 * 0.1) + (1 * 0.3) + (-0.1) = 0.3 - 0.1 = 0.2$

  Based on the Activation function;

  X = 1.



- Activation Functions:

  if $s_j \geq 0$ *then* $x_j = 1$,

  if $s_j < 0$ *then* $x_j = 0$.
- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Perceptron Example

- Now, we need to change the weights - as the predicted output (po = 1) is not equal to the original output (oo = 0) - as follow:

  $w_{new} = w_{old} + bias * (oo - po) * x_i$.

  $\Rightarrow w_{1new} = 0.1 + (-0.1) * (0-1) * 0 = 0.1$

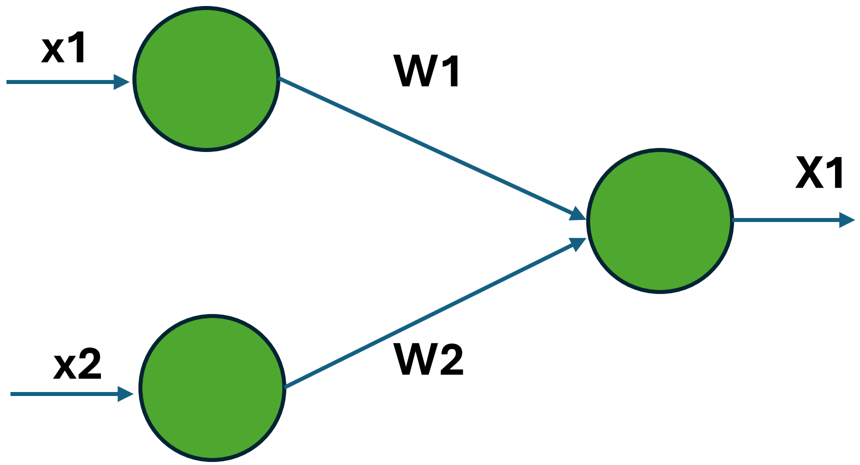
  $\Rightarrow w_{2new} = 0.3 + (-0.1) * (0-1) * 1 = 0.4$

- Next, calculate the third input (x_1=1, and x_2=0):

  $S_j = (1 * 0.1) + (0 * 0.4) + (-0.1) = 0$

  Based on the Activation function;

  X = 1.



- Activation Functions:

  if $s_j \geq 0$ then $x_j = 1$,

  if $s_j < 0$ then $x_j = 0$.

- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	**0**	**0**
1	1	1

Perceptron Example

- Now, we need to change the weights, as the predicted output (po = 1) is not equal to the original output (oo = 0) as follow:

  $w_{new} = w_{old} + bias * (oo - po) * x_i$.

  $\Rightarrow$ $w_{1new} = 0.1 + (-0.1) * (0-1) * 1 = 0.2$

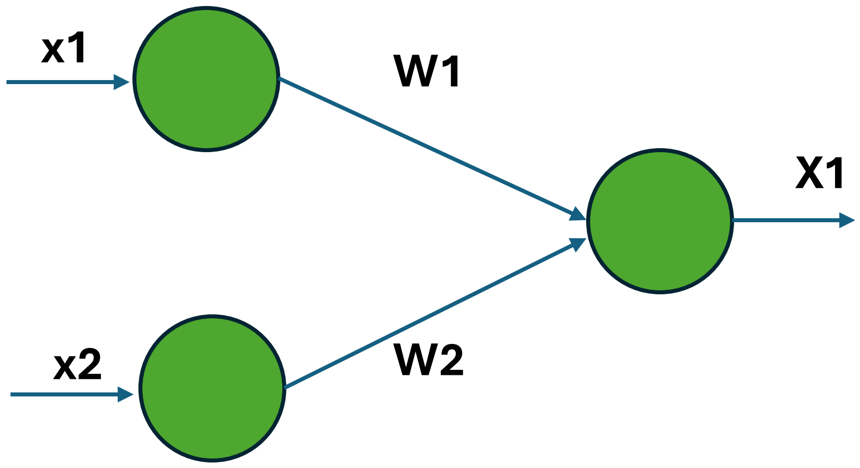
  $\Rightarrow$ $w_{2new} = 0.4 + (-0.1) * (0-1) * 0 = 0.4$

- Next, calculate the fourth input (x_1=1, and x_2=1):

  $S_j = (1 * 0.2) + (1 * 0.4) + (-0.1) = 0.6 – 0.1 = 0.5$

  Based on the Activation function;

  X = 1.



- Activation Functions:

  if $s_j \geq 0$ *then* $x_j = 1$,

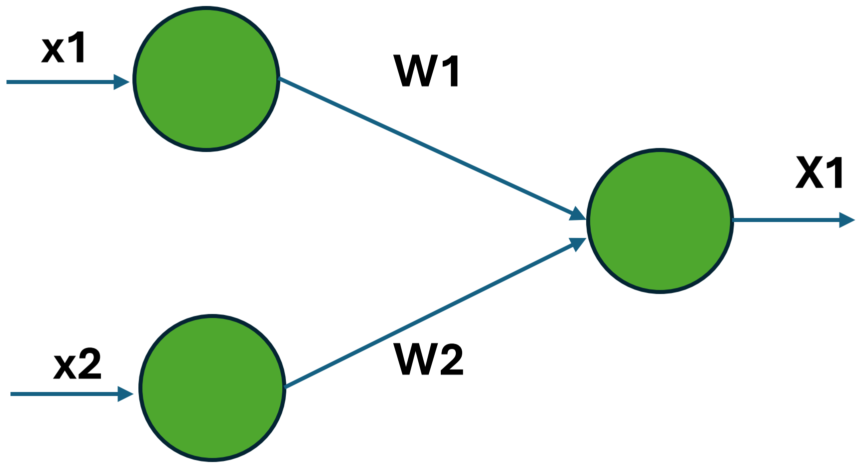
  if $s_j < 0$ *then* $x_j = 0$.

- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	0	0
1	**1**	**1**

Perceptron Example

- Now, we don't need to change the weights, as the predicted output (po = 1) is equal to the original output (oo = 1).



- Activation Functions:

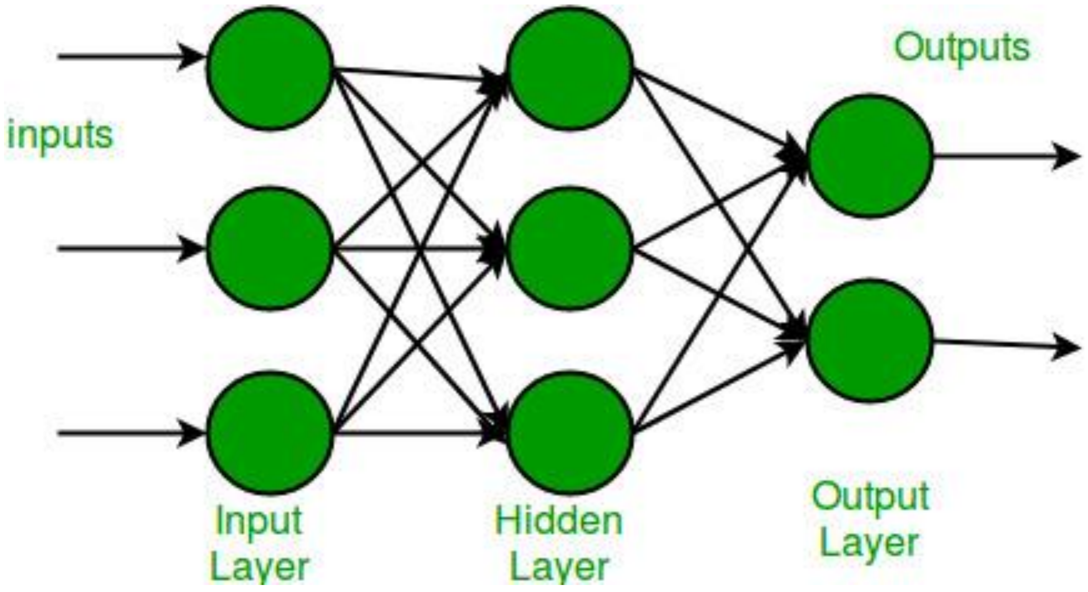
  if $s_j \geq 0$ *then* $x_j = 1,$

  if $s_j < 0$ *then* $x_j = 0.$

- bias = -0.1

X_1	X_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Feedforward NN or Multi-Layer Perceptron (MLP):



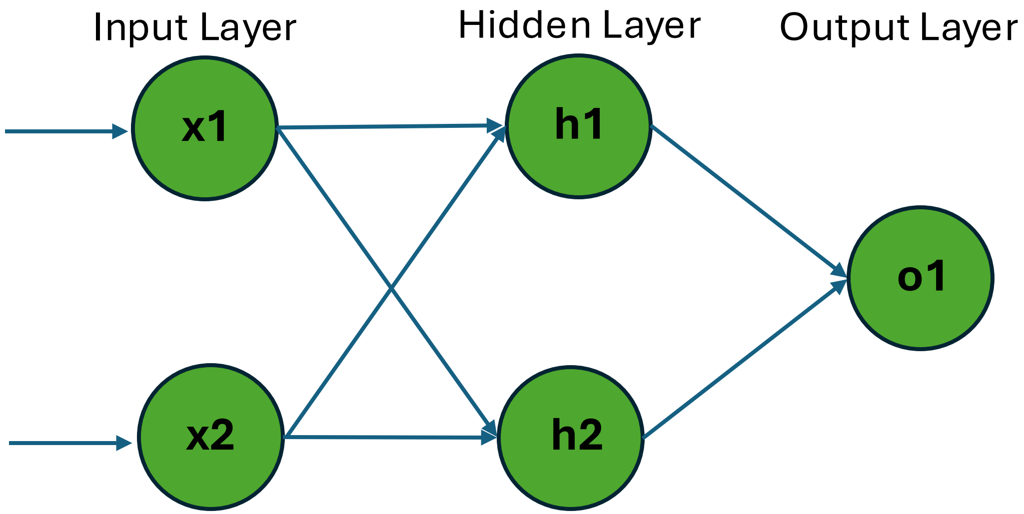
- Multi-layer perception is also known as MLP.
- It transforms any input dimension to the desired dimension.
- A multi-layer perception is a neural network that has multiple layers.
- It has one output layer with a single node for each output and it can have any number of hidden layers, and each hidden layer can have any number of nodes.
- The nodes in the input layer take input and forward it for further process, in the in the hidden layer.
- The hidden layer processes the information and passes it to the output layer.
- Every node in the multi-layer perception uses a sigmoid activation function.

Feedforward NN Example

- Firstly, the input and weights are presented.
- Secondly, the hidden components is calculated then apply Activation Function:

  $f(w * x + b)$ -> *(Linear Equation)*
- Thirdly, the output is computed via:

  $o_1 = f((w_1 * h_1) + (w_2 * h_2) + b)$
- Fourthly, the sigmoid function is applied over the result.
- Finally, the threshold is determined to decide the result.



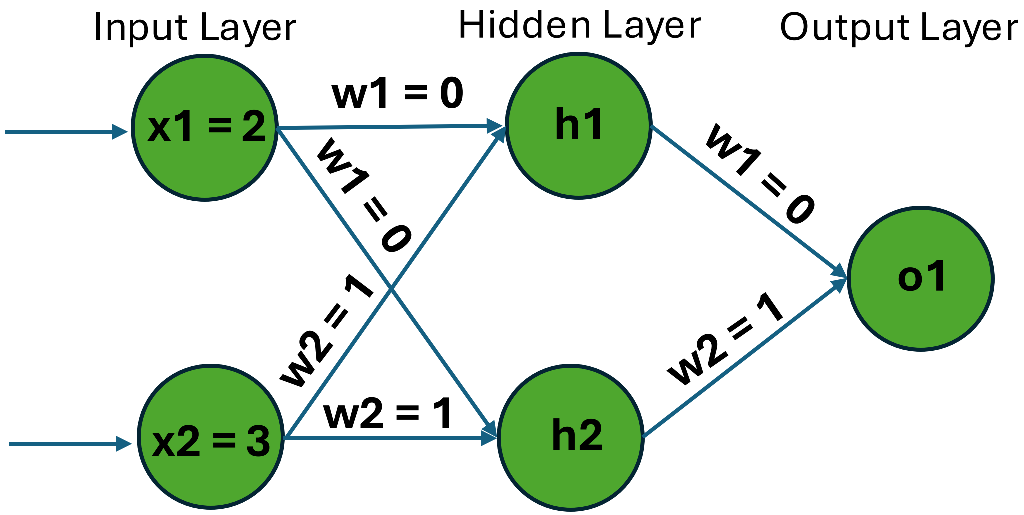
- $x = [2,3]$
- $w = [0,1]$
- $b = 0$
- *Activation Function: Sigmoid*

Feedforward NN Example

- Firstly, the input and weights are presented.
- Secondly, the hidden components is calculated then apply Activation Function:

  *f(w * x + b) -> (Linear Equation)*
- Thirdly, the output is computed via:

  $o1 = f((w_1 * h_1) + (w_2 * h_2) + b)$
- Fourthly, the sigmoid function is applied over the result.
- Finally, the threshold is determined to decide the result.



- $x = [2,3]$
- $w = [0,1]$
- $b = 0$
- *Activation Function: Sigmoid*

Feedforward NN Example

Feedforward NN Example

- Firstly, the input and weights are presented.
- Secondly, the hidden components is calculated then apply Activation Function:

$h_1 = (x_1 * w_1) + (x_2 * w_2) + bias$

$\quad = (2 * 0) + (3 * 1) + 0 \Rightarrow h_1 = 3$

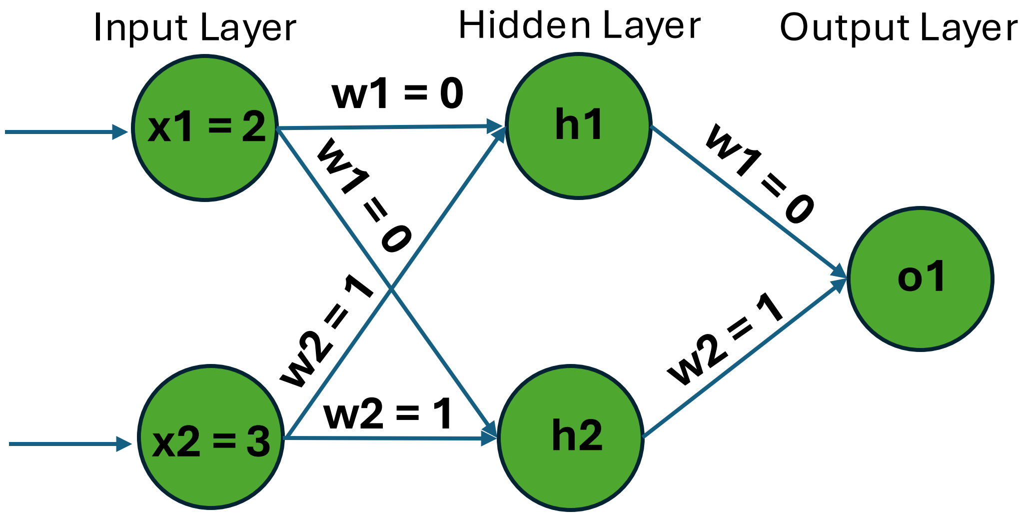
$h_2 = (x_1 * w_1) + (x_2 * w_2) + bias$

$\quad = (2 * 0) + (3 * 1) + 0 \Rightarrow h_2 = 3$

$\Rightarrow \; h_1 = h_2 = f(w * x + b)$

$\Rightarrow \quad = f((0 * 2) + (1 * 3) + 0) = f(3)$

$\Rightarrow \quad = 0.9526$ *(By Sigmoid Function)*



- $x = [2,3]$
- $w = [0,1]$
- $bias = 0$
- *Activation Function: Sigmoid*

Feedforward NN Example

- Thirdly and fourthly, the output is computed via:

  $o_1 = f((w_1 * h_1) + (w_2 * h_2) + b).$

  $= f((0 * 0.9526) + (1 * 0.9526) + 0)$

  $= f(0.9526)$

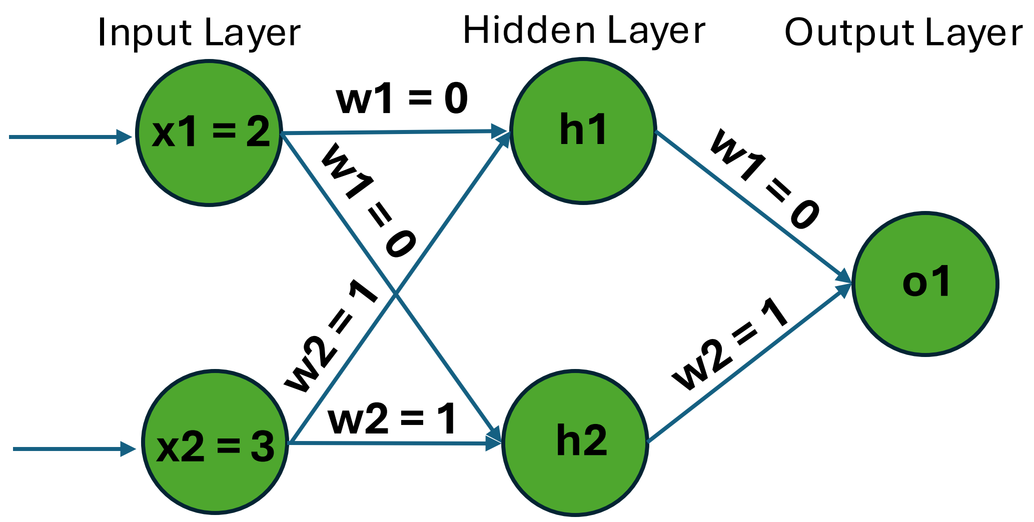
  $= 0.7216$

- Finally, the threshold is determined to decide the result.

  $Car > 0.5$

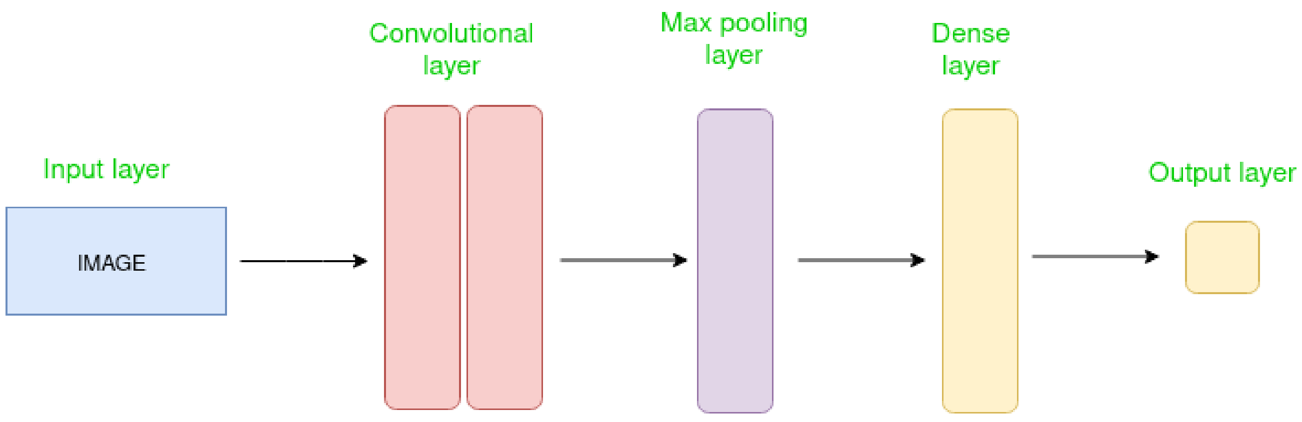
  $Bus \leq 0.5$

  The result (0.7216) is greater than 0.5 => Car



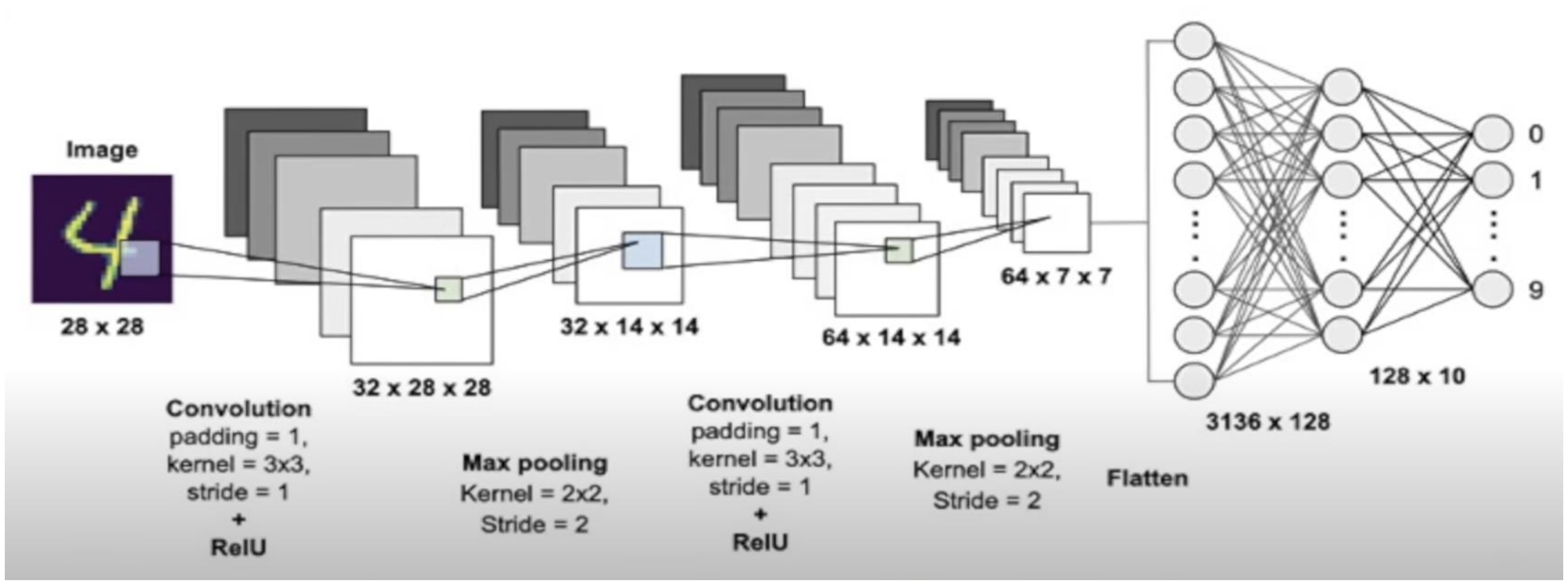
- $x = [2,3]$
- $w = [0,1]$
- $bias = 0$
- *Activation Function: Sigmoid*

Convolutional Neural Network (CNN):



- CNN is the extended version of artificial neural networks (ANN).
- It is a type of Deep Learning neural network architecture commonly used in Computer Vision (i.e. images, audio, and text).
- It is used to extract the feature from the grid-like matrix dataset (i.e. visual datasets like images or videos).
- The Convolutional layer applies filters to the input image to extract features.
- The Pooling layer down samples the image to reduce computation.
- The fully connected layer makes the final prediction.

CNN Example



- Passing the original images (i.e. 4k*4k pixels) to the fully connected NN leads to heavy expensive computations and consuming time.
- The important features of images is extracted by reducing the dimensions -> CNN.
- Kernel is known as filter is used to extract the features from the proposed image.
- It is a matrix moves over input data to perform the dot product with the sub-region of the input and get the output as the matrix of the dot products.

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 3*0 + 3*1 + 2*2
+ 0*2 + 0*2 + 1*0
+ 3*0 + 1*1 + 2*2

=

12		

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 3*0 + 2*1 + 1*2
+ 0*2 + 1*2 + 3*0
+ 1*0 + 2*1 + 2*2

=

12	12	

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 2*0 + 1*1 + 0*2
+ 1*2 + 3*2 + 1*0
+ 2*0 + 2*1 + 3*2

=

12	12	17

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 1*2
+ 3*2 + 1*2 + 2*0
+ 2*0 + 0*1 + 0*2

=

12	12	17
10		

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 1*1 + 3*2
+ 1*2 + 2*2 + 2*0
+ 0*0 + 0*1 + 2*2

=

12	12	17
10	17	

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 1*0 + 3*1 + 1*2
+ 2*2 + 2*2 + 3*0
+ 0*0 + 2*1 + 2*2

=

12	12	17
10	17	19

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 3*0 + 1*1 + 2*2
+ 2*2 + 0*2 + 0*0
+ 2*0 + 0*1 + 0*2

=

12	12	17
10	17	19
9		

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 1*0 + 2*1 + 2*2
+ 0*2 + 0*2 + 2*0
+ 0*0 + 0*1 + 0*2

=

12	12	17
10	17	19
9	6	

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 2*0 + 2*1 + 3*2
+ 0*2 + 2*2 + 2*0
+ 0*0 + 0*1 + 1*2

=

12	12	17
10	17	19
9	6	14

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow:

CNN Example (Kernel)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 2*0 + 2*1 + 3*2
+ 0*2 + 2*2 + 2*0
+ 0*0 + 0*1 + 1*2

=

Feature Map

12	12	17
10	17	19
9	6	14

3*3

- Suppose we have input image with 5*5 pixels and Kernel is determined by 3*3.
- The mathematical operations is conducted to get the **Feature Map**. How?
- We multiple the input (not 5*5 but 3*3 only) with the kernel as follow.
- The result matrix is known as the feature map.
- The following equation is used to calculated the size of the feature map.

*Feature Map Size = [(Input Size – Kernel Size) + 1] * [(Input Size – Kernel Size) + 1]*
= [(5-3) + 1] * [(5-3) + 1]
= [3] * [3]

CNN Example (Stride)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 3*0 + 3*1 + 2*2
+ 0*2 + 0*2 + 1*0
+ 3*0 + 1*1 + 2*2

=

Feature Map

12	12	17
10	17	19
9	6	14

3*3

- In the Kernel filter we had to move the window one step forward or downward.
- In the stride filter, we don't need to move step by step to catch the main feature of the input data.
- So, we move two steps!. How many windows we will get? = 4.

CNN Example (Stride)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

$$= 3*0 + 3*1 + 2*2 + 0*2 + 0*2 + 1*0 + 3*0 + 1*1 + 2*2$$

=

Feature Map

12	12	17
10	17	19
9	6	14

3*3

- In the Kernel filter we had to move the window one step forward or downward.
- In the stride filter, we don't need to move step by step to catch the main feature of the input data.
- So, we move two steps (2 strides)!. How many windows we will get? = 4.
- It means that, we are not considering the second column and row in the feature map.

CNN Example (Stride)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 3*0 + 3*1 + 2*2
+ 0*2 + 0*2 + 1*0
+ 3*0 + 1*1 + 2*2

=

Feature Map

12	12	17
10	17	19
9	6	14

3*3

- In the Kernel filter we had to move the window one step forward or downward.
- In the stride filter, we don't need to move step by step to catch the main feature of the input data.
- So, we move two steps (2 strides)!. How many windows we will get? = 4.
- It means that, we are not considering the second column and row in the feature map.
- The following equation is used to calculated the size of the feature map after applying the Stride filter.

$$\textit{Feature Map Size} = [\frac{(Input\ Size - Kernel\ Size)}{2} + 1] * [\frac{(Input\ Size - Kernel\ Size)}{2} + 1]$$

CNN Example (Stride)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

= 3*0 + 3*1 + 2*2 + 0*2 + 0*2 + 1*0 + 3*0 + 1*1 + 2*2 =

Feature Map

12	17
9	14

2*2

Keep repeating the process for the four window to get the feature map

- In the Kernel filter we had to move the window one step forward or downward.
- In the stride filter, we don't need to move step by step to catch the main feature of the input data.
- So, we move two steps (2 strides)!. How many windows we will get? = 4.
- It means that, we are not considering the second column and row in the feature map.
- The following equation is used to calculated the size of the feature map after applying the Stride filter.

$$\textit{Feature Map Size} = \left[\frac{(Input\ Size - Kernel\ Size)}{2} + 1\right] * \left[\frac{(Input\ Size - Kernel\ Size)}{2} + 1\right]$$

$$= \left[\frac{(5-3)}{2} + 1\right] * \left[\frac{(5-3)}{2} + 1\right] = 2 * 2$$

CNN Example (Padding)

3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

5*5
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

- Padding Filter is applied to fix the border effect problem of the input data.
- See the border of the input data (i.e. 3). How many times it involves in the Kernel filter?.
- Just 1 times, while the others may involve twice or more.
- Padding filter is used to deal with such challenge. How?

CNN Example (Padding)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

- Padding Filter is applied to fix the border effect problem of the input data.
- See the border of the input data (i.e. 3). How many times it involves in the Kernel filter?.
- Just 1 times, while the others may involve twice or more.
- Padding filter is used to deal with such challenge. How?

CNN Example (Padding)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 0*2
+ 0*2 + 3*2 + 3*0
+ 0*0 + 0*1 + 0*2

=

Feature Map

6				

5*5

- Padding Filter is applied to fix the border effect problem of the input data.
- See the border of the input data (i.e. 3). How many times it involves in the Kernel filter?.
- Just 1 times, while the others may involve twice or more.
- Padding filter is used to deal with such challenge. How?
- Then, we generate the Feature Map (5*5) as done before.

CNN Example (Padding)

Keep repeating the process for the four window to get the feature map

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

= 0*0 + 0*1 + 0*2 + 0*2 + 3*2 + 3*0 + 0*0 + 0*1 + 0*2 =

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

- Padding Filter is applied to fix the border effect problem of the input data.
- See the border of the input data (i.e. 3). How many times it involves in the Kernel filter?.
- Just 1 times, while the others may involve twice or more.
- Padding filter is used to deal with such challenge. How?
- Then, we generate the Feature Map (5*5) as done before.

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

= 0*0 + 0*1 + 0*2 + 0*2 + 3*2 + 3*0 + 0*0 + 0*1 + 0*2 =

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 0*2
+ 0*2 + 3*2 + 3*0
+ 0*0 + 0*1 + 0*2

=

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling
(2*2)

Max Pooling
(2*2)

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.
- *Pooling size is suggested to be (2*2).*

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 0*2
+ 0*2 + 3*2 + 3*0
+ 0*0 + 0*1 + 0*2

=

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling (2*2)

11.5	

Max Pooling (2*2)

14	

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.
- *Pooling size is suggested to be (2*2).*

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

$$= 0*0 + 0*1 + 0*2 + 0*2 + 3*2 + 3*0 + 0*0 + 0*1 + 0*2 =$$

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling (2*2)

11.5	14.25

Max Pooling (2*2)

14	17

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.
- *Pooling size is suggested to be (2*2).*

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

= = 0*0 + 0*1 + 0*2 + 0*2 + 3*2 + 3*0 + 0*0 + 0*1 + 0*2 =

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling (2*2)

11.5	14.25
9.5	

Max Pooling (2*2)

14	17
11	

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.
- *Pooling size is suggested to be (2*2).*

CNN Example (Pooling)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 0*2
+ 0*2 + 3*2 + 3*0
+ 0*0 + 0*1 + 0*2

=

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling
(2*2)

11.5	14.25
9.5	14.0

Max Pooling
(2*2)

14	17
11	19

- Padding Filter is used in order to down sampling feature map by summarizing the presence of features in patches of the feature map.
- There are two approached to do so; i) Average Pooling, ii) Max Pooling.
- *Pooling size is suggested to be (2*2).*

CNN Example (Flattening)

0	0	0	0	0	0	0
0	3	3	2	1	0	0
0	0	0	1	3	1	0
0	3	1	2	2	3	0
0	2	0	0	2	2	0
0	2	0	0	0	1	0
0	0	0	0	0	0	0

7*7
Input

0	1	2
2	2	0
0	1	2

3*3
Kernel

=

= 0*0 + 0*1 + 0*2
+ 0*2 + 3*2 + 3*0
+ 0*0 + 0*1 + 0*2

=

Feature Map

6	14	17	11	3
14	12	12	17	11
8	10	17	19	13
11	9	6	14	12
6	4	4	6	4

5*5

AVG Pooling (2*2)

11.5	14.25
9.5	14.0

11.5
14.25
9.5
14.0

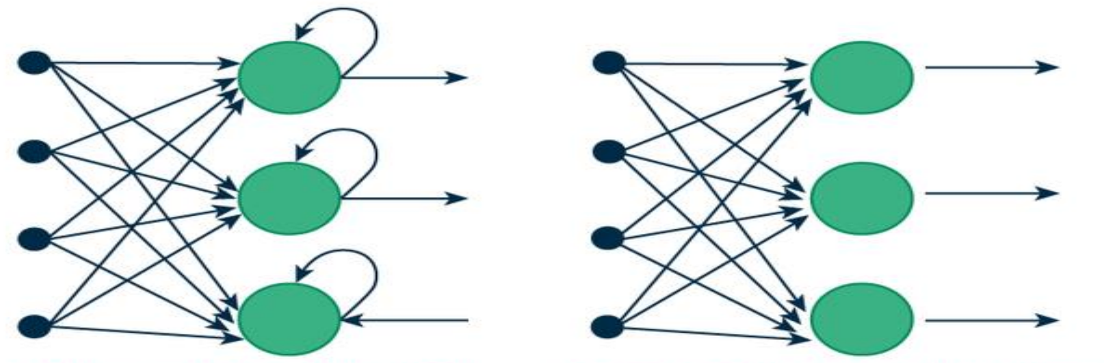
Max Pooling (2*2)

14	17
11	19

14
17
11
19

- After applying the **Pooled Feature Map,** the next step is to flatten it.
- In the Flattening approach, the entire pooled feature map matrix is transformed into single column.
- The single column is to be fed to the neural network (NN) for further processing.

Recurrent neural networks (RNNs):



- It is a type of Neural Network where the output from the previous step is fed as input to the current step.
- The fundamental processing unit in a Recurrent Neural Network (RNN) is a Recurrent Unit (Recurrent Neuron).
- The unit has the unique ability to maintain a hidden state, allowing the network to capture sequential dependencies by remembering previous inputs while processing.
- Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) versions improve the RNN's ability to handle long-term dependencies.
- Applications of Recurrent Neural Network including Language Modelling and Generating Text, Speech Recognition, Machine Translation, Image Recognition, Face detection, and Time series Forecasting.

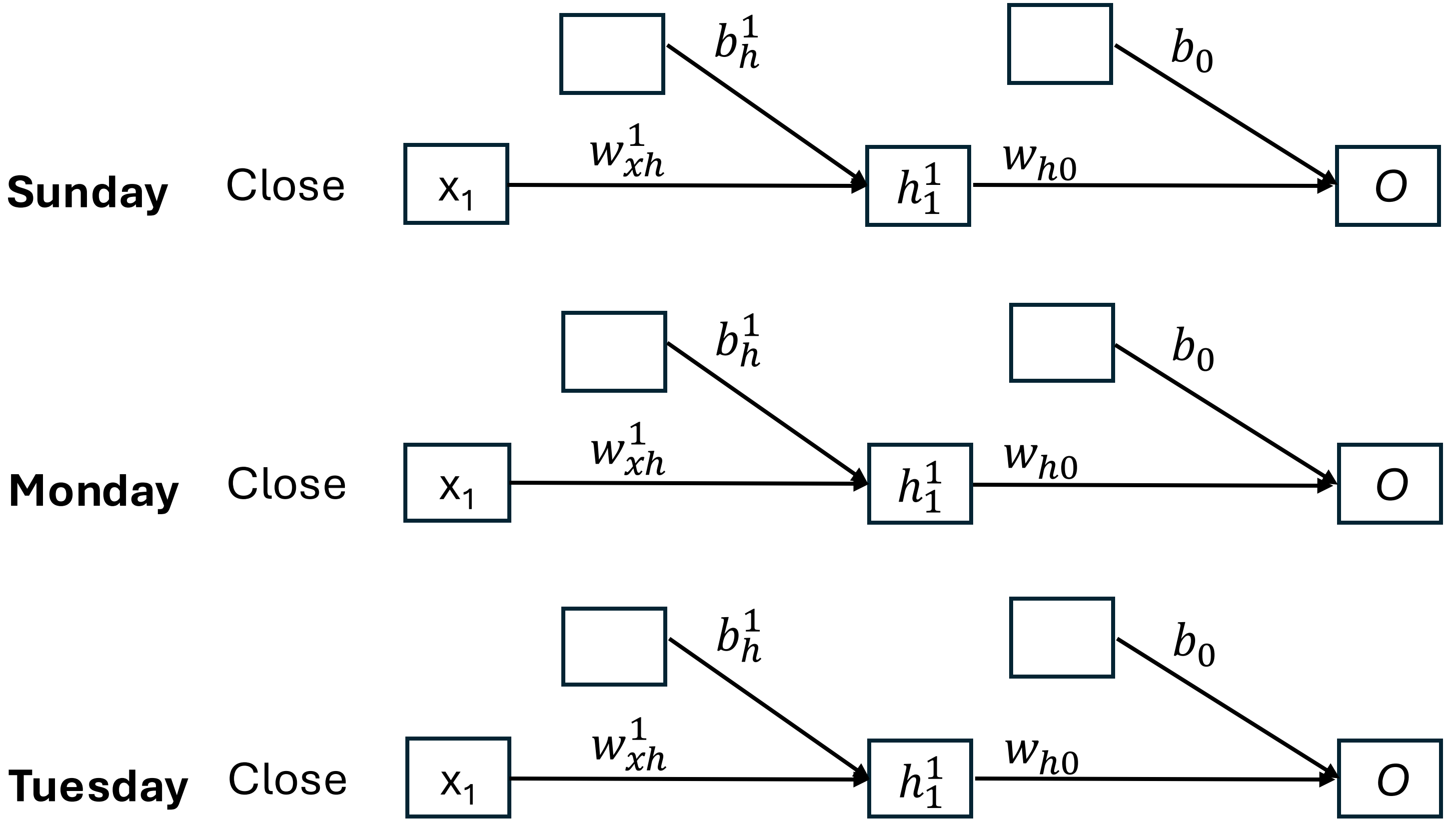
RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Unfolding

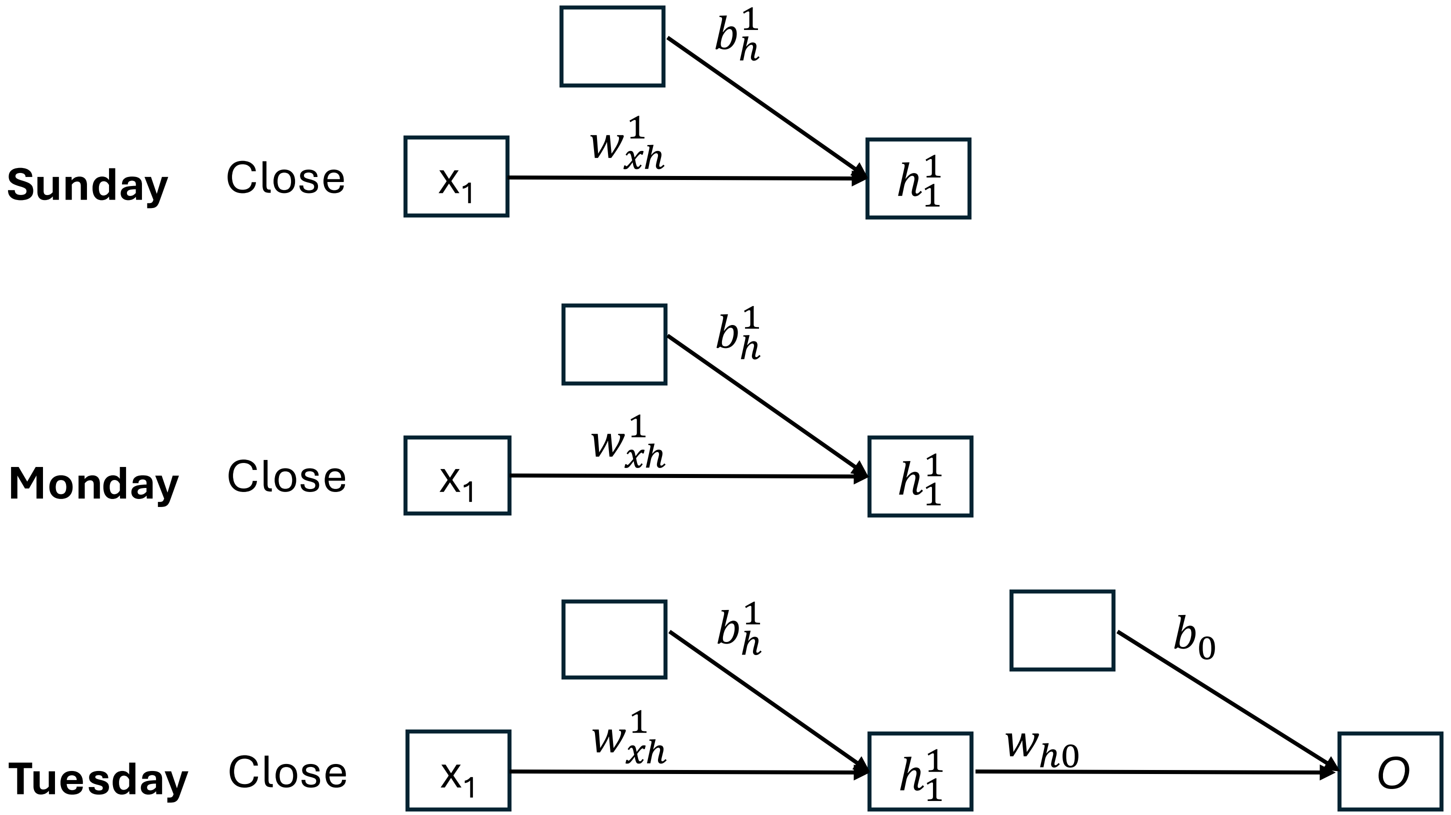
Or

Unrolling



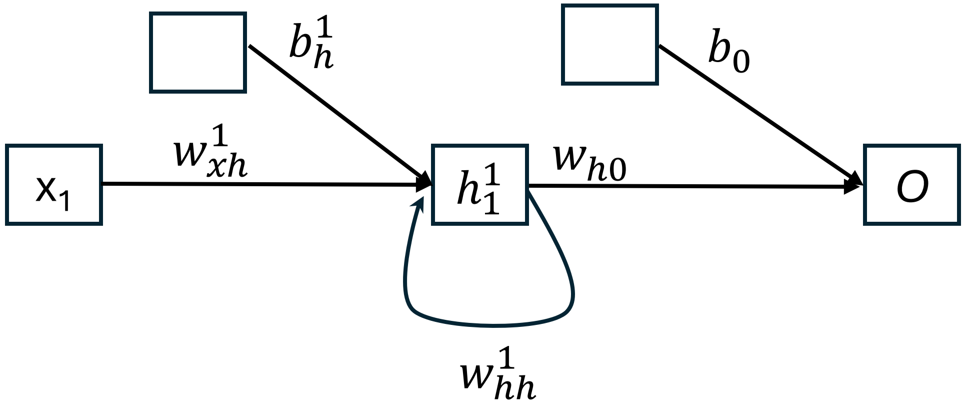
RNN Example

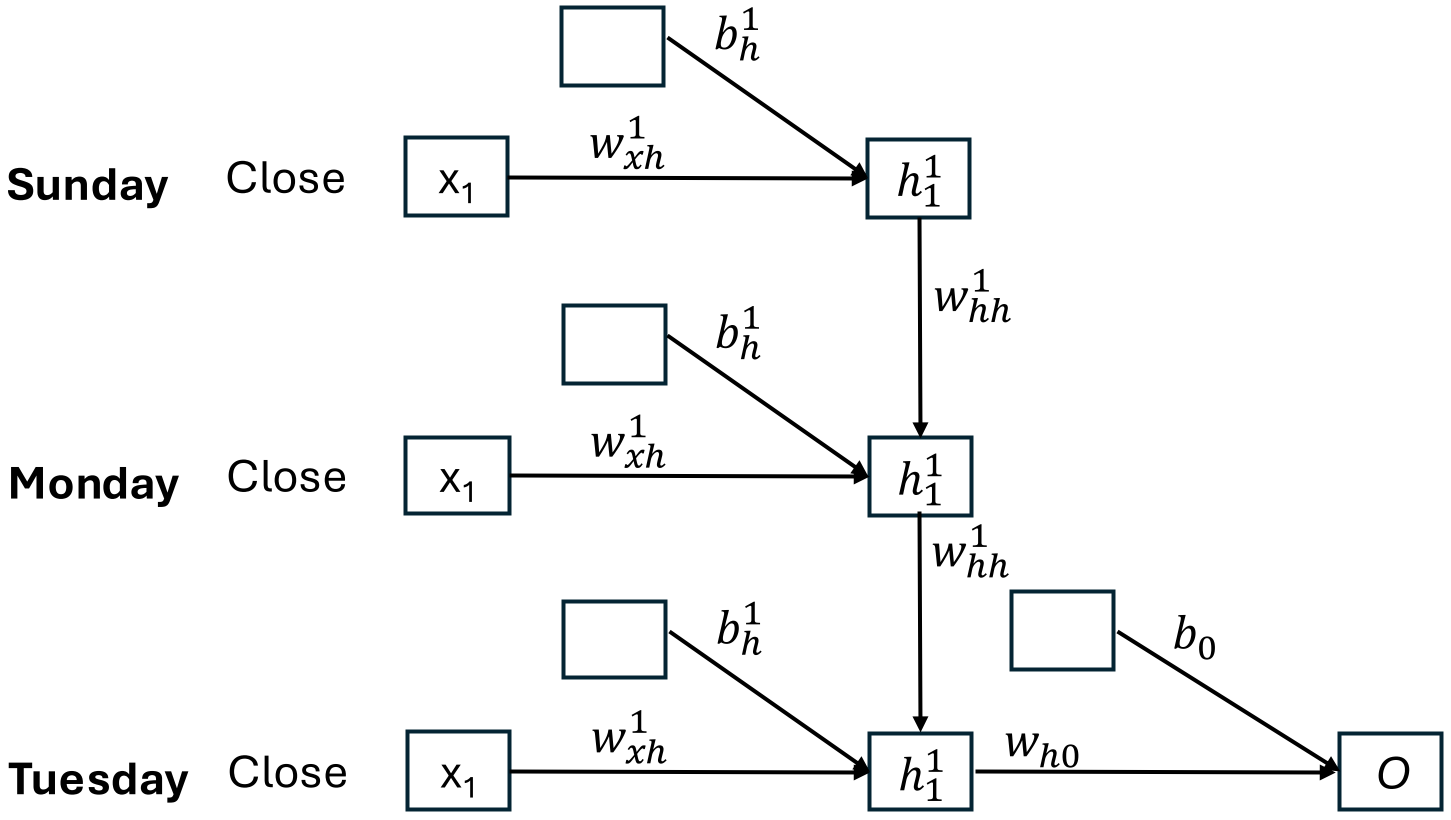
Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive



RNN Example

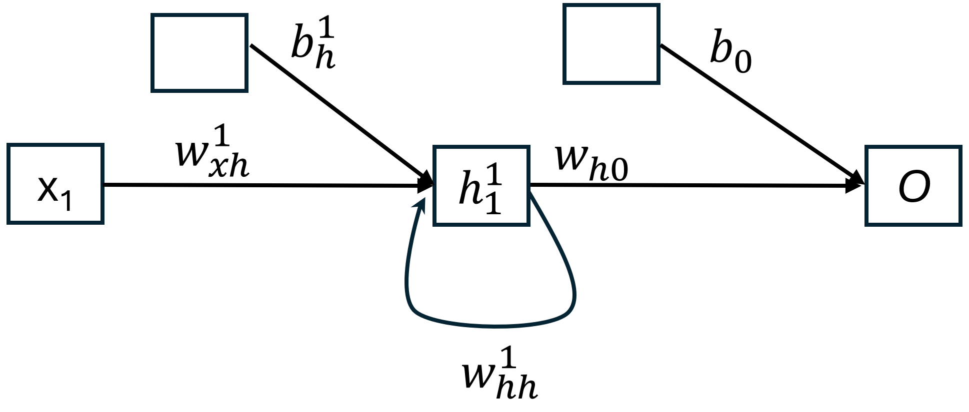
Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

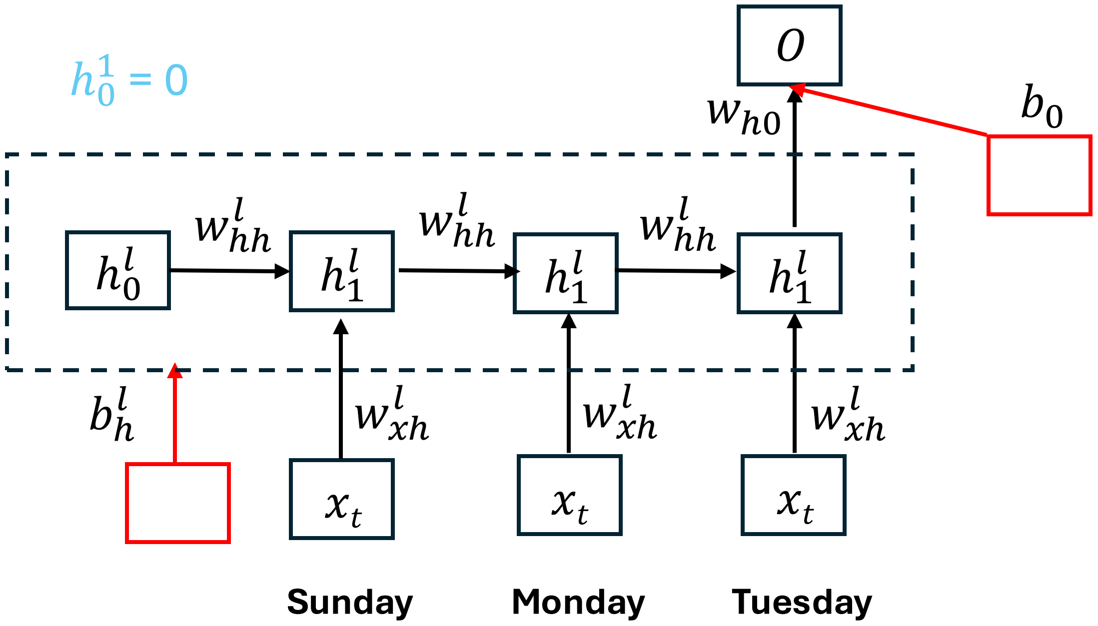




RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive





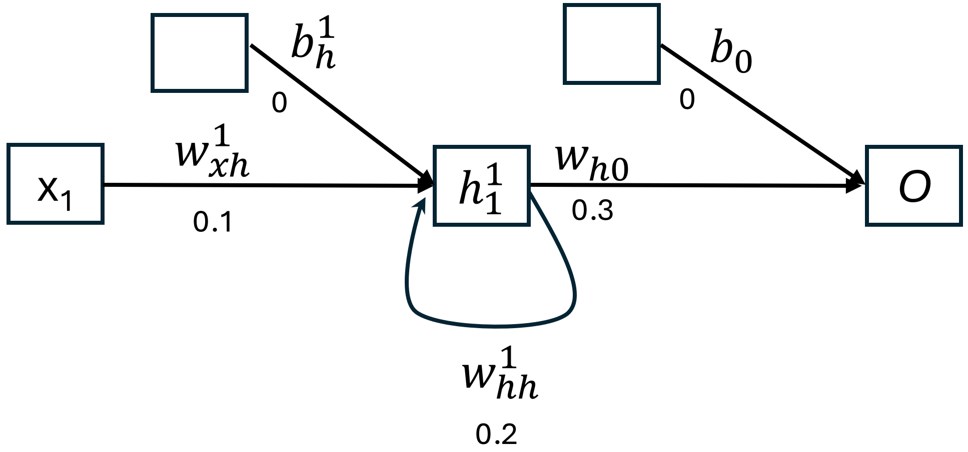
$$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$$

$$O = \emptyset(b_0 + h_t^1 * w_{h0})$$

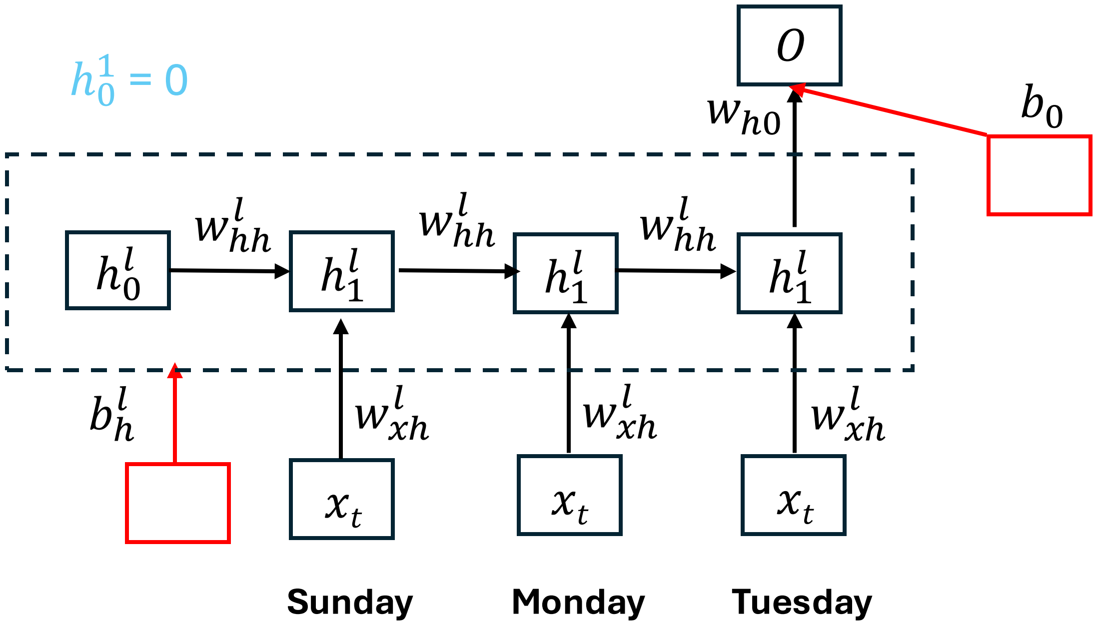
RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Assume



$h_0^1 = 0$



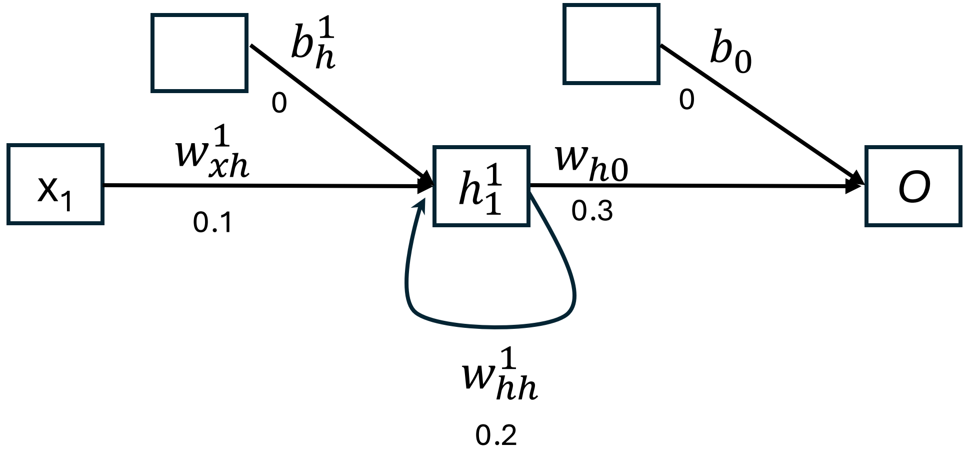
$$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$$

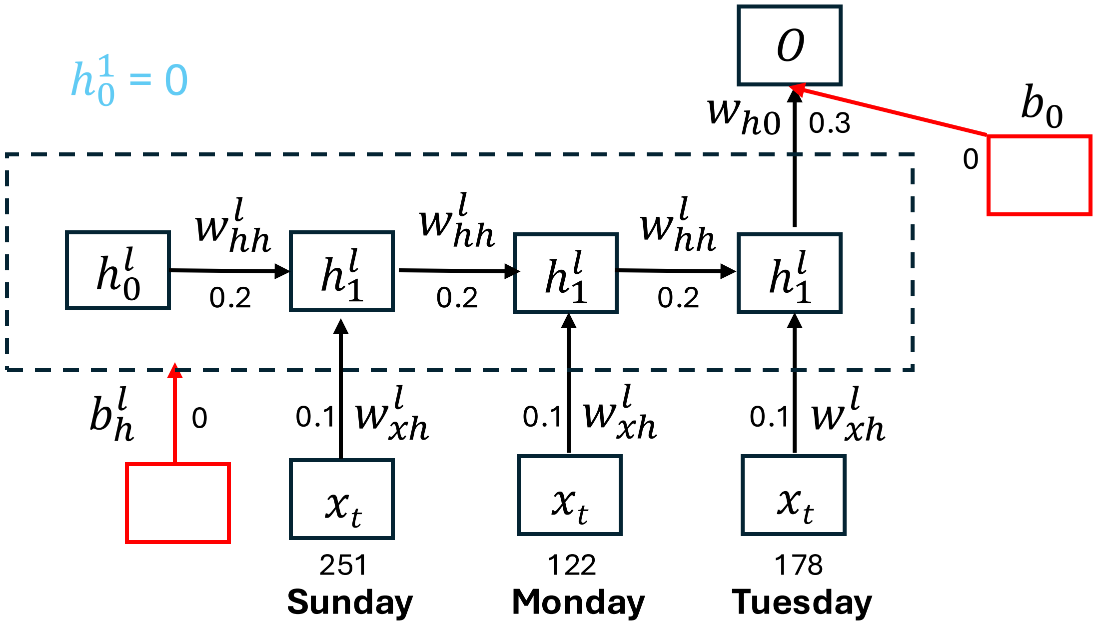
$$O = \emptyset(b_0 + h_t^1 * w_{h0})$$

RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Assume





$$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$$

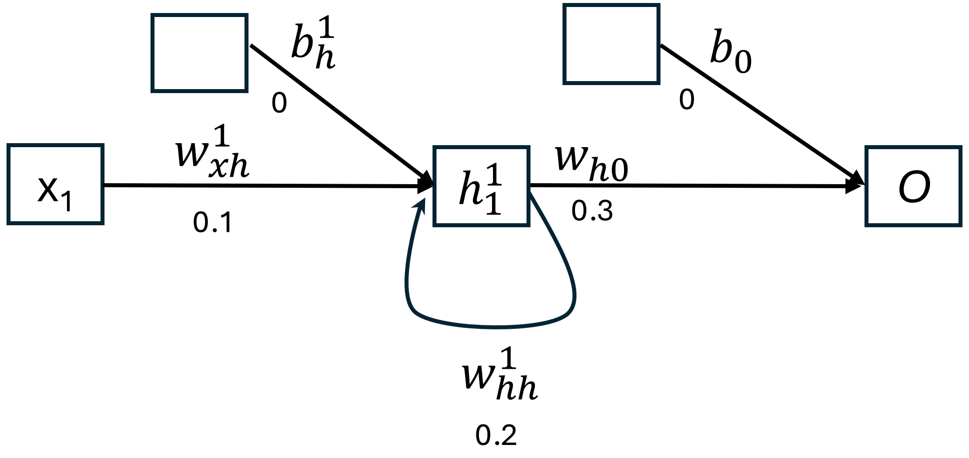
$$O = \emptyset(b_0 + h_t^1 * w_{h0})$$

Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

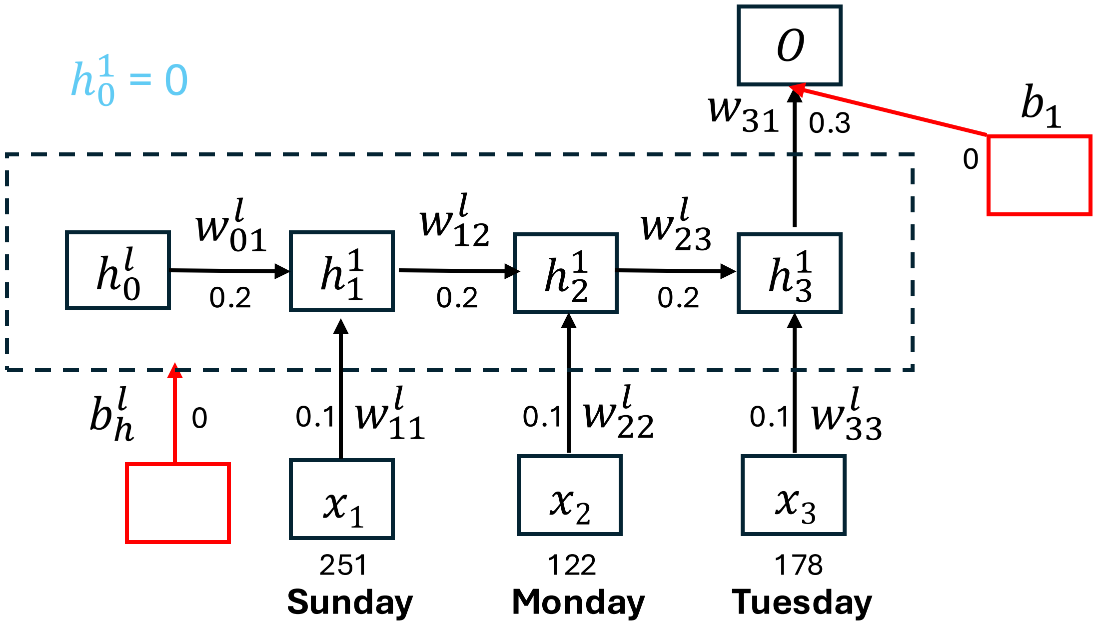
RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Assume



$h_0^1 = 0$



$$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$$

$$O = \emptyset(b_0 + h_t^1 * w_{h0})$$

RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Solve

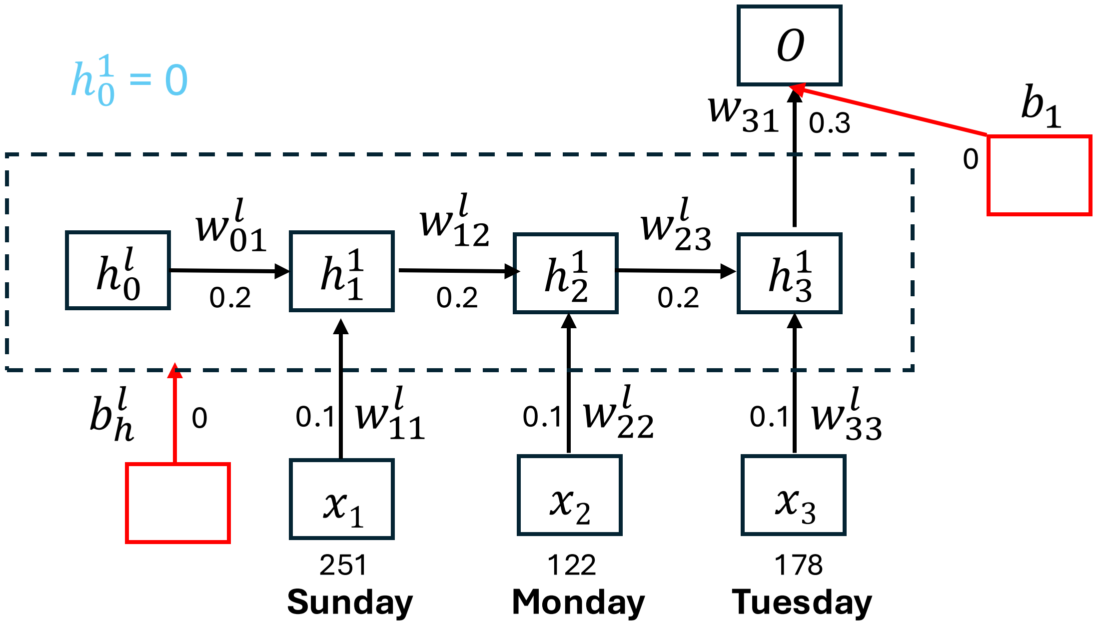
$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$ → $\emptyset = relu, where\ r(h) = \max(o, h)$

$h_1^1 = \emptyset(b_1^1 + x_1 * w_{11}^1 + h_{1-1}^1 * w_{01}^1)$

$h_1^1 = \emptyset(0 + 251 * 0.1 + 0 * 0.2)$

$h_1^1 = \emptyset(25.1)$

$h_1^1 = 25.1$



$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$

$O = \emptyset(b_0 \ + h_t^1 * w_{h0})$

RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Solve

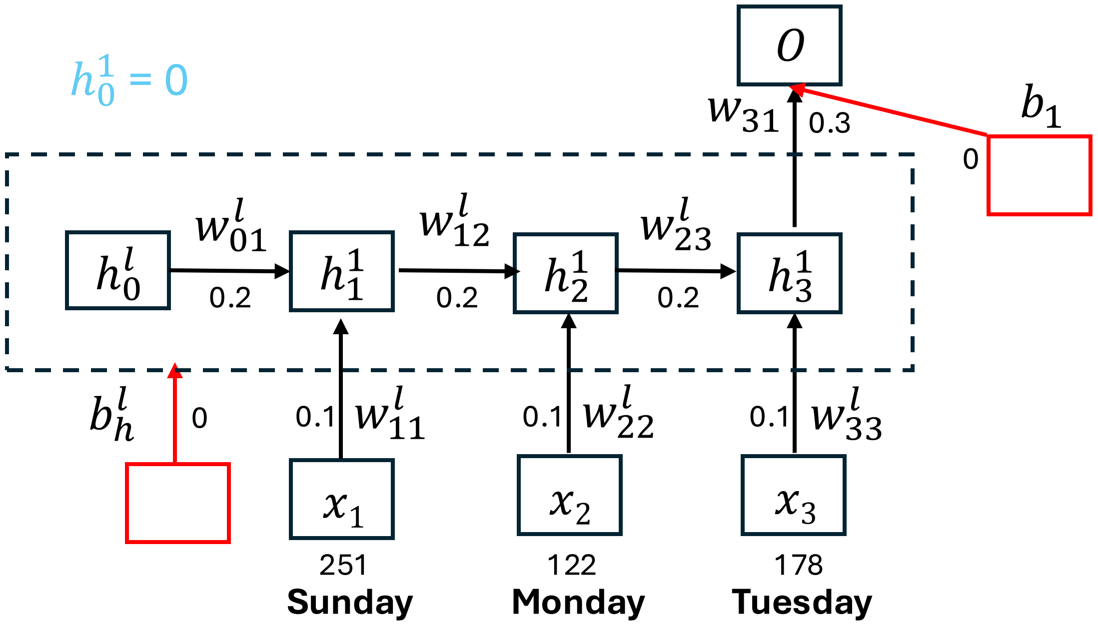
$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$ → $\emptyset = relu, where\ r(h) = \max(o, h)$

$h_2^1 = \emptyset(b_2^1 + x_2 * w_{22}^1 + h_{2-1}^1 * w_{12}^1)$

$h_2^1 = \emptyset(0 + 122 * 0.1 + 25.1 * 0.2)$

$h_2^1 = \emptyset(17.22)$

$h_2^1 = 17.22$



$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$

$O = \emptyset(b_0 + h_t^1 * w_{h0})$

RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Solve

$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$ → $\emptyset = relu, where\ r(h) = \max(o, h)$

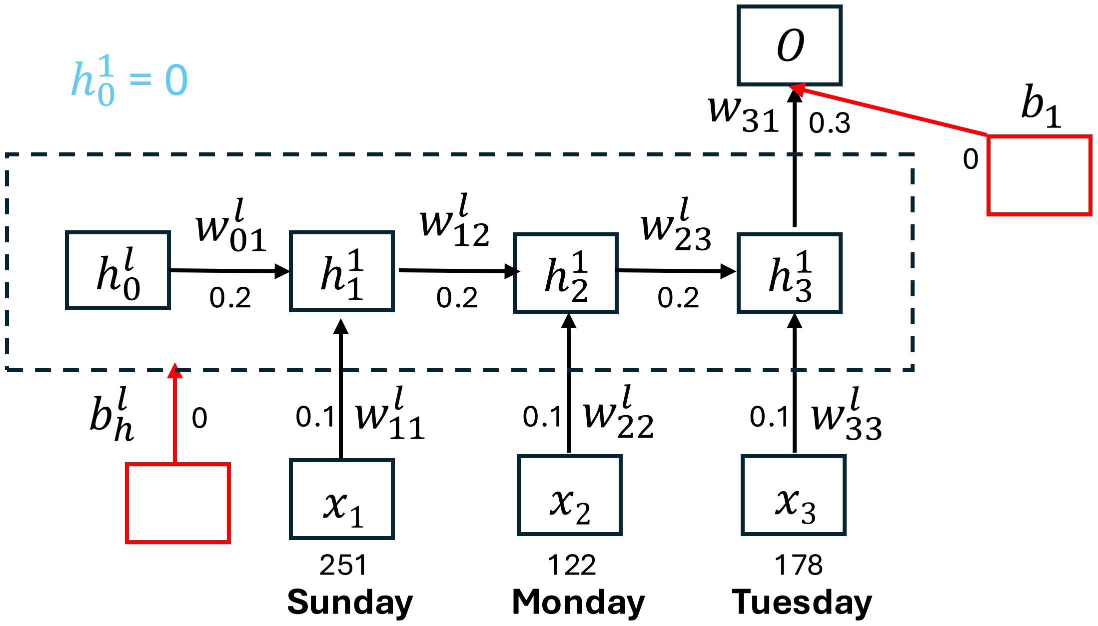
$h_3^1 = \emptyset(b_3^1 + x_3 * w_{33}^1 + h_{3-1}^1 * w_{23}^1)$

$h_3^1 = \emptyset(0 + 178 * 0.1 + 17.22 * 0.2)$

$h_3^1 = \emptyset(21.24)$

$h_3^1 = 21.24$

$h_0^1 = 0$



$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$

$O = \emptyset(b_0 \ + h_t^1 * w_{h0})$

RNN Example

Day	Close	Trend
Sunday	251	
Monday	122	
Tuesday	178	Positive

Solve

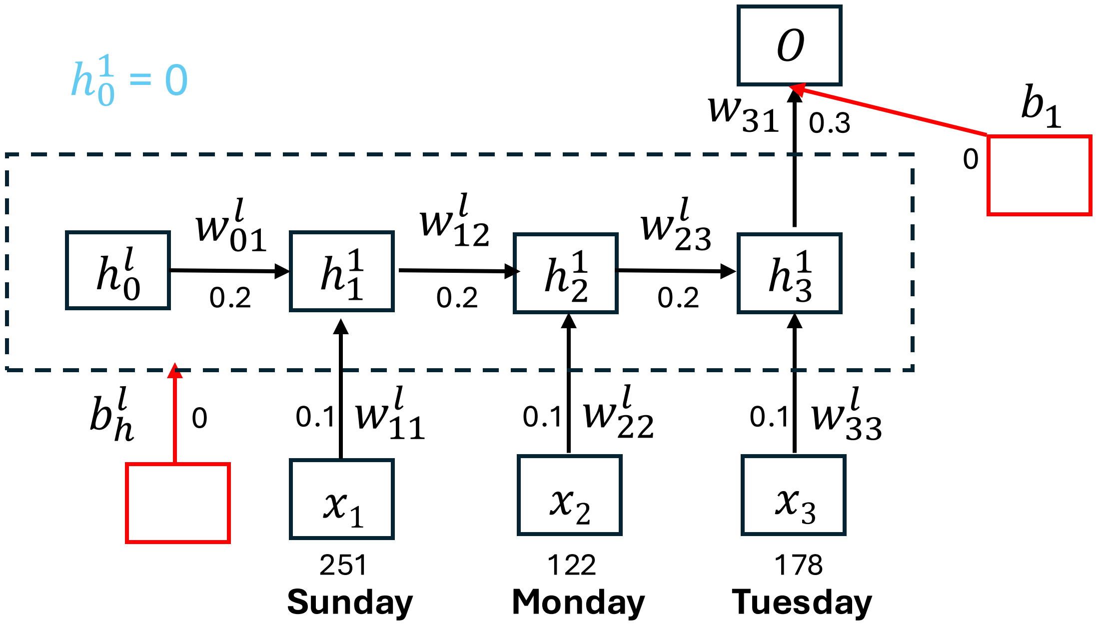
$O = \emptyset(b_0 + h_t^1 * w_{h0}) \rightarrow \emptyset = sigmoid, where\ r(O) = \frac{1}{1+e^{-O}}$

$O = \emptyset(b_1 + h_3^1 * w_{31})$

$O = \emptyset(0 + 21.24 * 0.3)$

$O = \emptyset(6.37)$

$O = 0.99$ *Result = Positive*

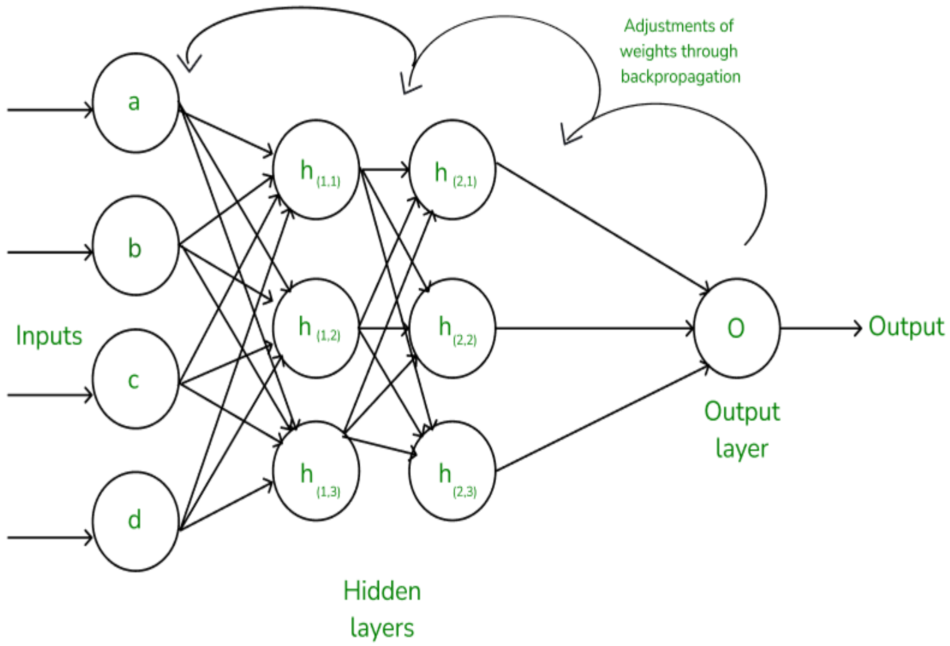


$h_t^1 = \emptyset(b_h^1 + x_t * w_{xh}^1 + h_{t-1}^1 * w_{hh}^1)$

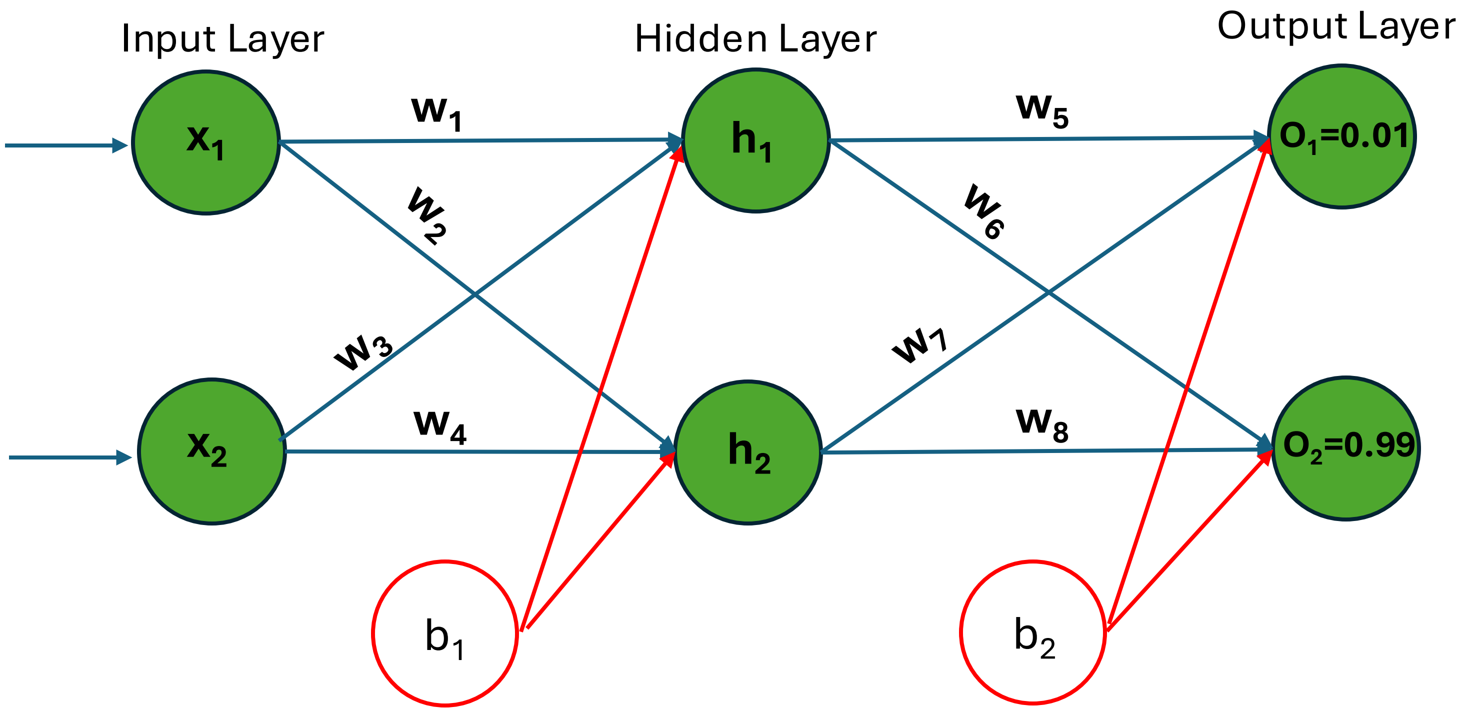
$O = \emptyset(b_0 + h_t^1 * w_{h0})$

Backpropagation in Neural Network

- Backpropagation is a common method for training a neural network.
- It is an iterative algorithm, that helps to minimize the cost function by determining which **weights** and **biases** should be adjusted.
- It involves two popular optimization algorithms, such as **gradient descent** or **stochastic gradient descent**.
- Computing the gradient in the backpropagation algorithm helps to minimize the cost function and it can be implemented by using the mathematical rule called **chain rule** from calculus to navigate through complex layers of the neural network.

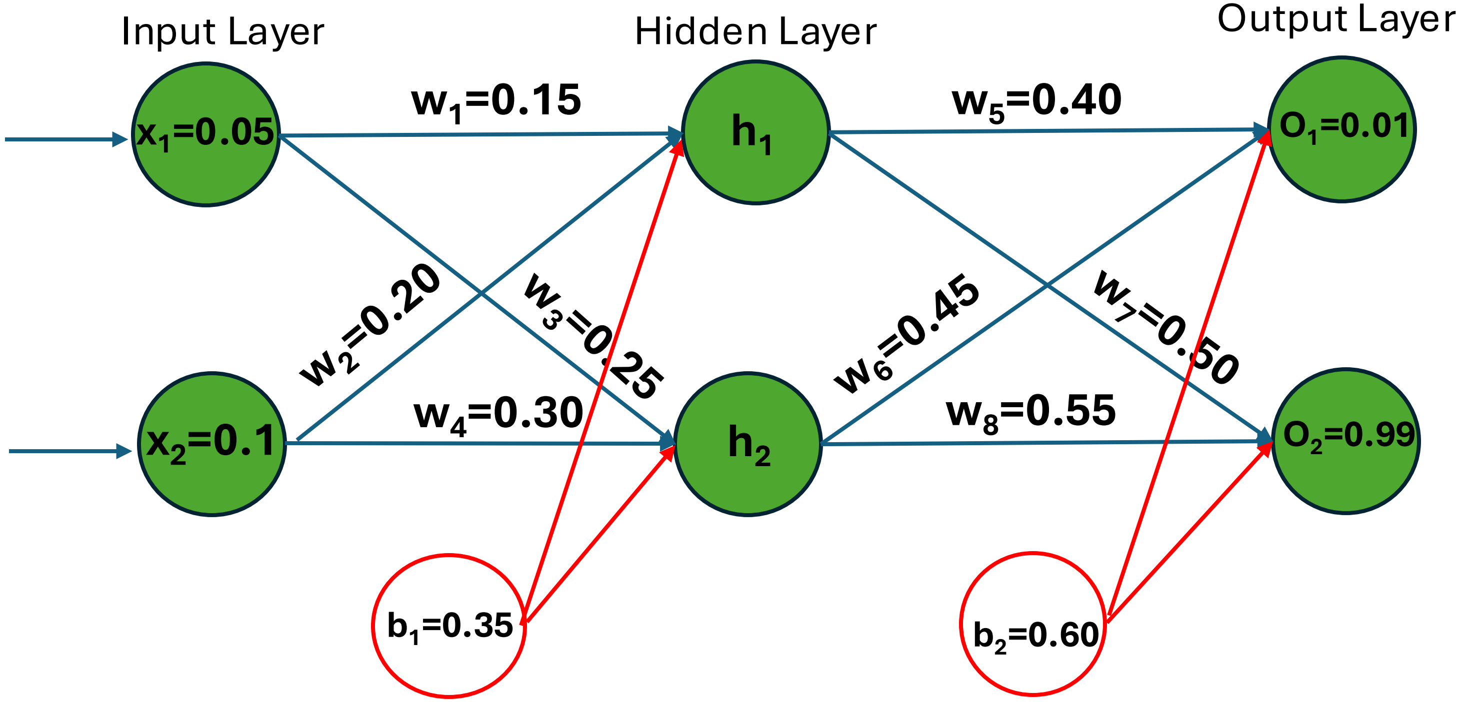


Example of Backpropagation



- **Assume** that the neurons have the sigmoid activation function to perform forward and backward pass on the network.
- **Assume** that the actual outputs of o_1 is 0.01 and o_2 is 0.99, where the learning rate is 1.
- The **Feedforward** training will be performed to compute the predicted outputs.
- Starting with the hidden neurons (h_1, and h_2)

Example of Backpropagation



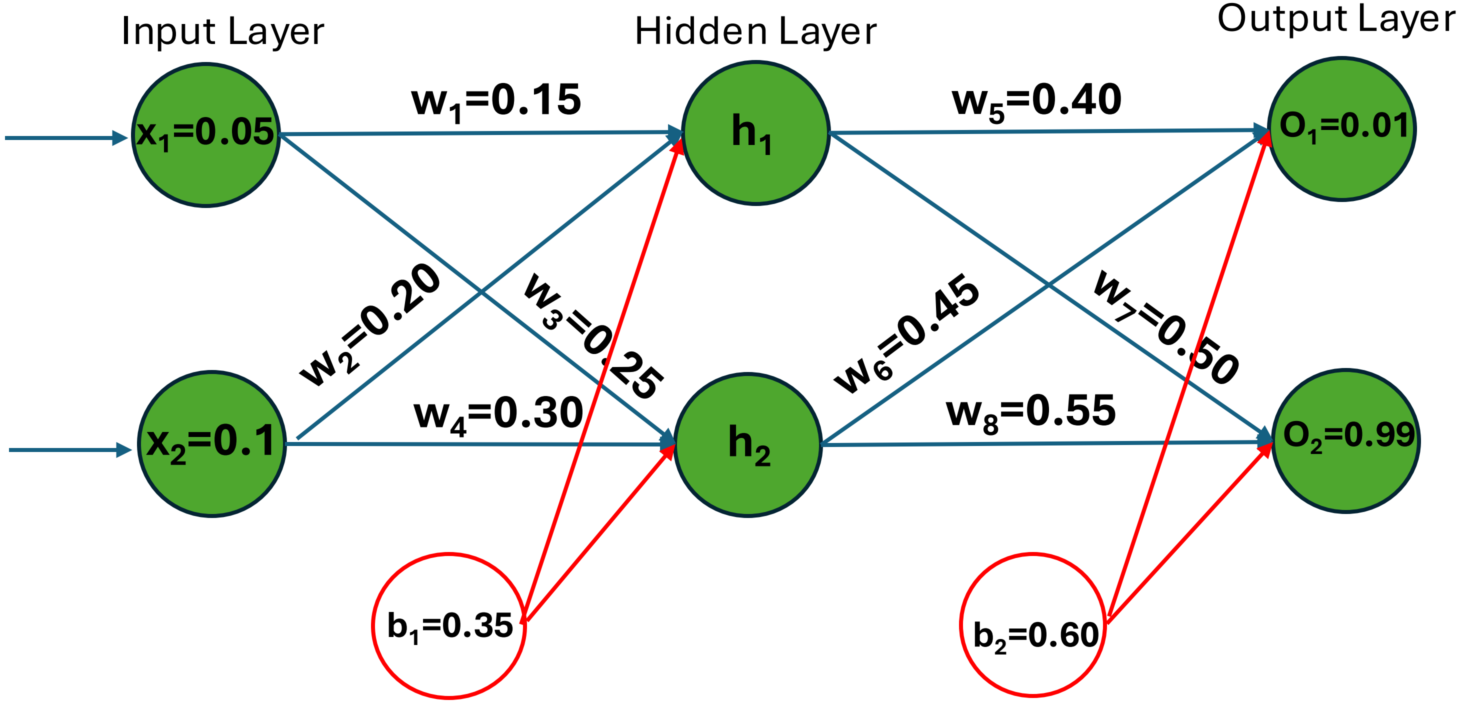
- $h_1 = (x_1 * w_1) + (x_2 * w_2) + b_1$

  $= (0.05*0.15) + (0.1*0.20) + 0.35$

  $= 0.3775$

$out_{h1} = Sigmoid(h_1) = 0.593269992$

Example of Backpropagation



- $h_1 = (x_1 * w_1) + (x_2 * w_2) + b_1$

  $= (0.05*0.15) + (0.1*0.20) + 0.35$

  $= 0.3775$

$out_{h1} = Sigmoid(h_1) = 0.593269992$

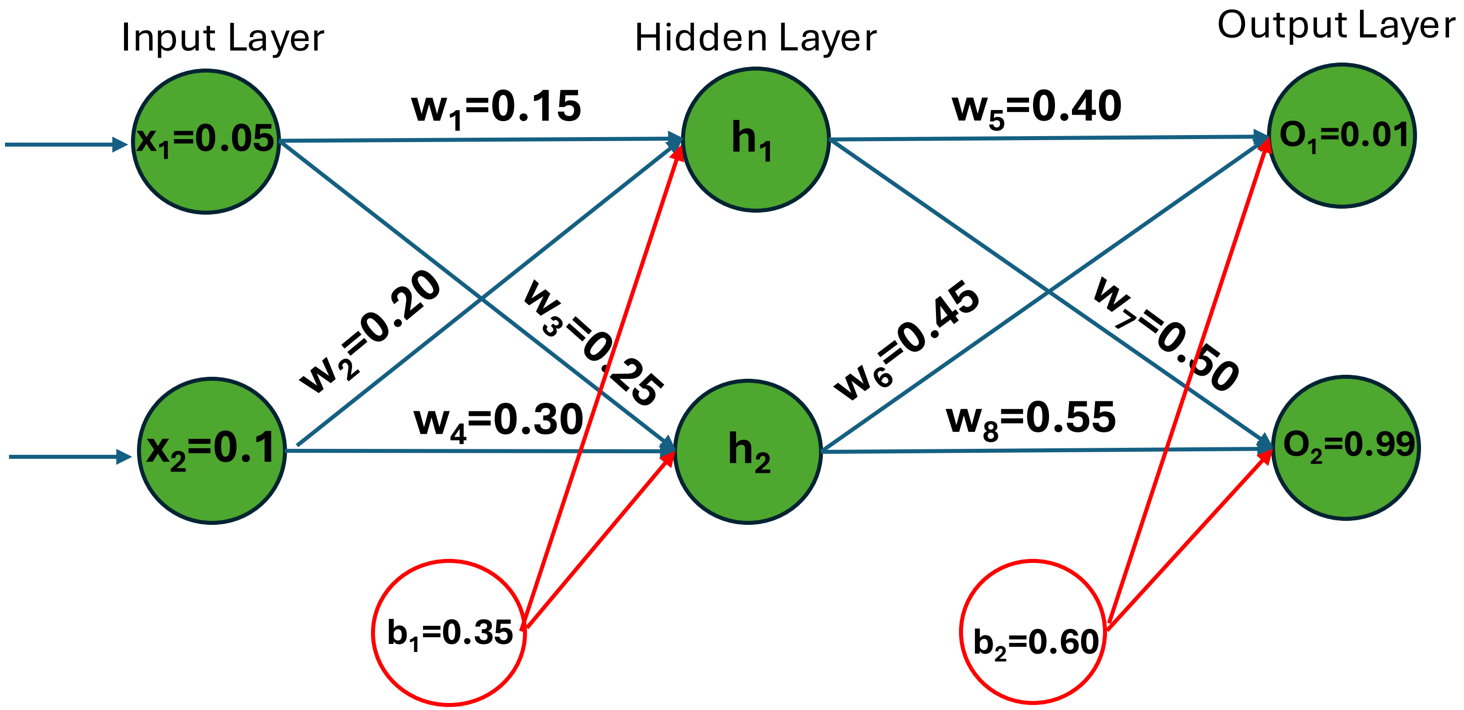
- $h_2 = (x_1 * w_3) + (x_2 * w_4) + b_1$

  $= (0.05*0.25) + (0.1*0.30) + 0.35$

  $= 0.3925$

$out_{h2} = Sigmoid(h_2) = 0.596884378$

Example of Backpropagation



- $h_1 = (x_1 * w_1) + (x_2 * w_2) + b_1$

  $= (0.05*0.15) + (0.1*0.20) + 0.35$

  $= 0.3775$

$out_{h1} = Sigmoid(h_1) = 0.593269992$

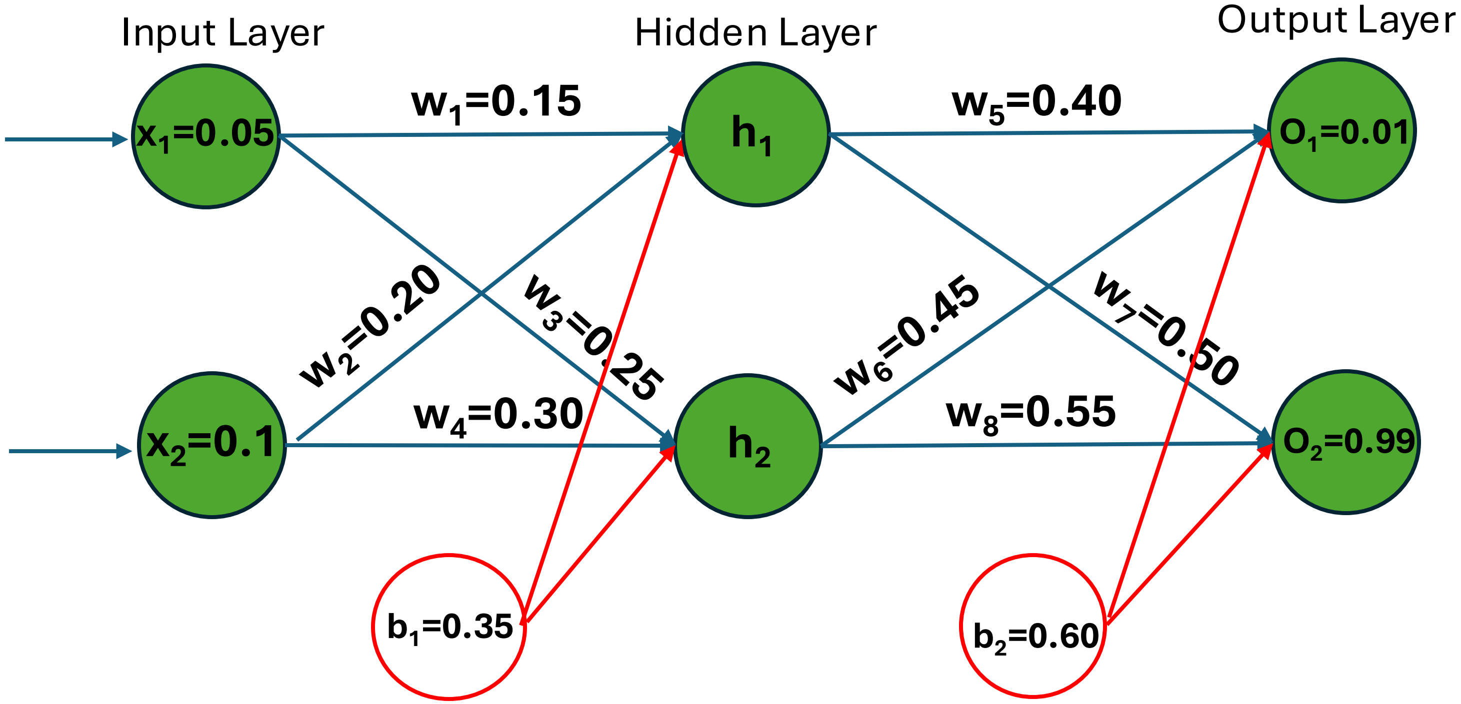
- $h_2 = (x_1 * w_3) + (x_2 * w_4) + b_1$

  $= (0.05*0.25) + (0.1*0.30) + 0.35$

  $= 0.3925$

$out_{h2} = Sigmoid(h_2) = 0.596884378$

Example of Backpropagation



Now, Computing the output neurons (o_1, and o_2)

$$o_1 = (out_{h1} * w_5) + (out_{h2} * w_6) + b_2$$

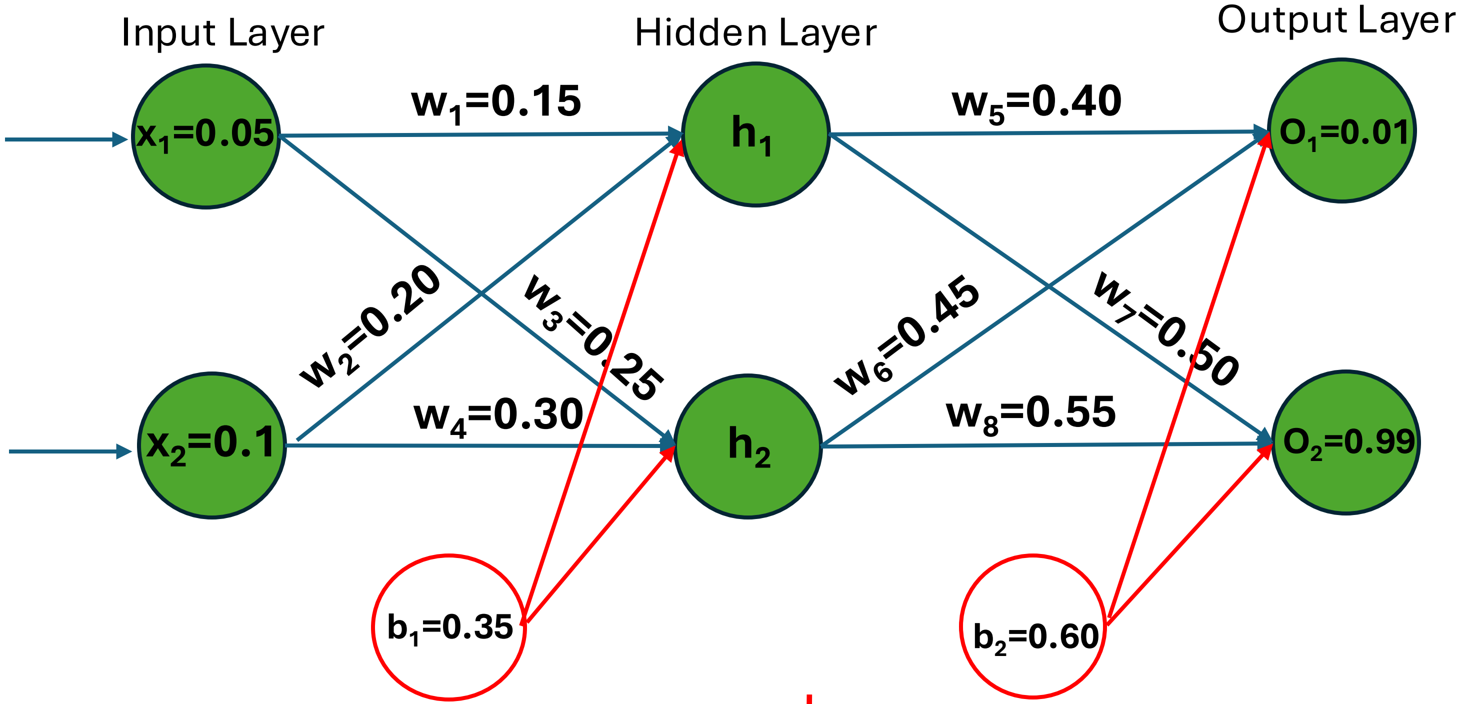
$$=(0.593269992*0.40)+(0.596884378*0.45)$$

$$+ 0.60$$

$$= 1.105905967$$

$$out_{o1} = Sigmoid(o_1) = 0.75136507$$

Example of Backpropagation

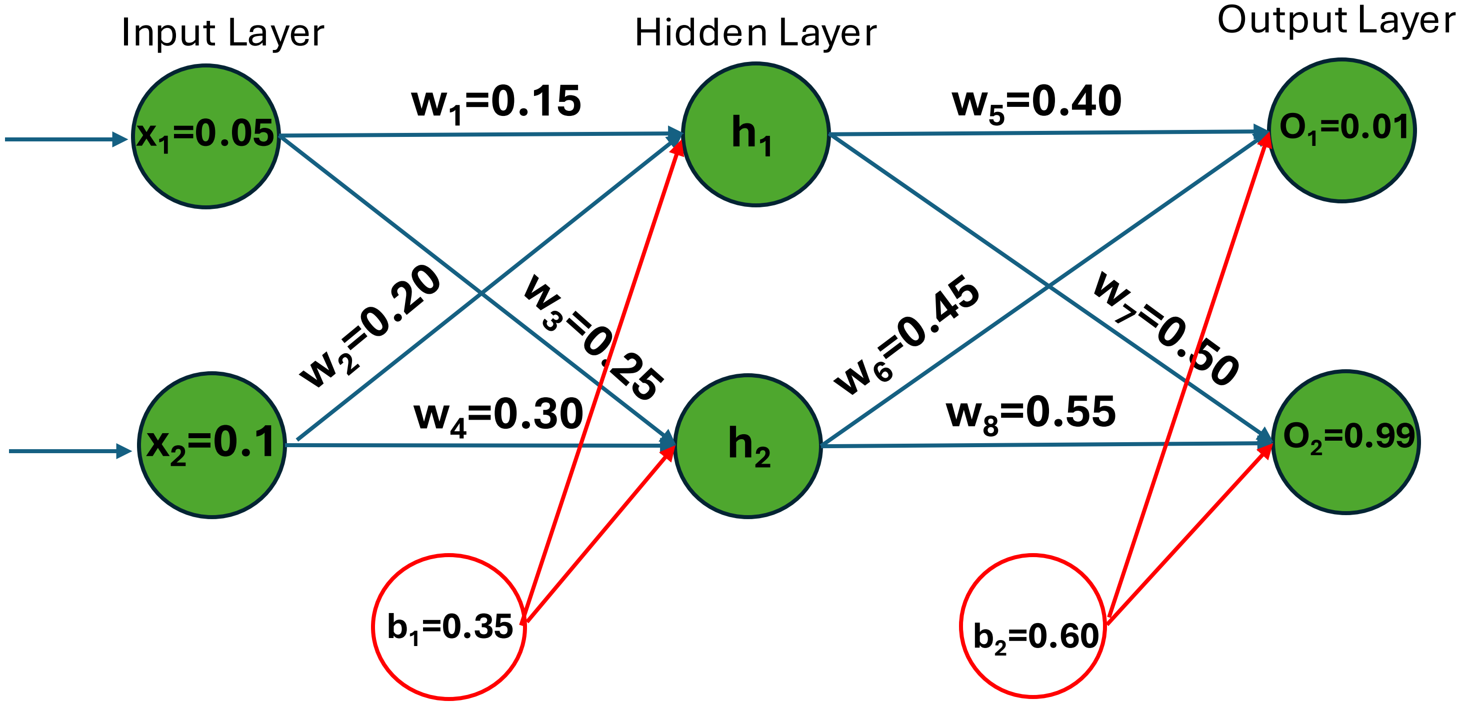


Now, Computing the output neurons (o_1, and o_2)

$$o_1 = (out_{h1} * w_5) + (out_{h2} * w_6) + b_2$$
$$= (0.593269992*0.40) + (0.596884378*0.45) + 0.60$$
$$= 1.105905967$$
$$out_{o1} = Sigmoid(o_1) = 0.75136507$$

$$o_2 = (out_{h1} * w_7) + (out_{h2} * w_8) + b_2$$
$$= (0.593269992*0.50) + (0.596884378*0.55) + 0.60$$
$$= 1.2249214039$$
$$out_{o2} = Sigmoid(o_2) = 0.772928465$$

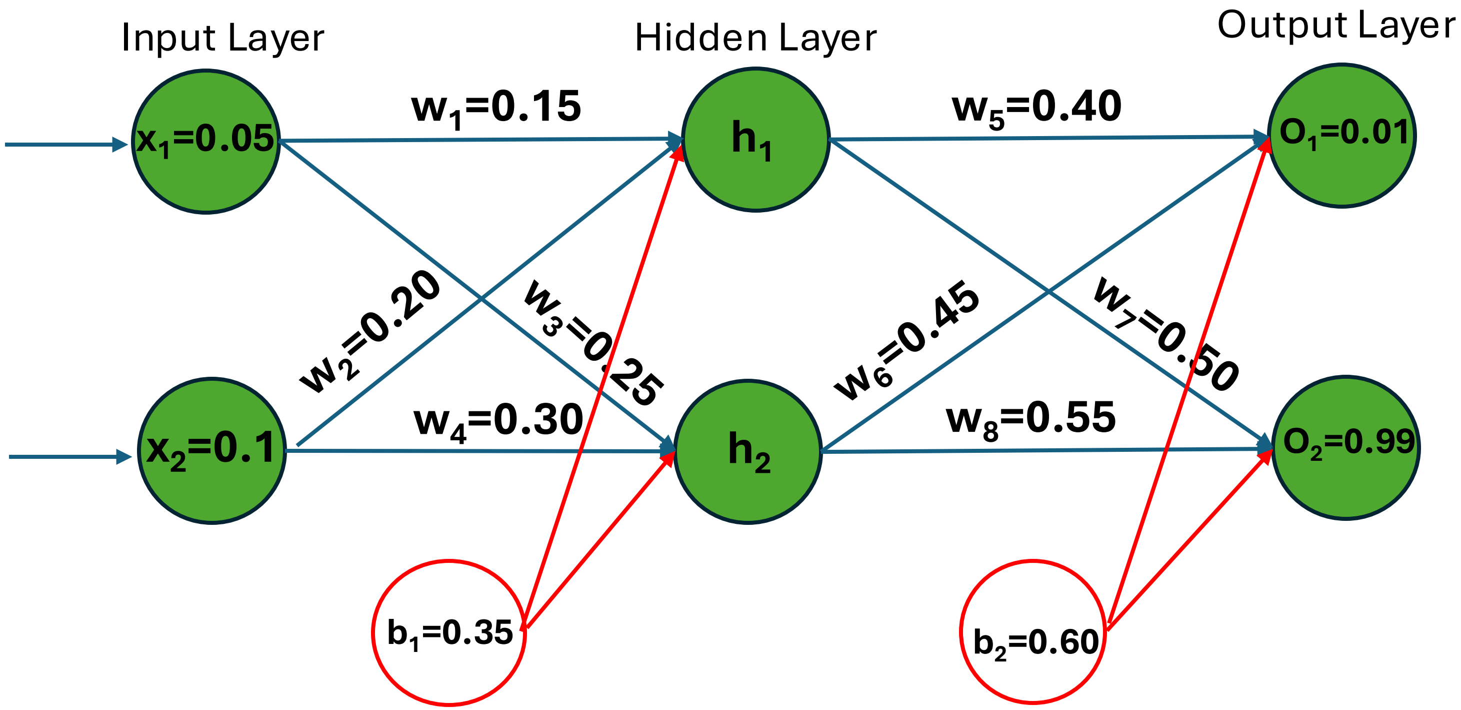
Example of Backpropagation



- The differences (***Error***) between the predicted (output) and actual (target) are computed by using this formula (Loss Function):

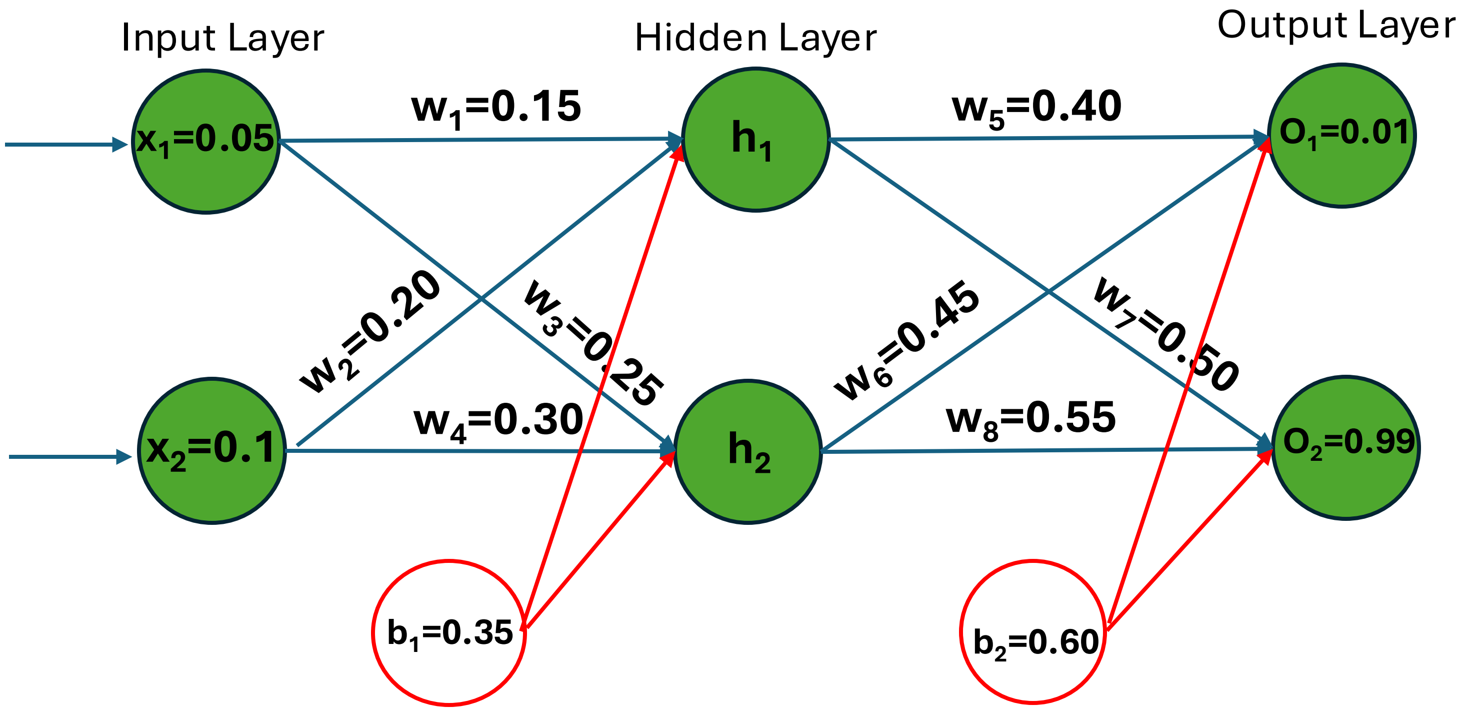
$$E_{total} = \sum \frac{1}{2}(target - output)^2$$

Example of Backpropagation



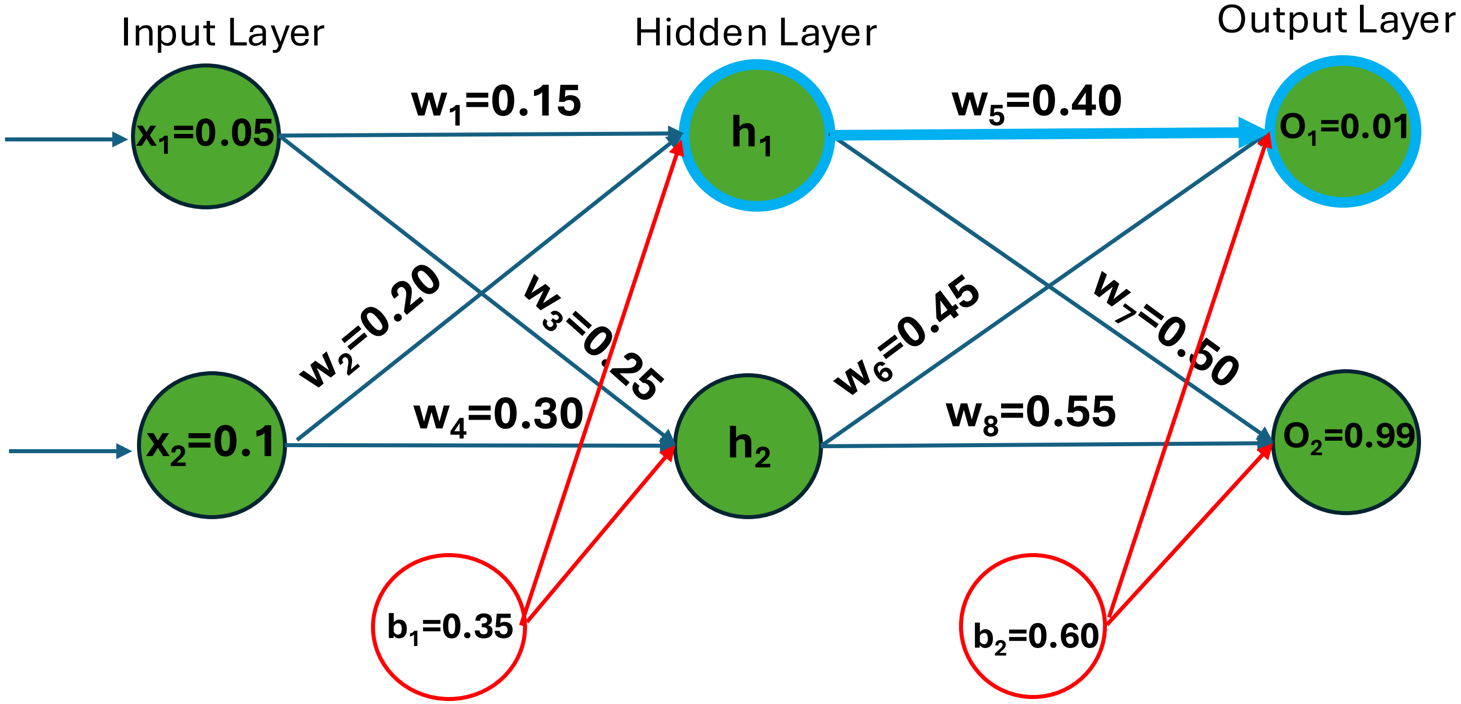
Target (Actual)	Output (Predicted)	Error ($\sum \frac{1}{2}(target - output)^2$)
0.01	0.75136507	0.274811083
0.99	0.772928465	0.023560026
$E_{total} = E_{o1} + E_{o2}$		0.298371109

Example of Backpropagation



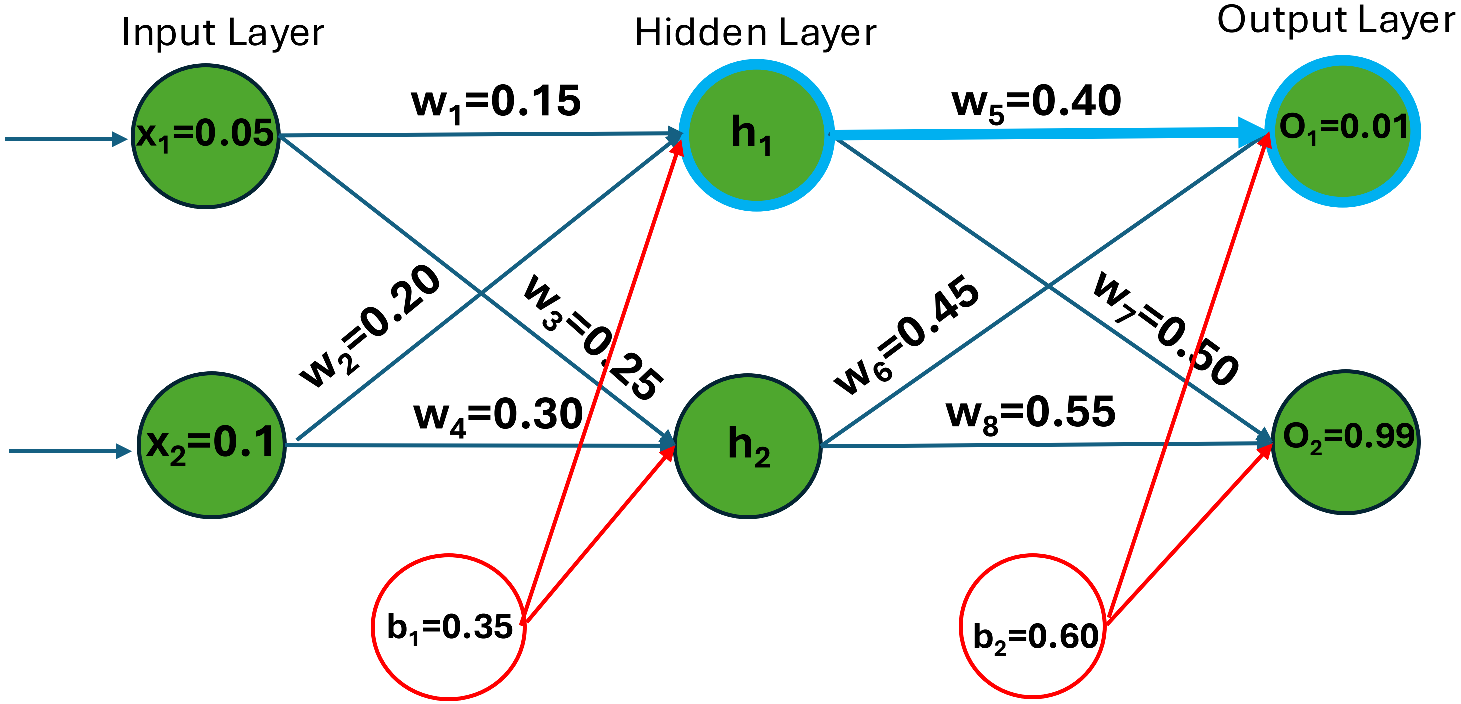
- **The Backpropagation Algorithm:**
  - The goal is to update each of the weights in the network so that they cause the actual output to be closer the target output, thereby minimizing the error for each output neuron and the network (<u>Backpropagation</u>).
  - The partial derivative function by applying the Chain Rule formula, is applied starting from output layer to hidden layer.

Example of Backpropagation



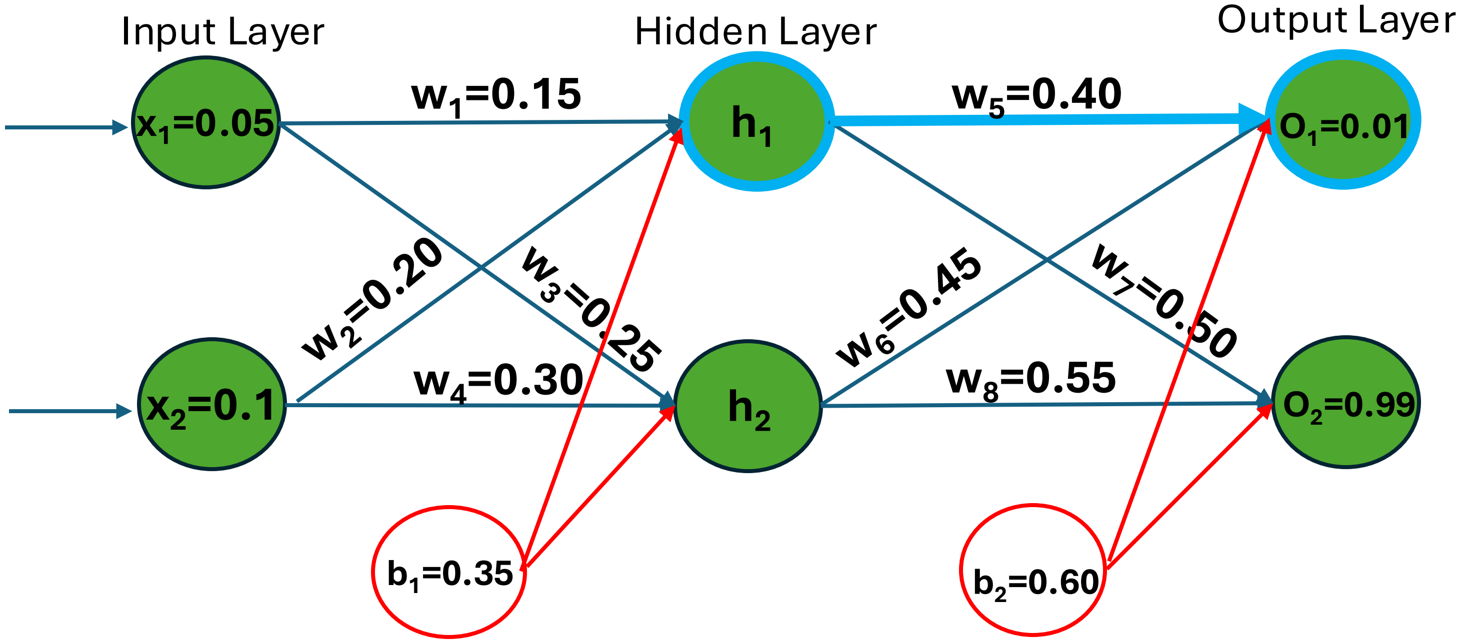
- **The Backpropagation Algorithm (Output Layer):**
  - First, we need to know how much a change in w_5 affects the total error.
  - It is stated as $\frac{\partial E_{total}}{\partial w_5}$, and read as the partial derivative of E_{total} with respect to w_5.
  - As, there is no direct connection from the equation between the E_{total} and w_5, we use the chain rule.

Example of Backpropagation



- **The Backpropagation Algorithm (Output Layer):**
  - However, there is direct connection (from equations) between E_{total} and out_{o1}, and between out_1 and h_1, and then h_1 and w_5.
  - Therefore, we use the following formula to compute the $\frac{\partial E_{total}}{\partial w_5}$, as follow:

Example of Backpropagation



$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial Etotal}{\partial out_{o1}} * \frac{\partial outo_1}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial w_5}$$, where;

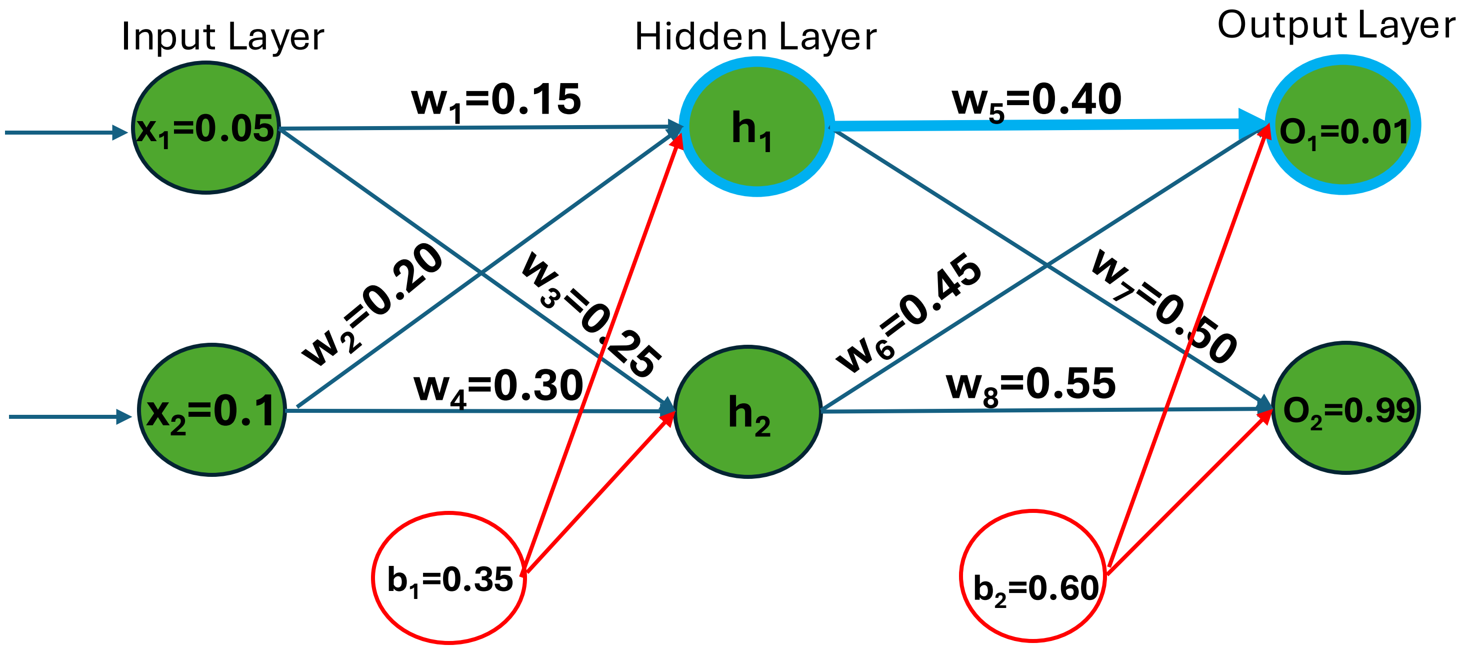
1) $\frac{\partial_{Etotal}}{\partial out_{o1}} = 2 * \frac{1}{2}\ (target_{o1} - outo_1)^{2\ -\ 1} * -1 + 0$

$= out_{o1} - \ target_{o1}$

2) $\frac{\partial outo_1}{\partial out_{h1}} = out_{o1} * (1 - out_{o1})$

3) $\frac{\partial h_1}{\partial w_5} = 1 * h_1 * w_5^{(1-1)}$

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_5} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial outh_1} * \frac{\partial outh_1}{\partial w_5}$$, where;

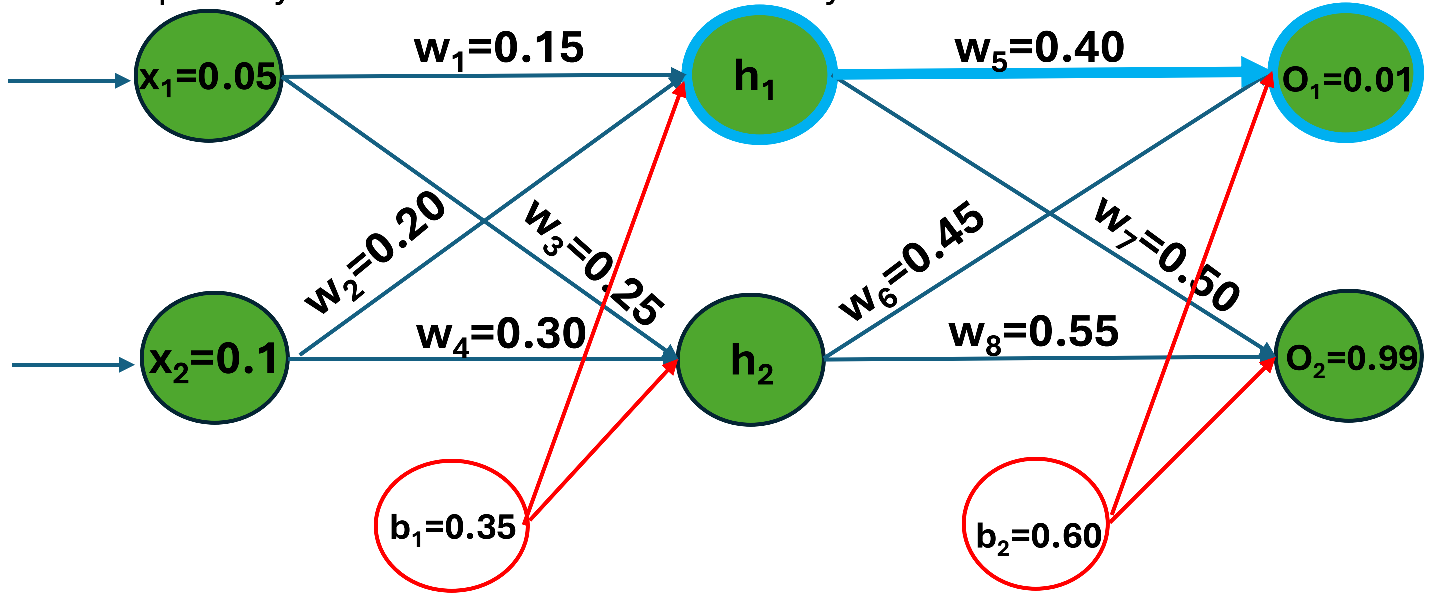
1) $\frac{\partial_{Etotal}}{\partial out_{o1}} = 2 * \frac{1}{2}\ (target_{o1} - outo_1)^{2\ -\ 1} * -1 + 0$

$= out_{o1}\ -\ target_{o1}$

$= 0.75136507 - 0.01$

$= 0.74136507$

Example of Backpropagation



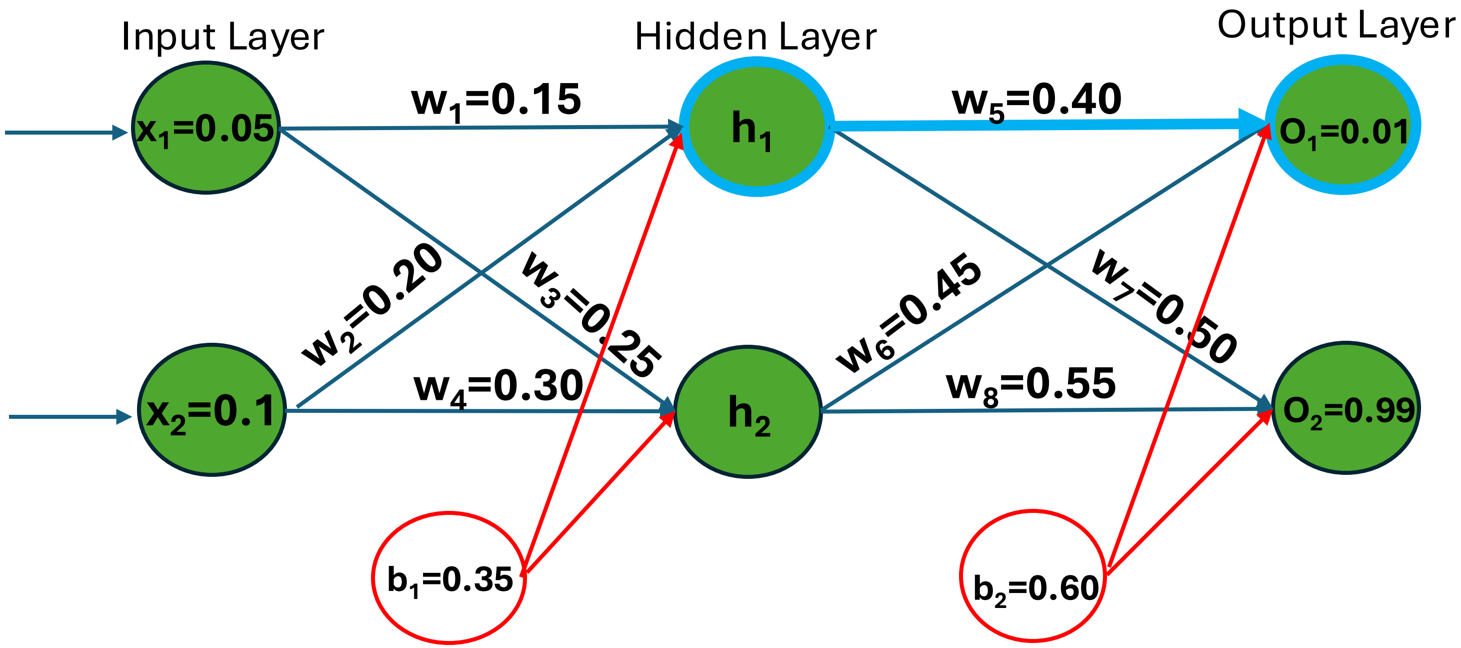
$$\frac{\partial Etotal}{\partial w_5} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial w_5}$$, where;

2) $\frac{\partial outo_1}{\partial out_{h1}} = out_{o1} * (1 - out_{o1})$

$= 0.75136507 * (1 - 0.75136507)$

$= 0.186815602$

Example of Backpropagation



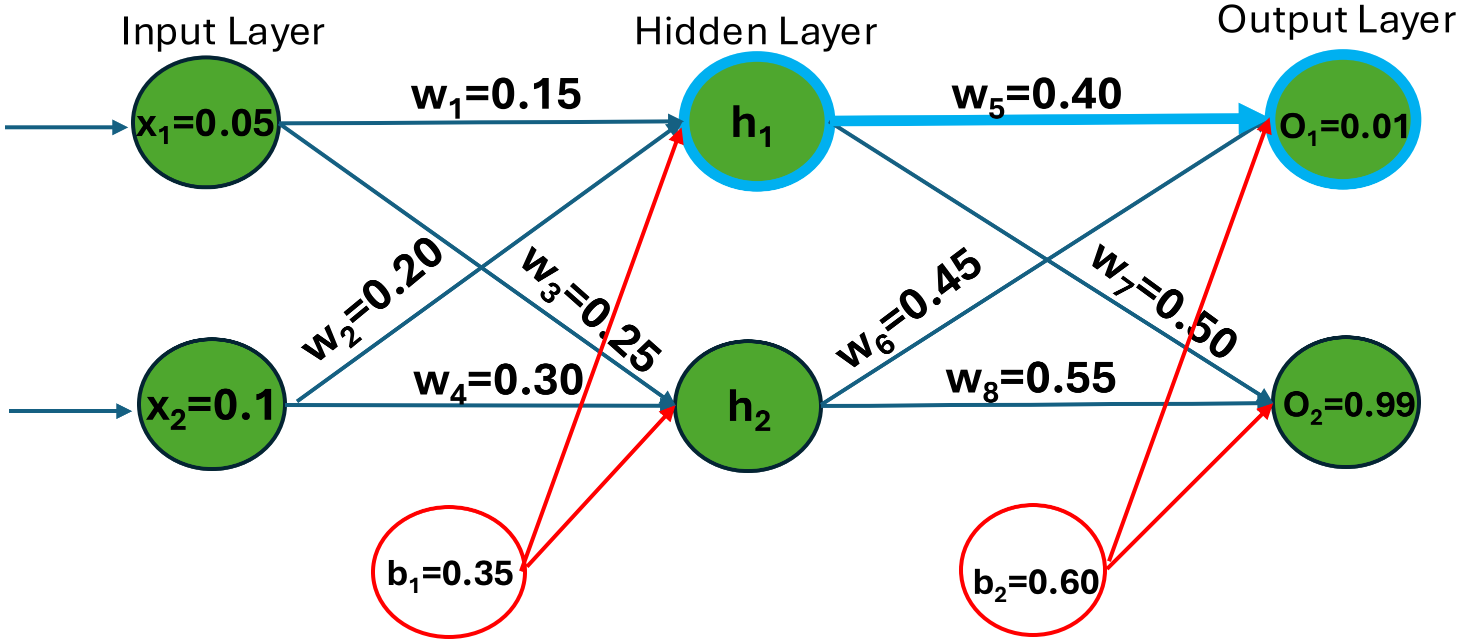
$$\frac{\partial Etotal}{\partial w_5} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial w_5}$$, where;

3) $\frac{\partial out_{h1}}{\partial w_5}$ **= 1 * out_{h1} * $w_5^{(1-1)}$ + 0 +0**

$= 1 * 0.593269992 * 1 + 0 + 0$

$= 0.593269992$

Example of Backpropagation



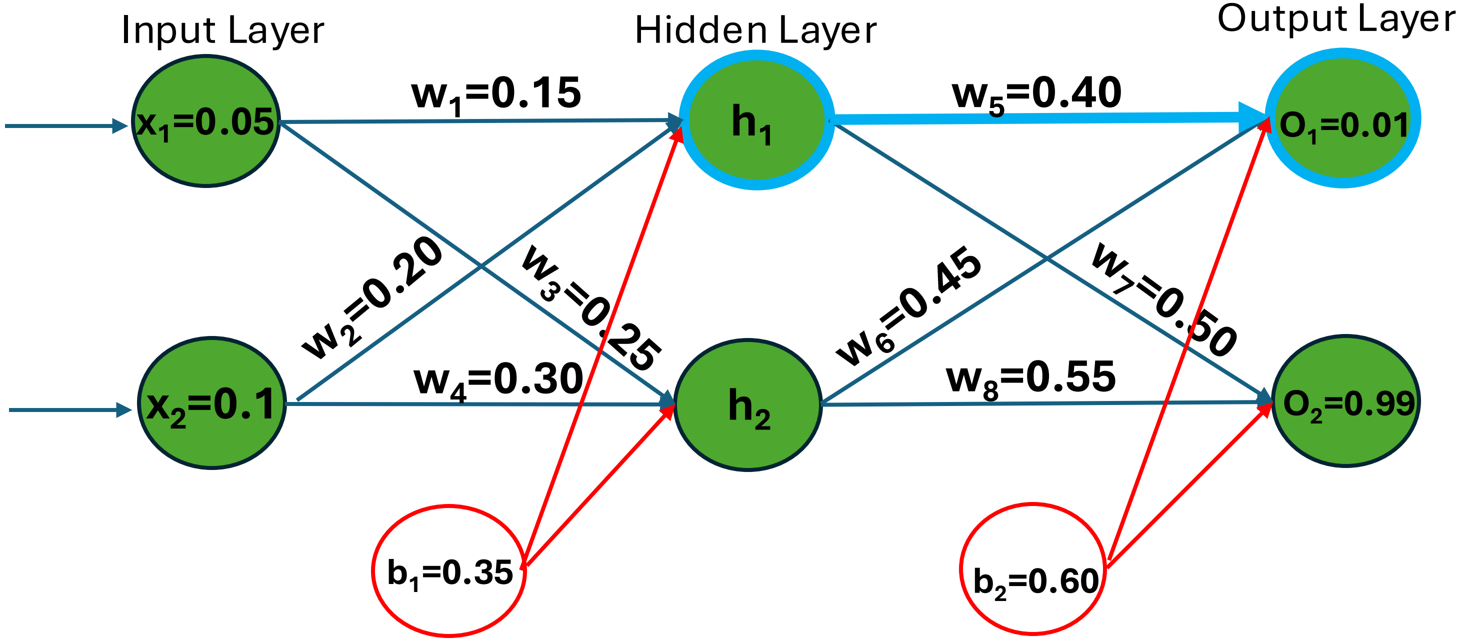
$$\frac{\partial Etotal}{\partial w_5} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial w_5}$$, where;

So,

$$\frac{\partial Etotal}{\partial w_5} = 0.74136507 * 0.186815602 * 0.593269992$$

$$= 0.082167041$$

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_5} = 0.082167041$$

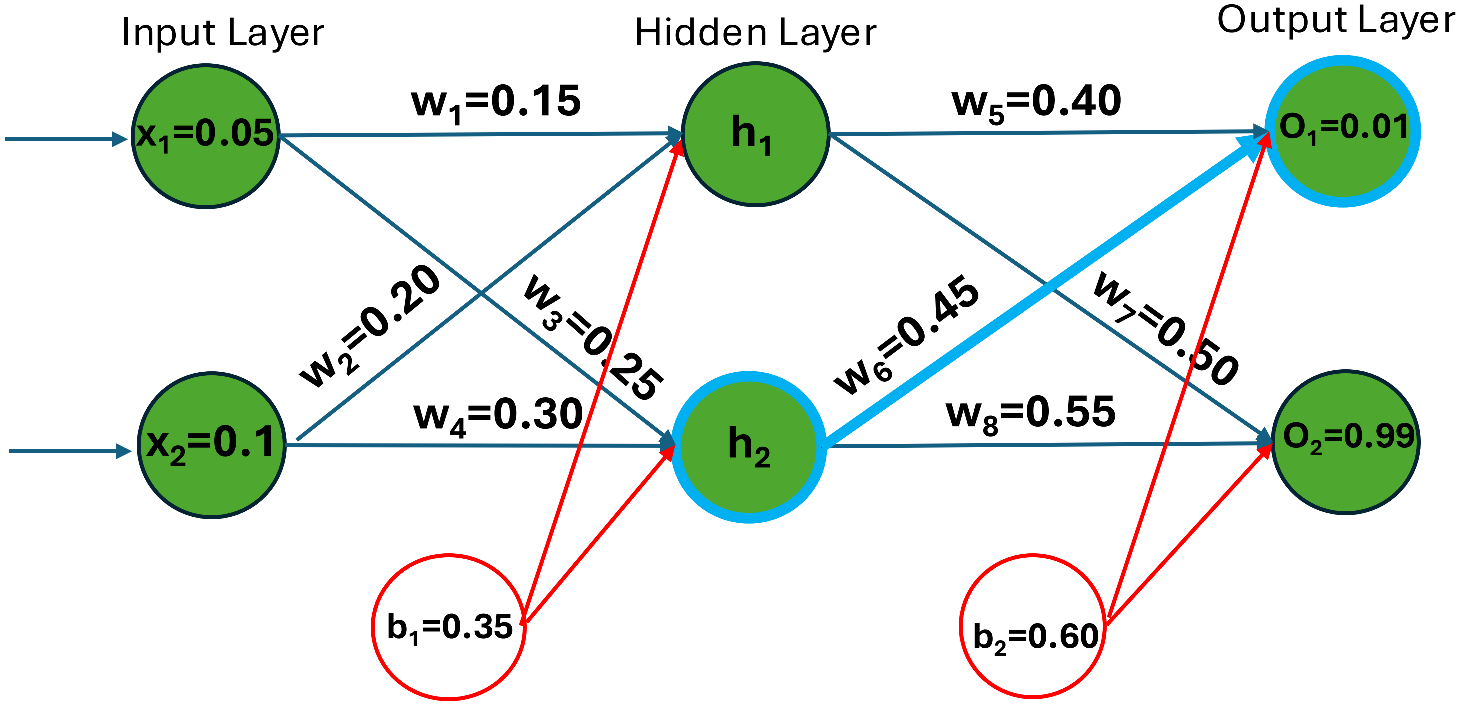
Now, to decrease the error, we need to change the wight (w_5) to the new value (w_5^+) as follow, where η represent the learning rate (assume it is equal = 0.5):

$$w_5^+ = w5 - \eta * \frac{\partial Etotal}{\partial w_5}$$

$$= 0.40 - 0.5 * 0.082167041$$

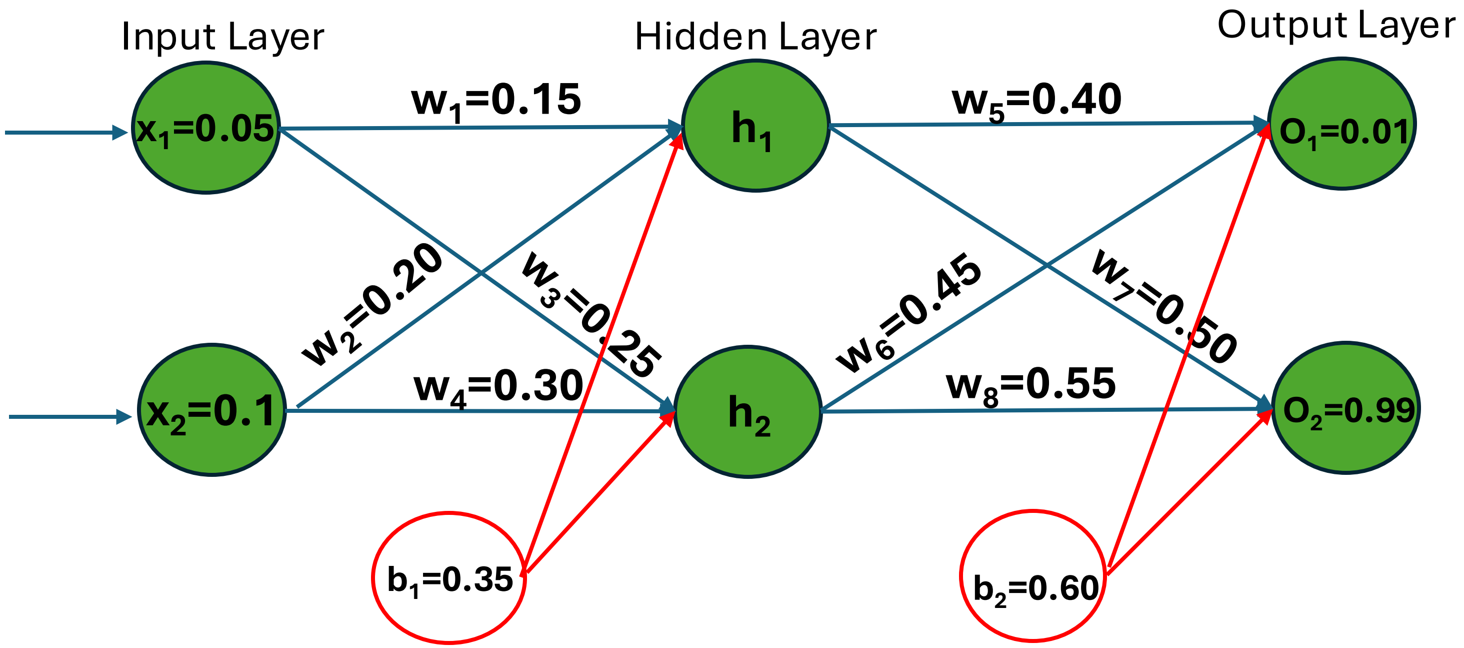
$$= 0.35891648$$

Example of Backpropagation



- After finding out the new value of w_5, the value of values of w_6, w_7, w_8.
- For the value of w_6, There is direct connection (from equations) between E_{total} and out_{o1}, and between out_1 and h_2, and then h_2 and w_6.
- Therefore, we use the following formula to compute the $\frac{\partial E_{total}}{\partial w_6}$, as follow:

Example of Backpropagation



$$\frac{\partial E_{total}}{\partial w_6} = \frac{\partial Etotal}{\partial out_{o1}} * \frac{\partial outo_1}{\partial out_{h2}} * \frac{\partial out_{h2}}{\partial w_6}$$, where;

1) $\frac{\partial_{Etotal}}{\partial out_{o1}} = 2 * \frac{1}{2}\ (target_{o1} - outo_1)^{2\ -\ 1} * -1 + 0$

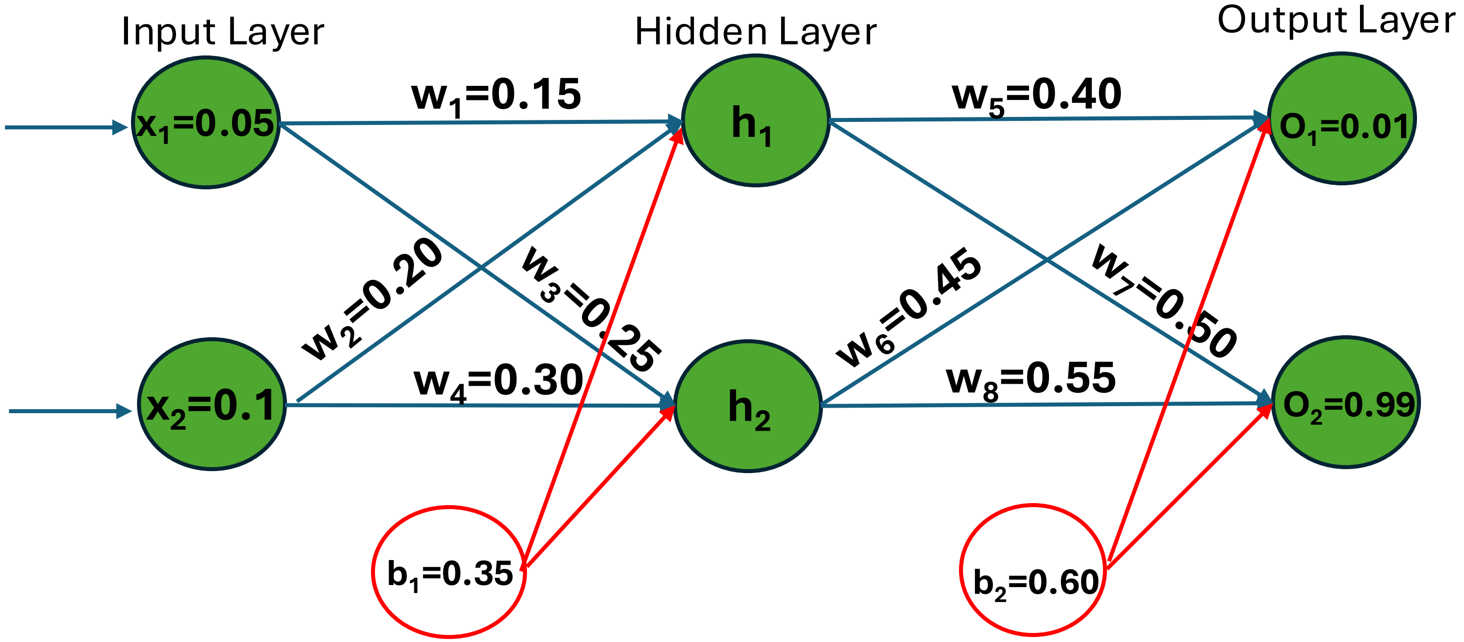
$= out_{o1} - \ target_{o1}$

2) $\frac{\partial outo_1}{\partial out_{h2}} = out_{o1} * (1 - out_{o1})$

3) $\frac{\partial out_{h2}}{\partial w_6} = 1 * out_{h2} * w_6^{(1-1)}$

Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_6} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial outh_2} * \frac{\partial outh_2}{\partial w_6}$$, where;

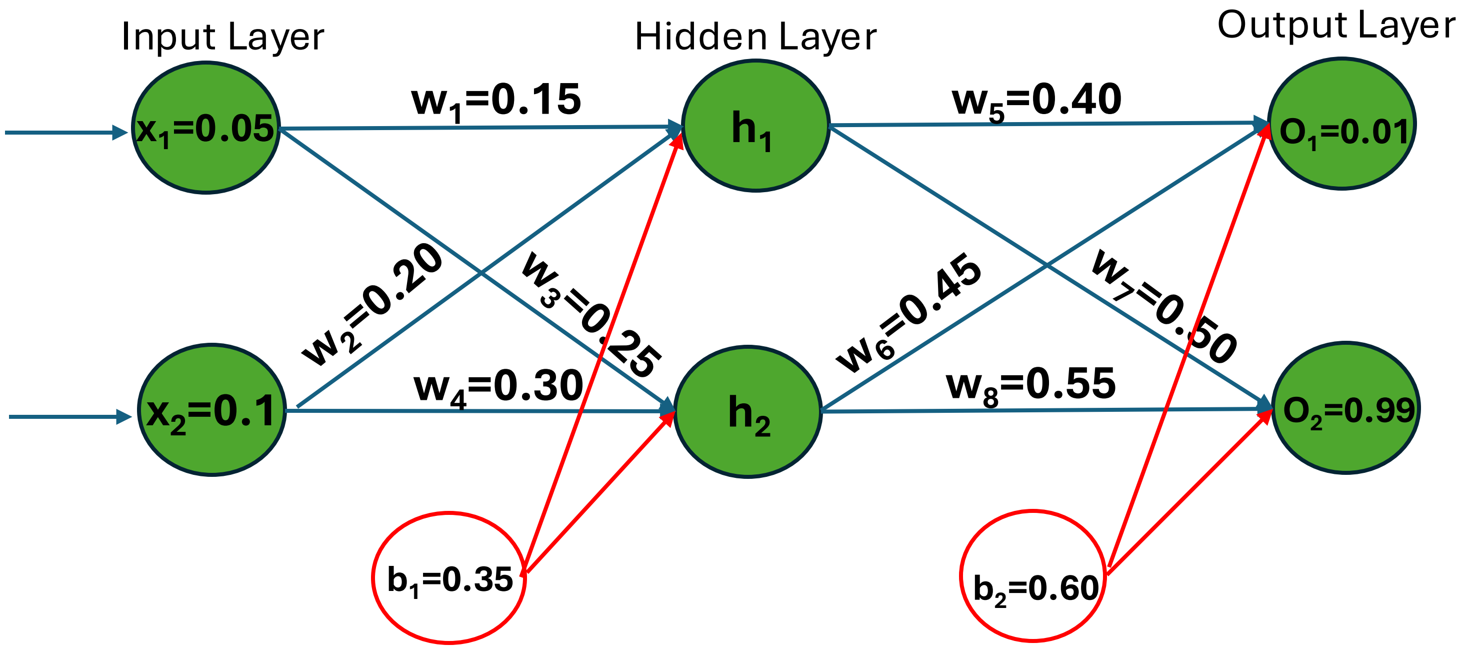
1) $\frac{\partial_{Etotal}}{\partial outo_1} = 2 * \frac{1}{2}\,(target_{o1} - outo_1)^{2-1} * -1 + 0$

$= out_{o1} - targeto_1$

$= 0.75136507 - 0.01$

$= 0.74136507$

Example of Backpropagation



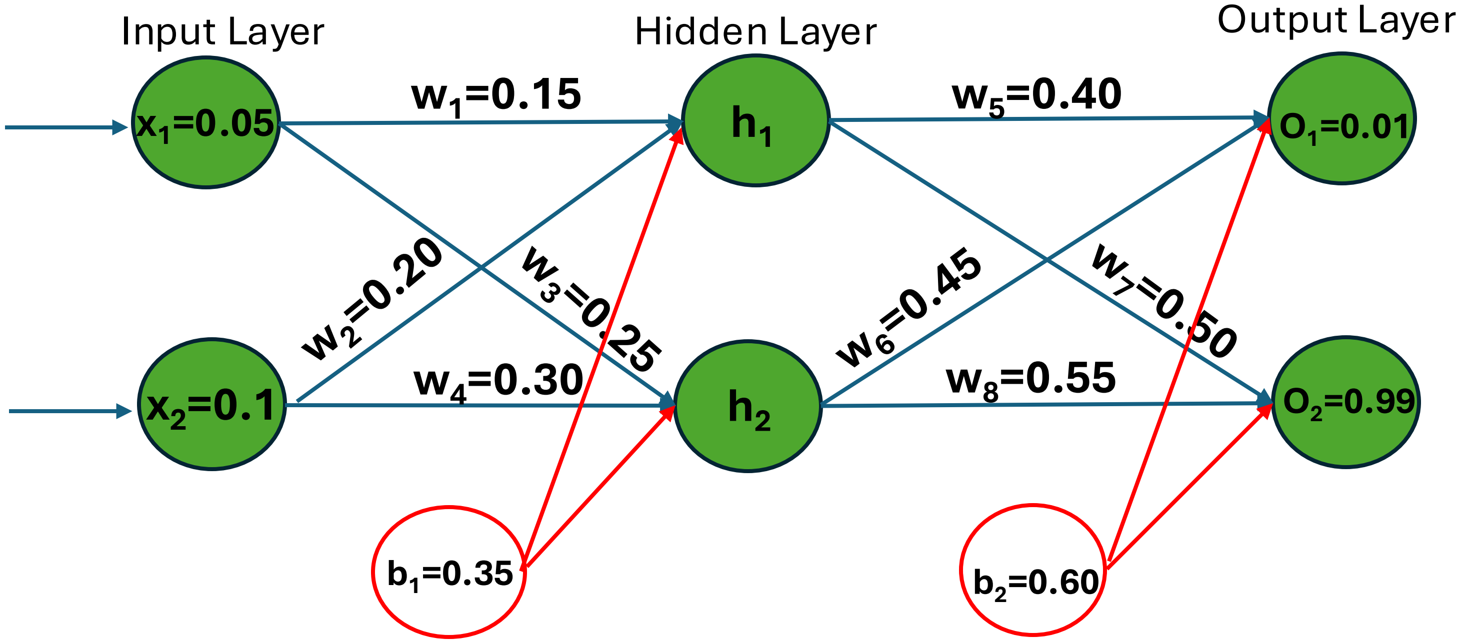
$$\frac{\partial Etotal}{\partial w_6} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial outh_2} * \frac{\partial outh_2}{\partial w_6}$$, where;

2) $\frac{\partial outo_1}{\partial outh_2} = out_{o1} * (1 - out_{o1})$

$= 0.75136507 * (1 - 0.75136507)$

$= 0.186815602$

Example of Backpropagation



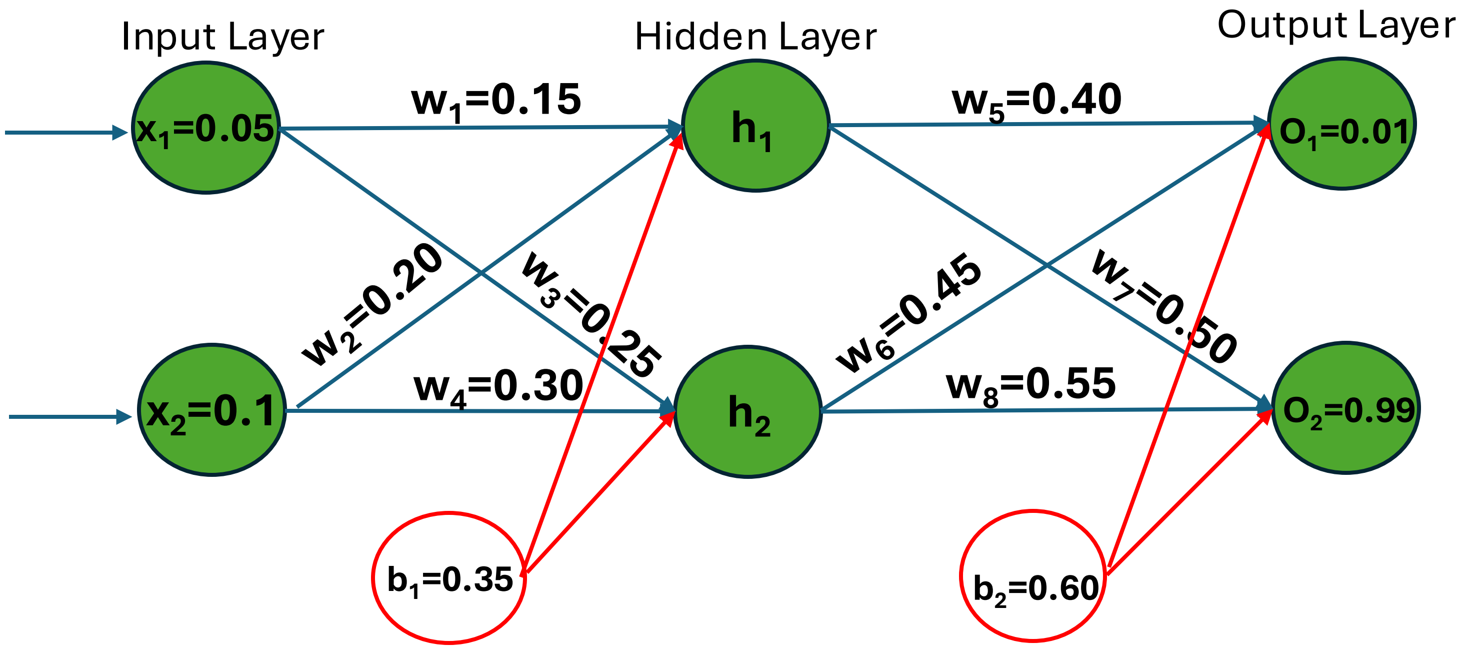
$$\frac{\partial Etotal}{\partial w_6} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial outh_2} * \frac{\partial outh_2}{\partial w_6}$$, where;

3) $\frac{\partial out_{h2}}{\partial w_6}$ **= 1 * out_{h2} * $w_6^{(1-1)}$ + 0 +0**

= 1 * 0.596884378 * 1 + 0 + 0

= 0.596884378

Example of Backpropagation



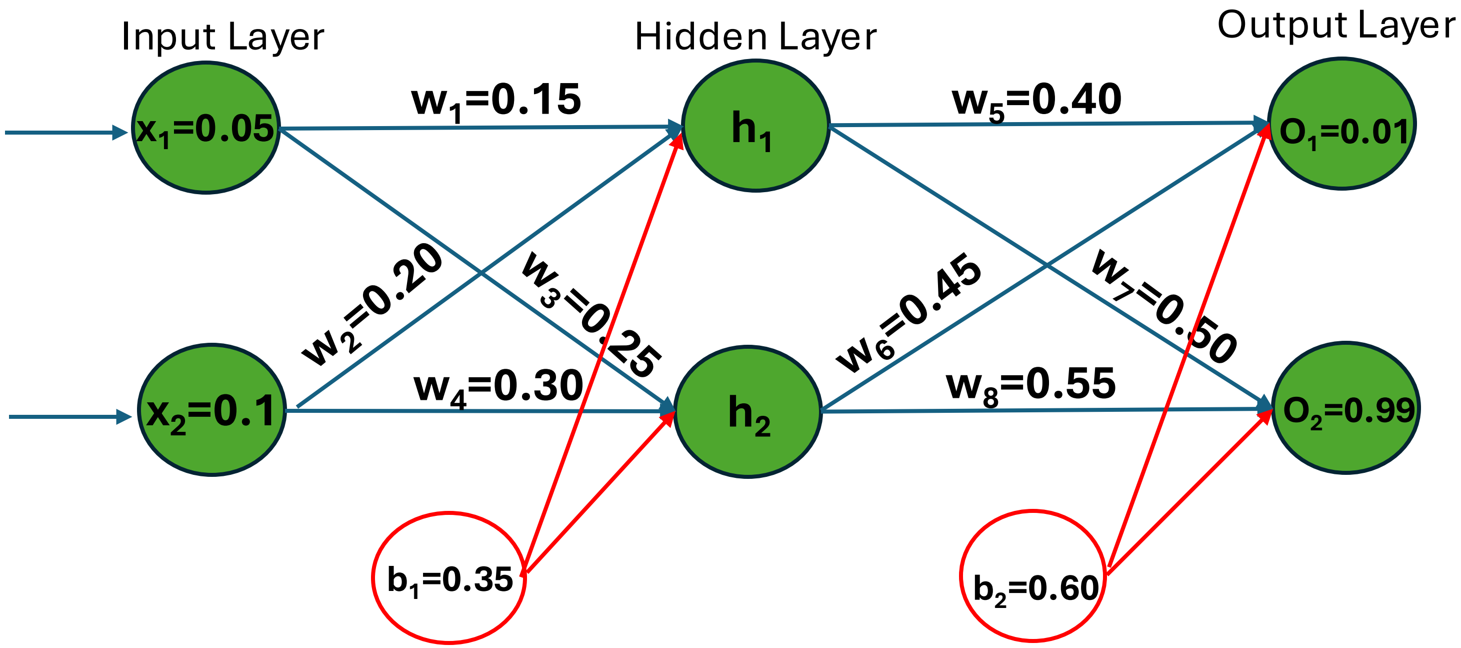
$$\frac{\partial Etotal}{\partial w_6} = \frac{\partial Etotal}{\partial outo_1} * \frac{\partial outo_1}{\partial outh_2} * \frac{\partial outh_2}{\partial w_6}$$, where;

So,

$$\frac{\partial Etotal}{\partial w_6} = 0.74136507 * 0.186815602 * 0.596884378$$

$= 0.082667628$

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_6} = 0.082667628$$

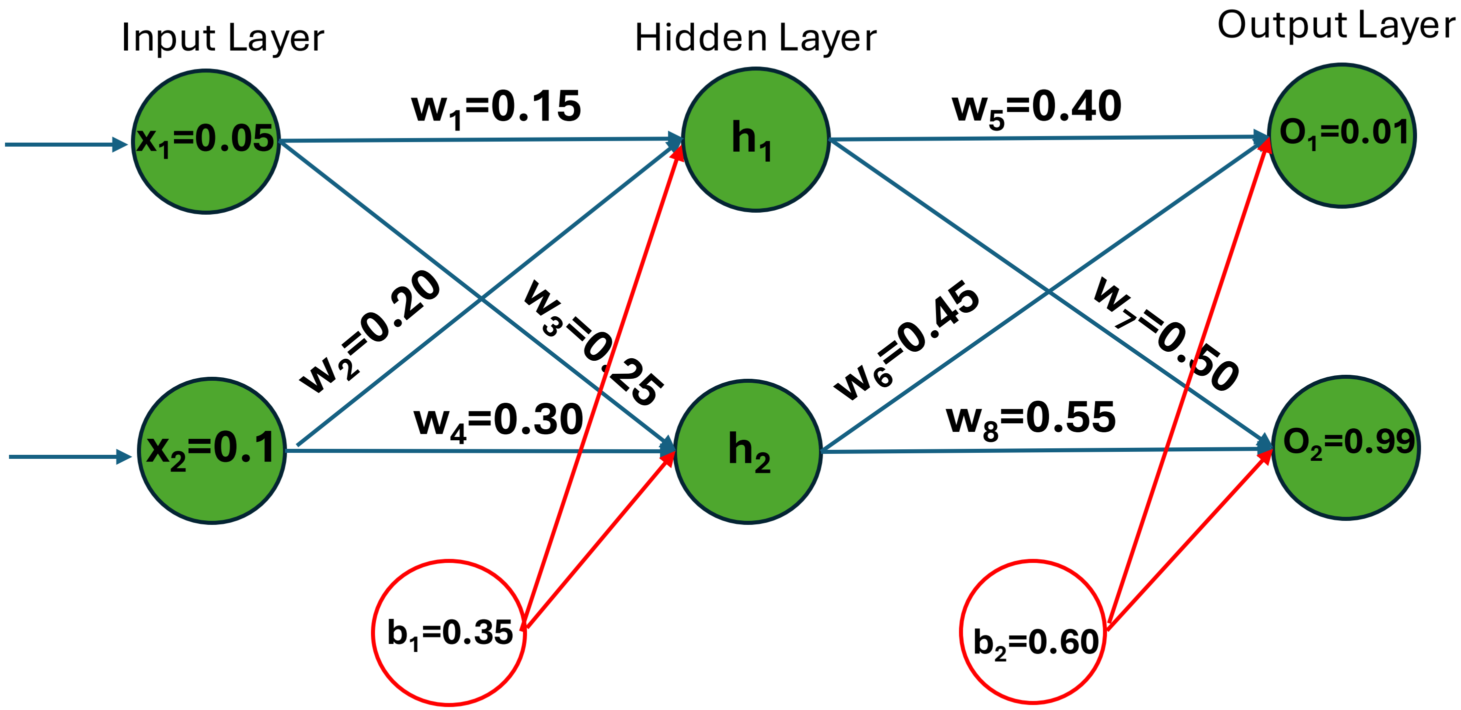
Now, to decrease the error, we need to change the wight (w_6) to the new value (w_6^+) as follow, where η represent the learning rate (assume it is equal = 0.5):

$$w_6^+ = w_6 - \eta * \frac{\partial Etotal}{\partial w_6}$$

$$= 0.45 – 0.5 * 0.082667628$$

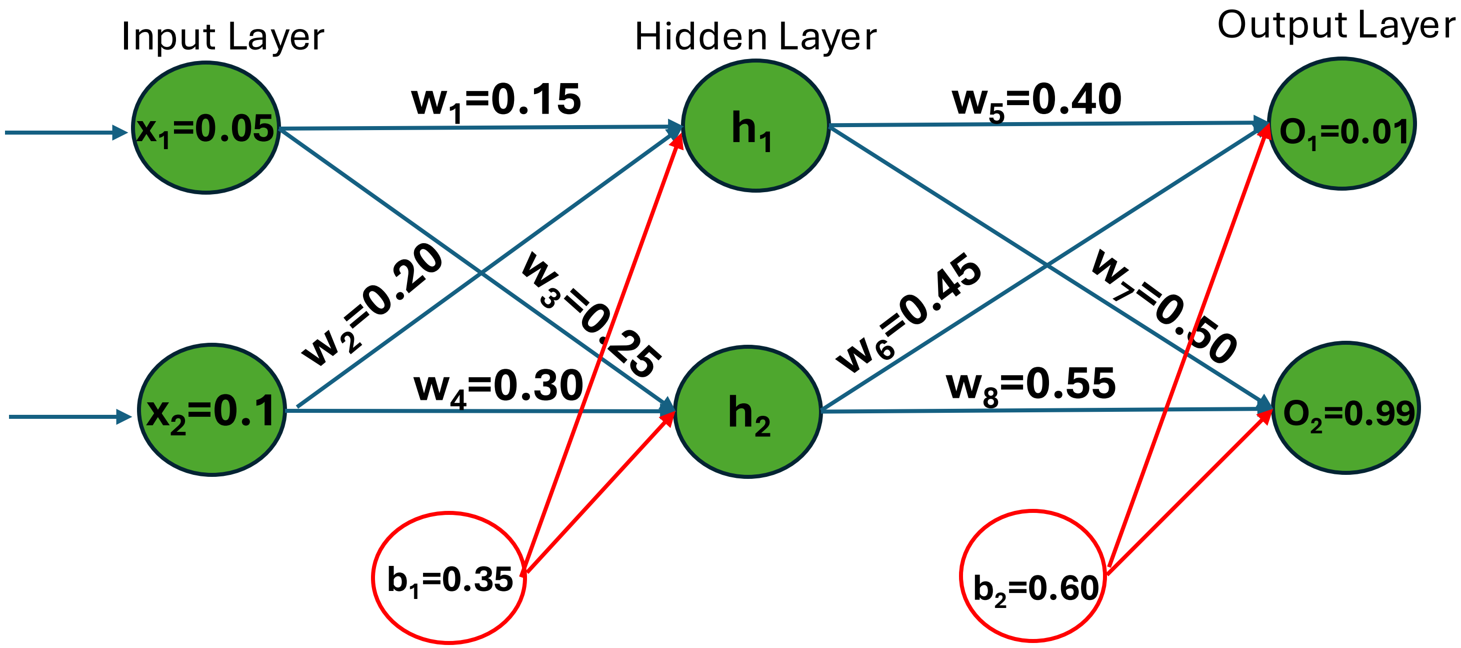
$$= 0.408666186$$

Example of Backpropagation



- After finding out the new value of w_6, the value of values of w_7, and w_8.
- For the value of w_7, There is direct connection (from equations) between E_{total} and out_{o2}, and between out_2 and h_1, and then h_1 and w_7.
- Therefore, we use the following formula to compute the $\frac{\partial E_{total}}{\partial w_7}$, as follow:

Example of Backpropagation



$$\frac{\partial E_{total}}{\partial w_7} = \frac{\partial Etotal}{\partial out_{o2}} * \frac{\partial outo_2}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial w_7}$$, where;

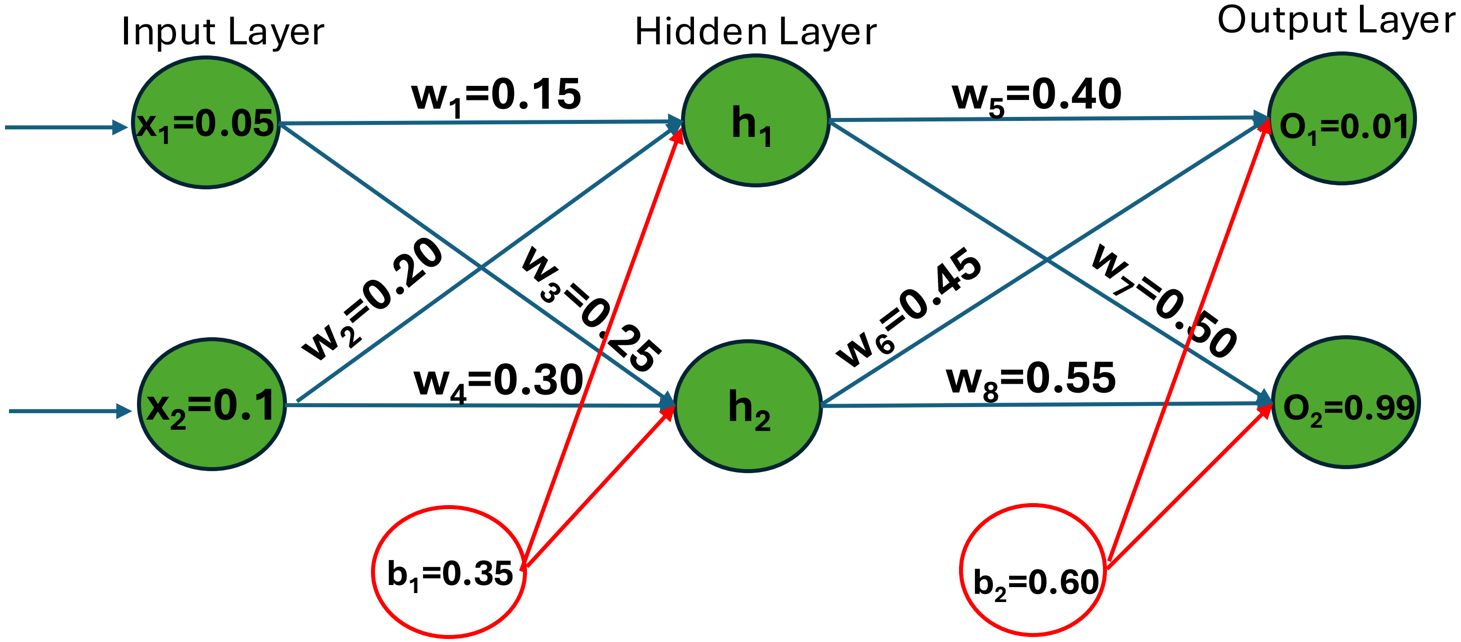
1) $\frac{\partial_{Etotal}}{\partial out_{o2}} = 2 * \frac{1}{2}\ (target_{o2} - outo_2)^{2\ -\ 1} * -1 + 0$

$= out_{o2}\ \_\ target_{o2}$

2) $\frac{\partial outo_2}{\partial out_{h1}} = out_{o2} * (1 - out_{o2})$

3) $\frac{\partial out_{h1}}{\partial w_7} = 1 * out_{h1} * w_7^{(1-1)}$

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_7} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1} * \frac{\partial outh_1}{\partial w_7}$$, where;

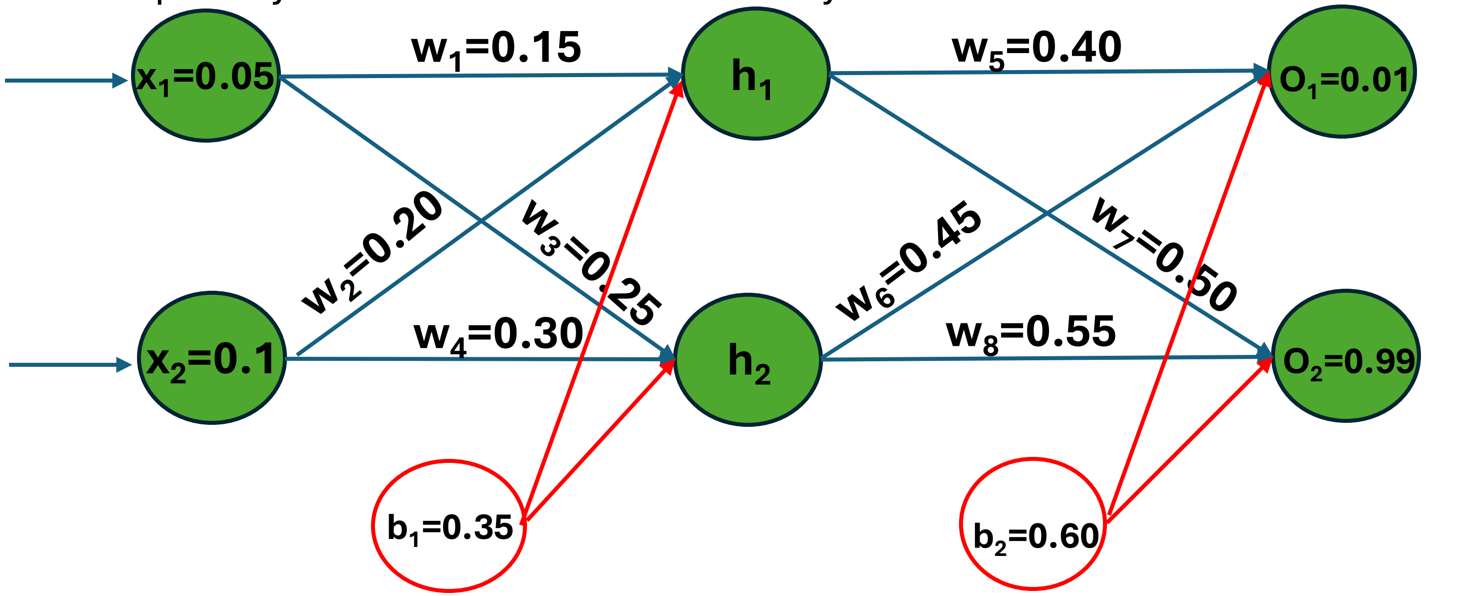
1) $\frac{\partial_{Etotal}}{\partial outo_2} = 2 * \frac{1}{2}\ (target_{o2} - outo_2)^{2\ -\ 1} * -1 + 0$

$= out_{o2}\ \_\ target_{o2}$

$= 0.772928465 - 0.99$

$= -0.217071535$

Example of Backpropagation



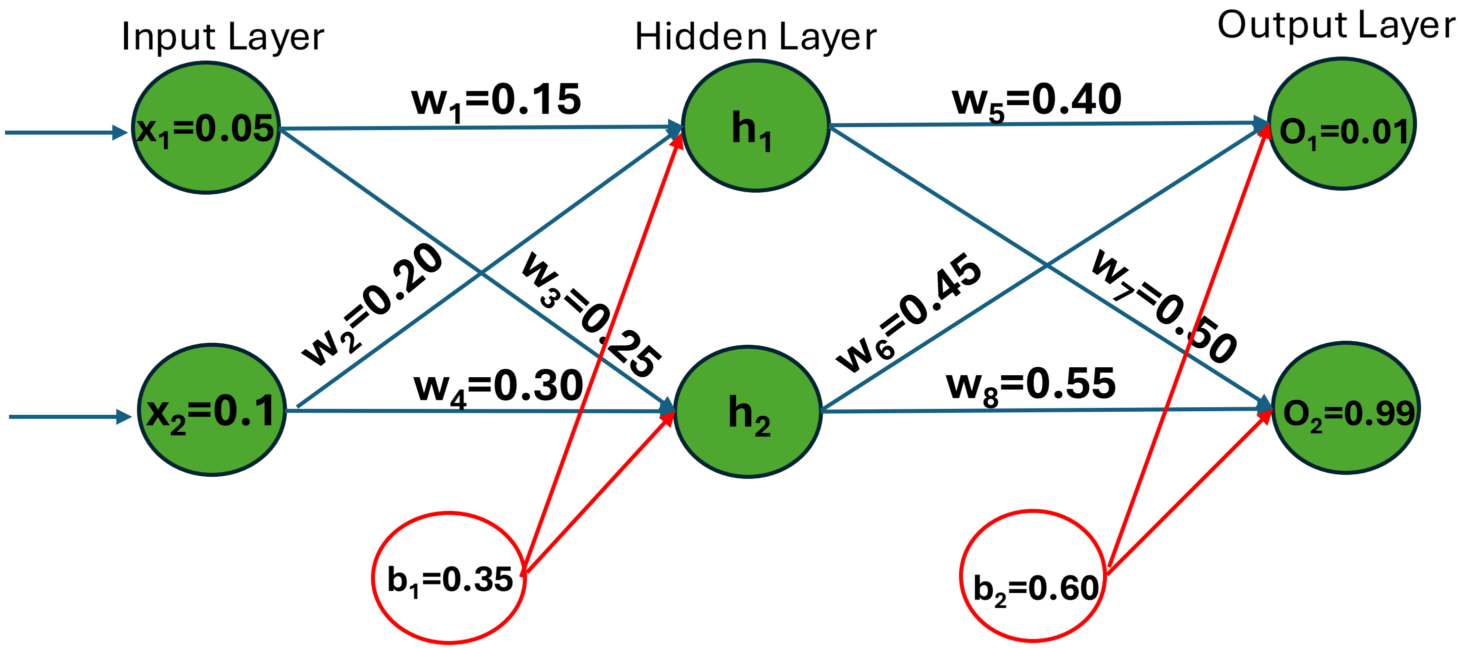
$$\frac{\partial Etotal}{\partial w_7} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1} * \frac{\partial outh_1}{\partial w_7}$$, where;

2) $\frac{\partial outo_2}{\partial outh_1} = out_{o2} * (1 - out_{o2})$

$= 0.772928465 * (1 - 0.772928465)$

$= 0.17551005299$

Example of Backpropagation



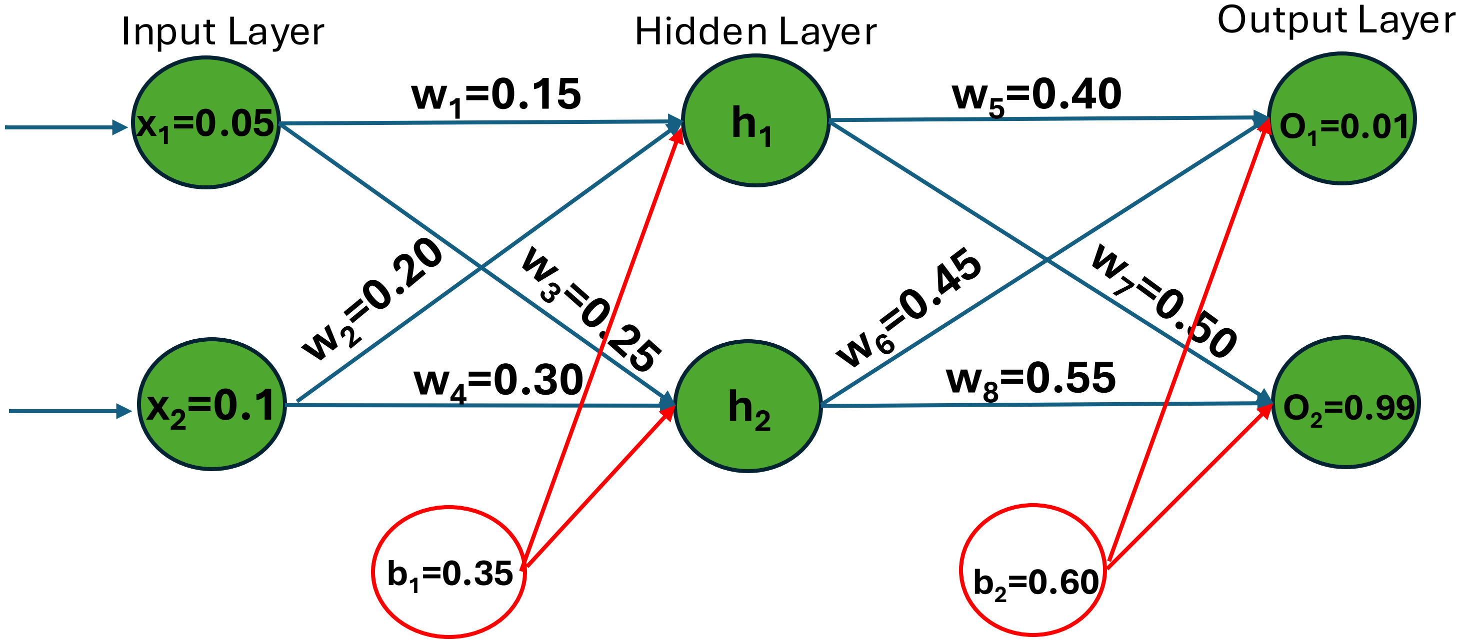
$$\frac{\partial Etotal}{\partial w_7} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1} * \frac{\partial outh_1}{\partial w_7}$$, where;

3) $\frac{\partial out_{h1}}{\partial w_7}$ **= 1 * out_{h1} * $w_7^{(1-1)}$**

= 1 * 0.593269992 * 1 + 0 + 0

= 0.593269992

Example of Backpropagation



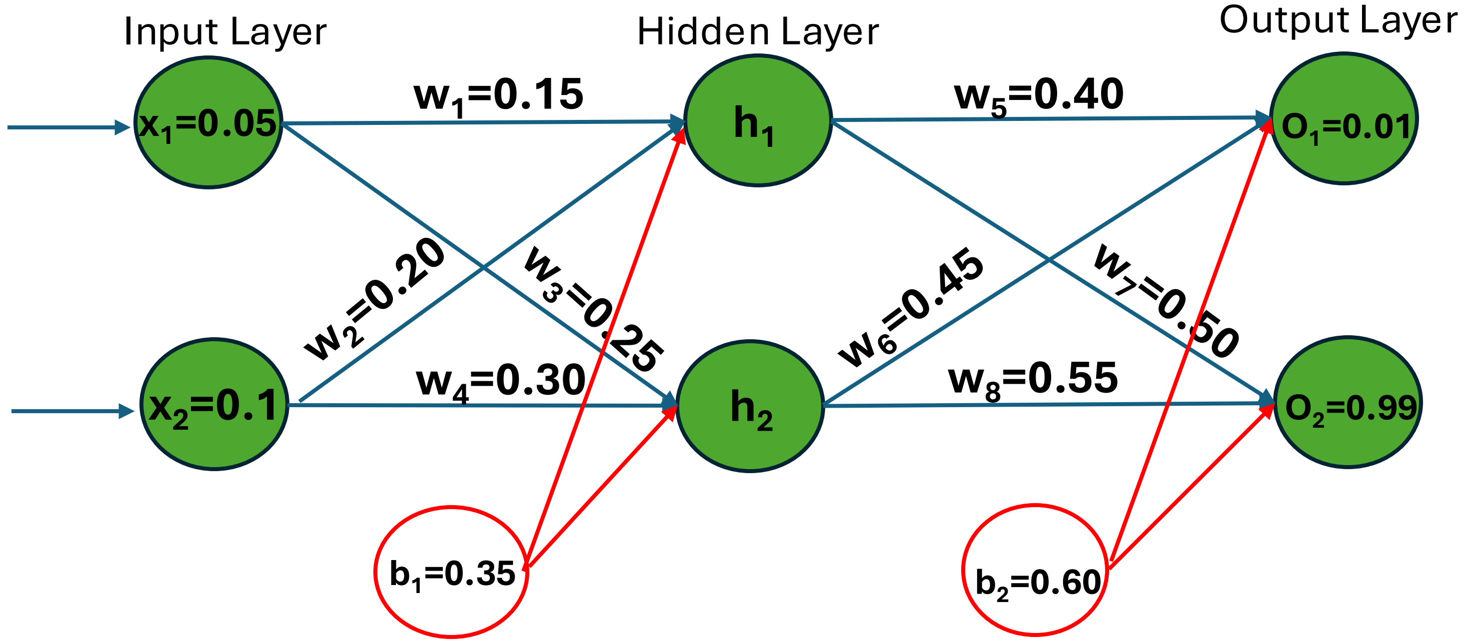
$$\frac{\partial Etotal}{\partial w_7} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1} * \frac{\partial outh_1}{\partial w_7}$$, where;

So,

$\frac{\partial Etotal}{\partial w_7}$ = -0.217071535 * 0.17551005299 * 0.593269992

= -0.02260254052

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_7} = -0.02260254052$$

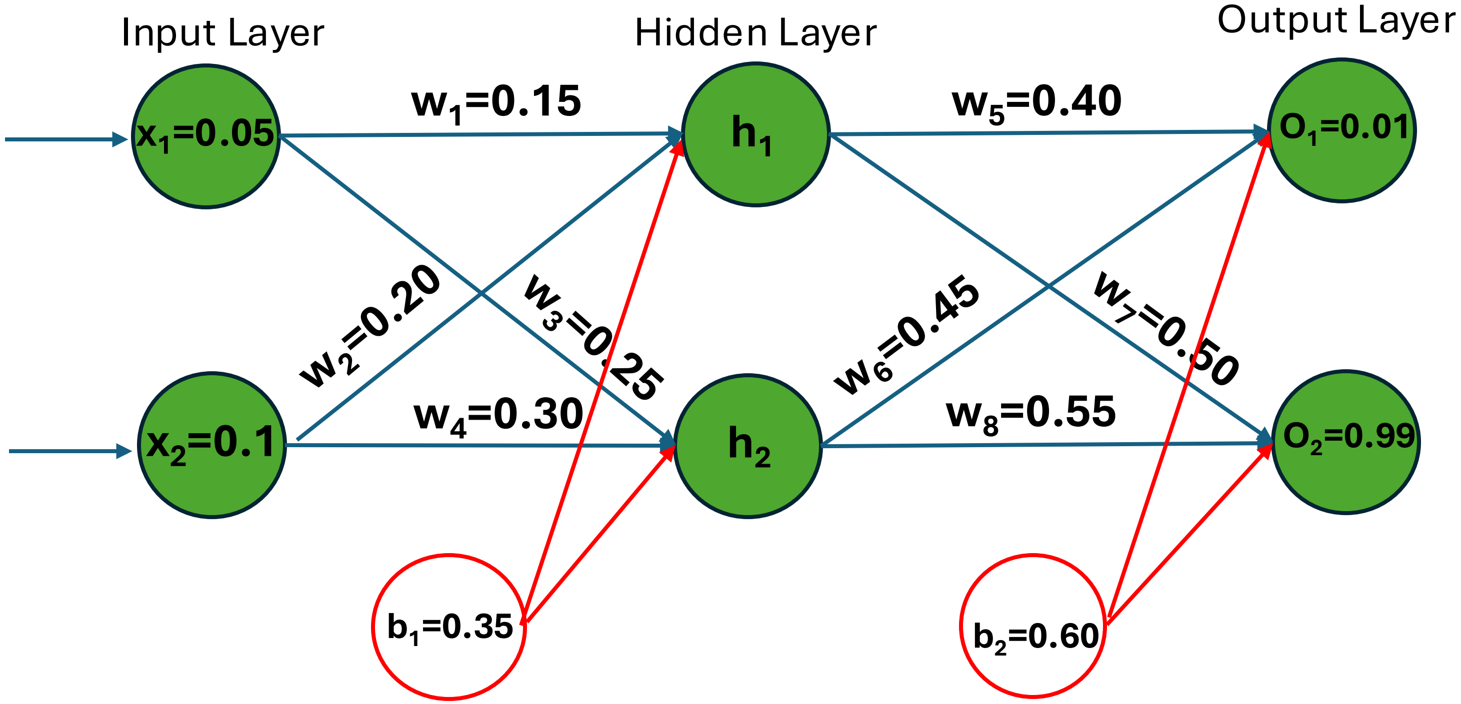
Now, to decrease the error, we need to change the wight (w_7) to the new value (w_7^+) as follow, where η represent the learning rate (assume it is equal = 0.5):

$$w_7^+ = w_7 - \eta * \frac{\partial Etotal}{\partial w_7}$$

$$= 0.50 – 0.5 * -0.02260254052$$

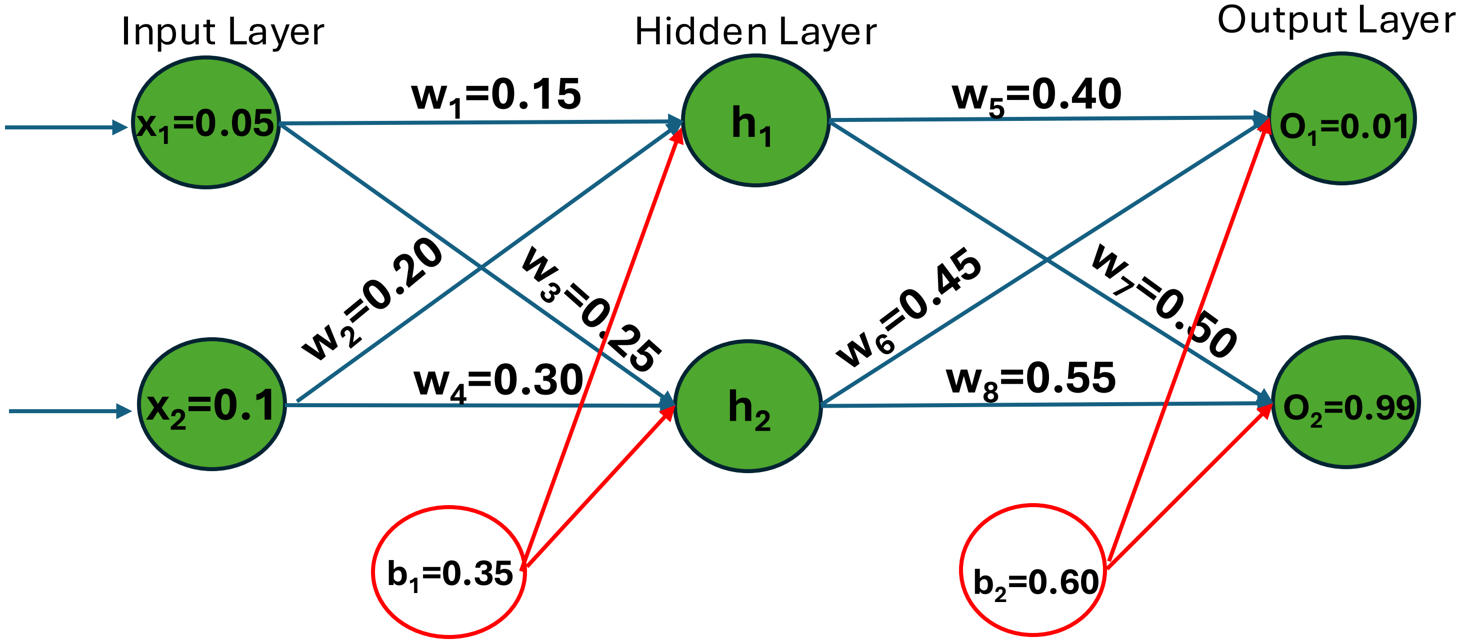
$$= 0.511301270$$

Example of Backpropagation



- After finding out the new value of w_7, the value of values of w_8.
- For the value of w_8, There is direct connection (from equations) between E_{total} and out_{o2}, and between out_2 and h_2, and then h_2 and w_8.
- Therefore, we use the following formula to compute the $\frac{\partial E_{total}}{\partial w_8}$, as follow:

Example of Backpropagation



$$\frac{\partial E_{total}}{\partial w_8} = \frac{\partial Etotal}{\partial out_{o2}} * \frac{\partial outo_2}{\partial out_{h2}} * \frac{\partial out_{h2}}{\partial w_8}$$, where;

1) $\frac{\partial Etotal}{\partial out_{o2}} = 2 * \frac{1}{2}\left(target_{o2} - outo_2\right)^{2-1} * -1 + 0$

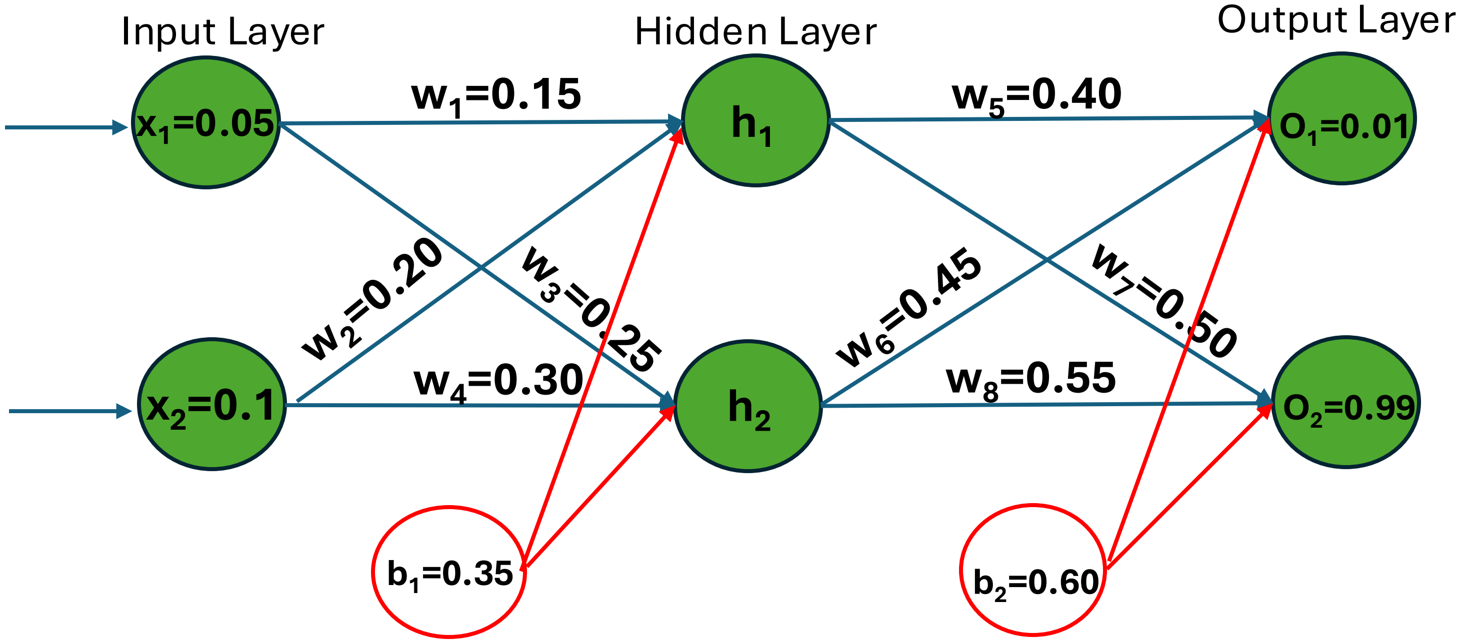
$= out_{o2} \_ \ target_{o2}$

2) $\frac{\partial outo_2}{\partial out_{h2}} = out_{o2} * (1 - out_{o2})$

3) $\frac{\partial out_{h2}}{\partial w_8} = 1 * out_{h2} * w_8^{(1-1)}$

Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

Example of Backpropagation



$$\frac{\partial Etotal}{\partial w_8} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2} * \frac{\partial outh_2}{\partial w_8}$$, where;

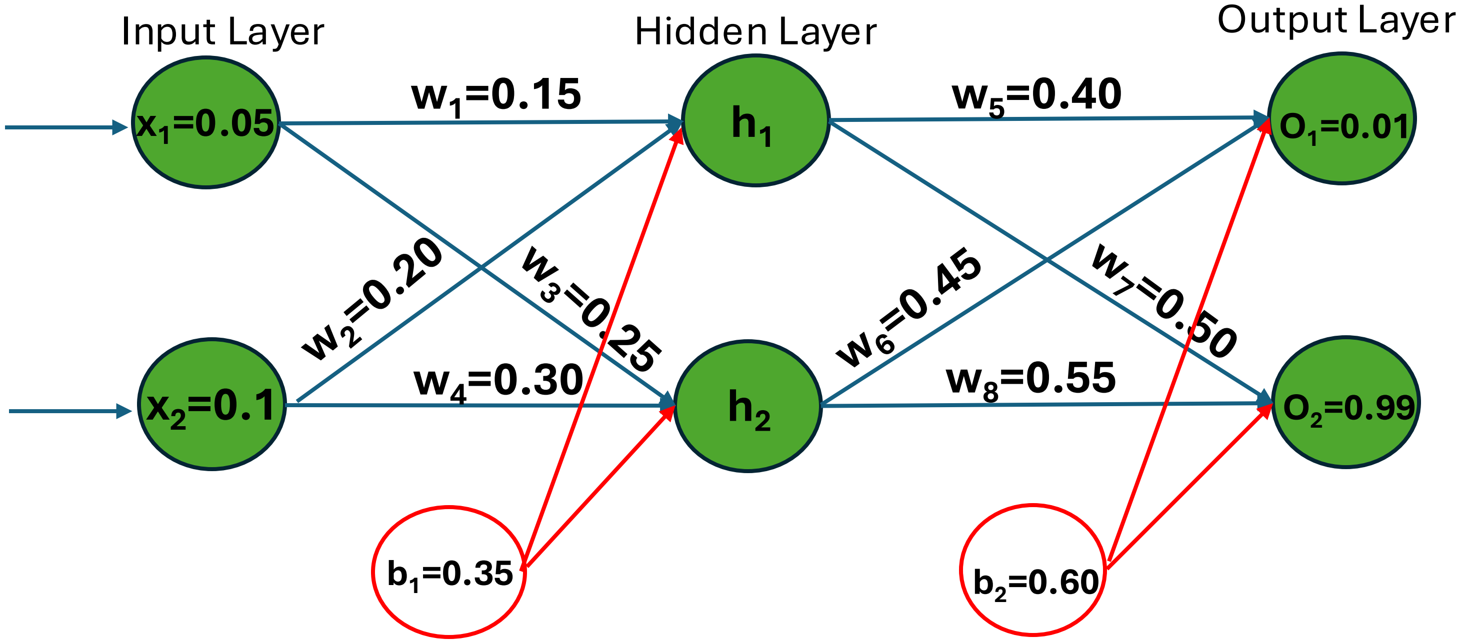
1) $\frac{\partial Etotal}{\partial outo_2} = 2 * \frac{1}{2}\ (target_{o2} - outo_2)^{2\ -\ 1} * -1 + 0$

$= out_{o2}\ \_\ target_{o2}$

= 0.772928465 – 0.99

= -0.217071535

Example of Backpropagation



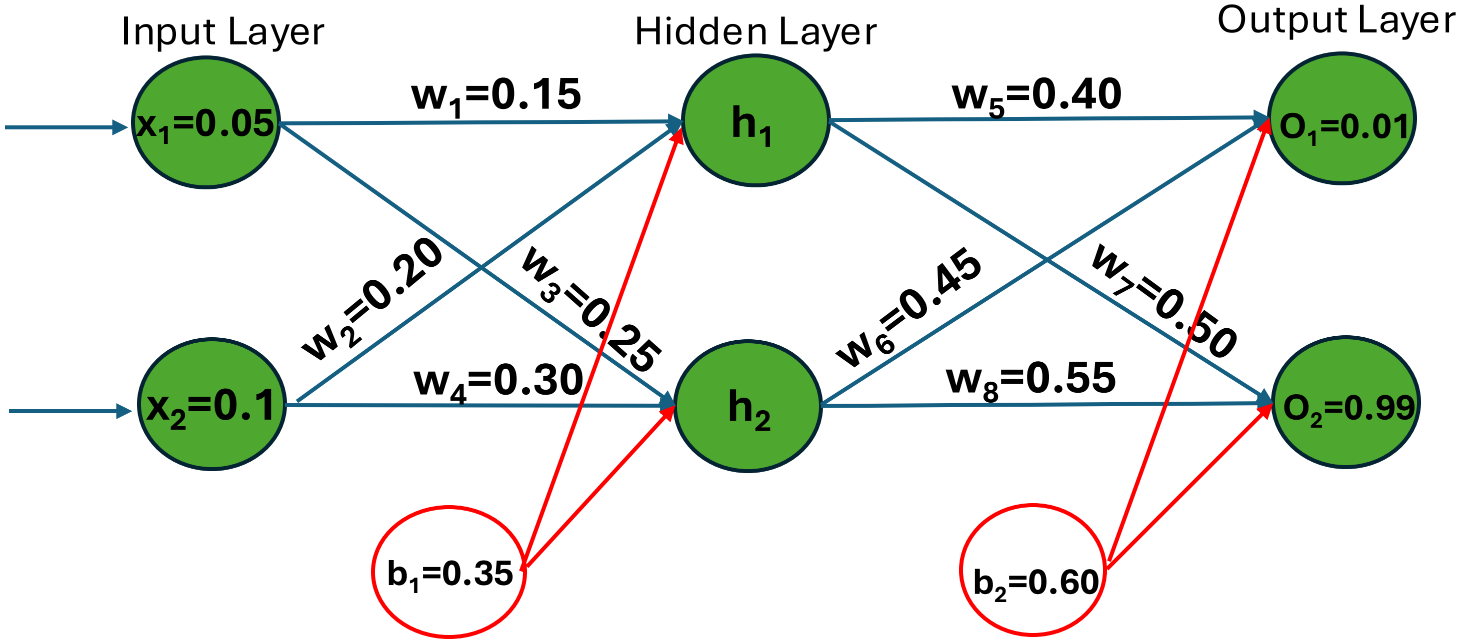
$$\frac{\partial Etotal}{\partial w_8} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2} * \frac{\partial outh_2}{\partial w_8}$$, where;

2) $\frac{\partial outo_2}{\partial outh_2} = out_{o2} * (1 - out_{o2})$

$= 0.772928465 * (1 - 0.772928465)$

$= 0.17551005299$

Example of Backpropagation



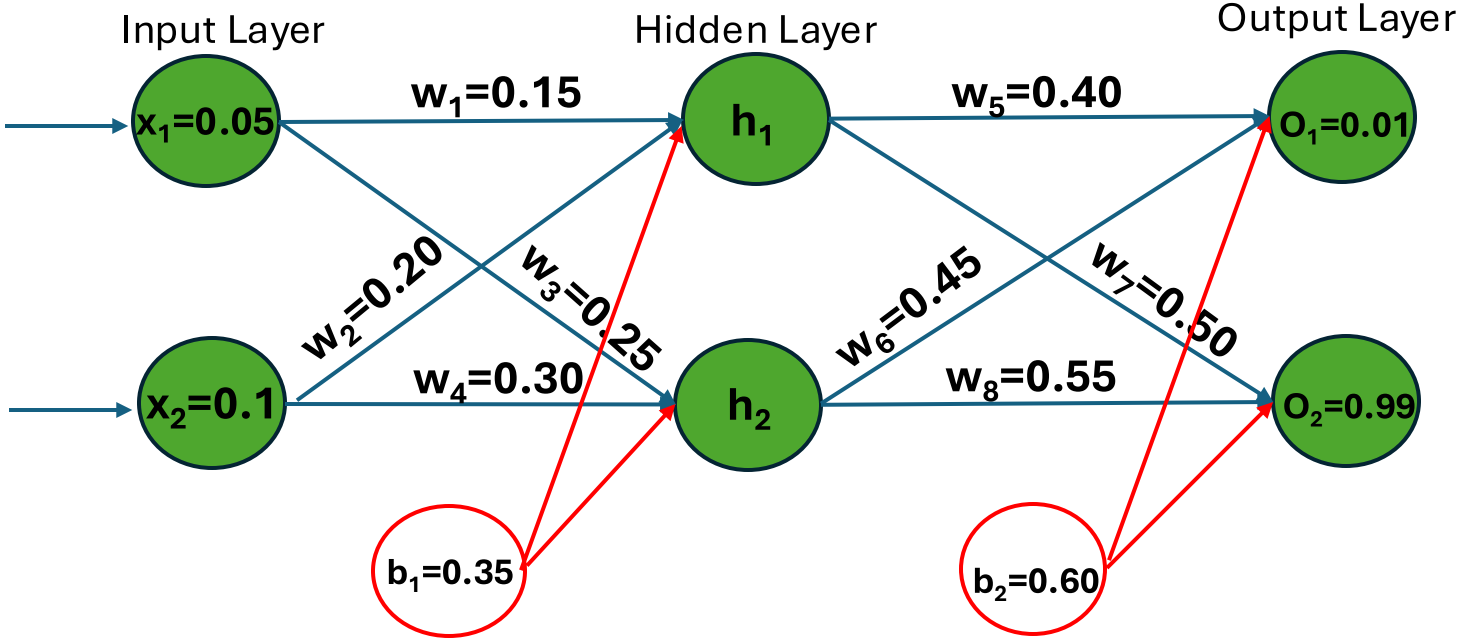
$$\frac{\partial Etotal}{\partial w_8} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2} * \frac{\partial outh_2}{\partial w_8}$$, where;

3) $\frac{\partial out_{h2}}{\partial w_8}$ **= 1 * out$_{h2}$* w$_8^{(1-1)}$**

= 1 * 0.596884378 * 1 + 0 + 0

= 0.596884378

Example of Backpropagation



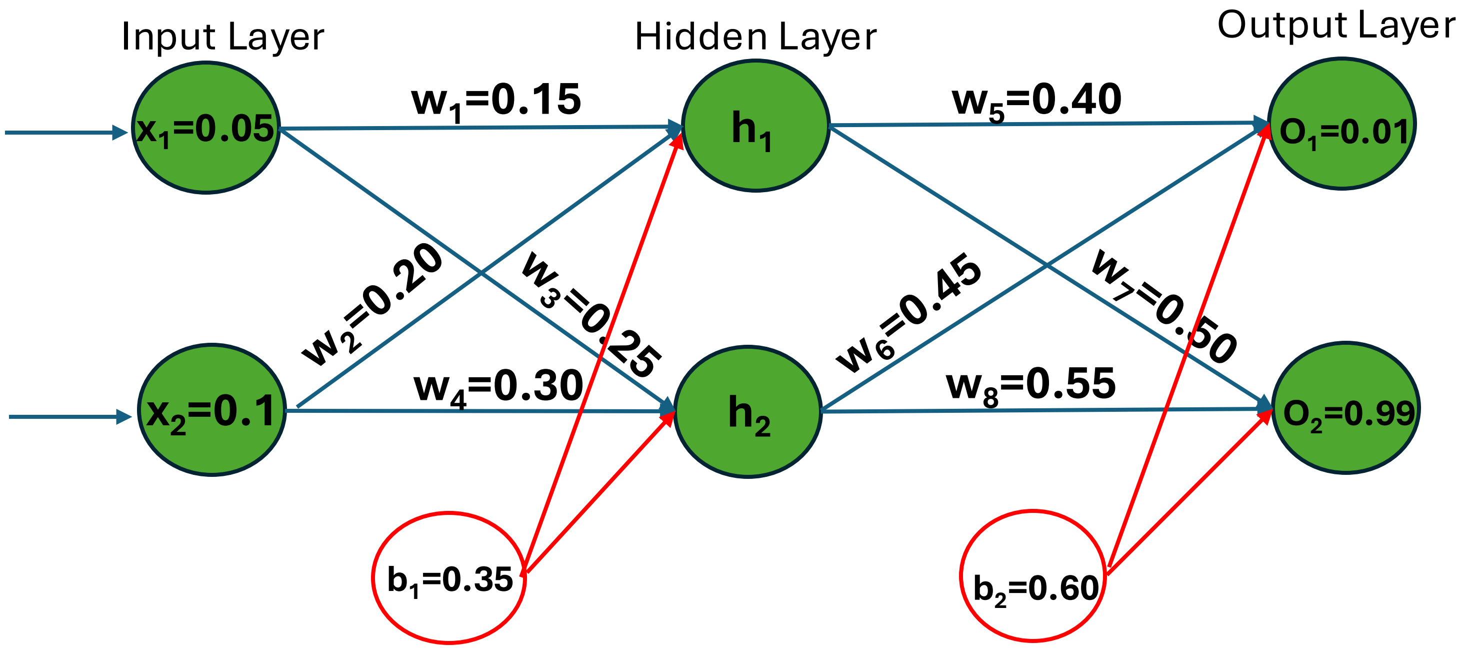
$$\frac{\partial Etotal}{\partial w_8} = \frac{\partial Etotal}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2} * \frac{\partial outh_2}{\partial w_8}$$, where;

So,

$\frac{\partial Etotal}{\partial w_8}$ = -0.217071535 * 0.17551005299 * 0.596884378

= -0.02274024226

Example of Backpropagation



$\frac{\partial Etotal}{\partial w_8}$ = -0.02274024226

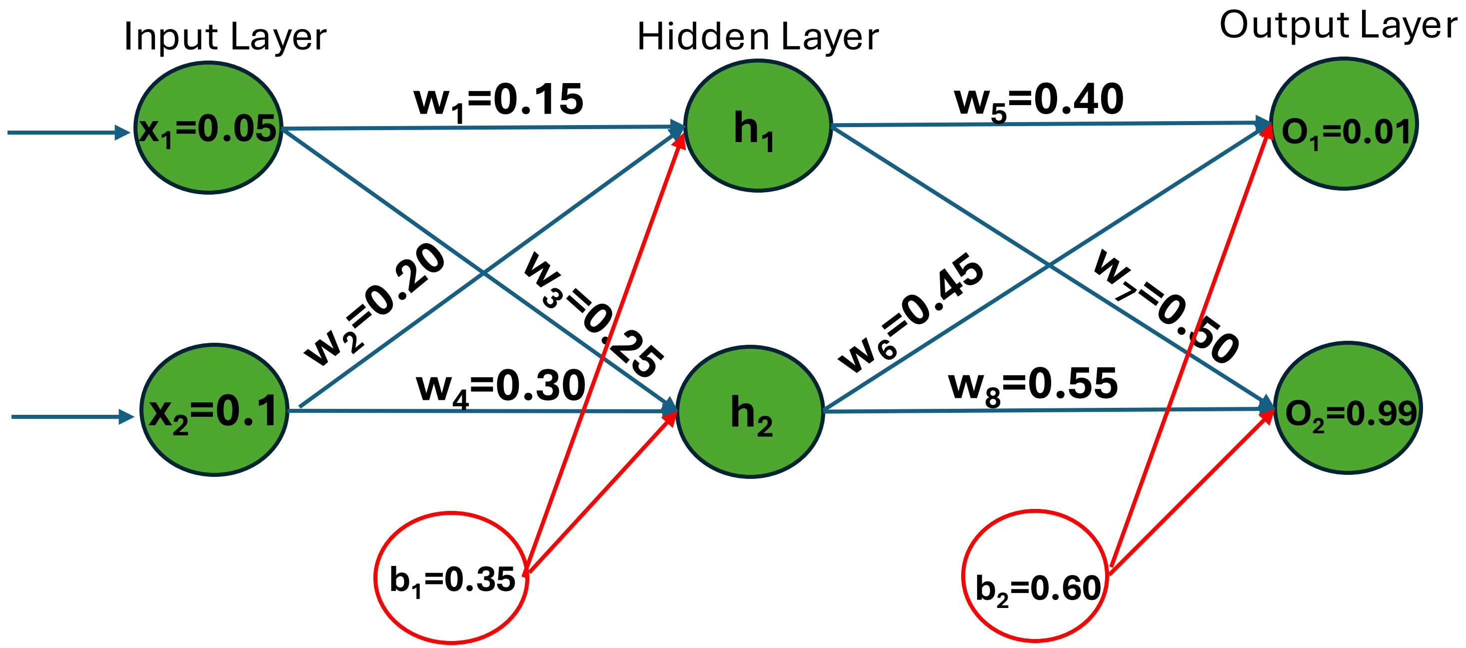
Now, to decrease the error, we need to change the wight (w_8) to the new value (w_8^+) as follow, where η represent the learning rate (assume it is equal = 0.5):

$$w_8^+ = w_8 - \eta * \frac{\partial Etotal}{\partial w_8}$$

$$= 0.55 – 0.5 * -0.02274024226$$

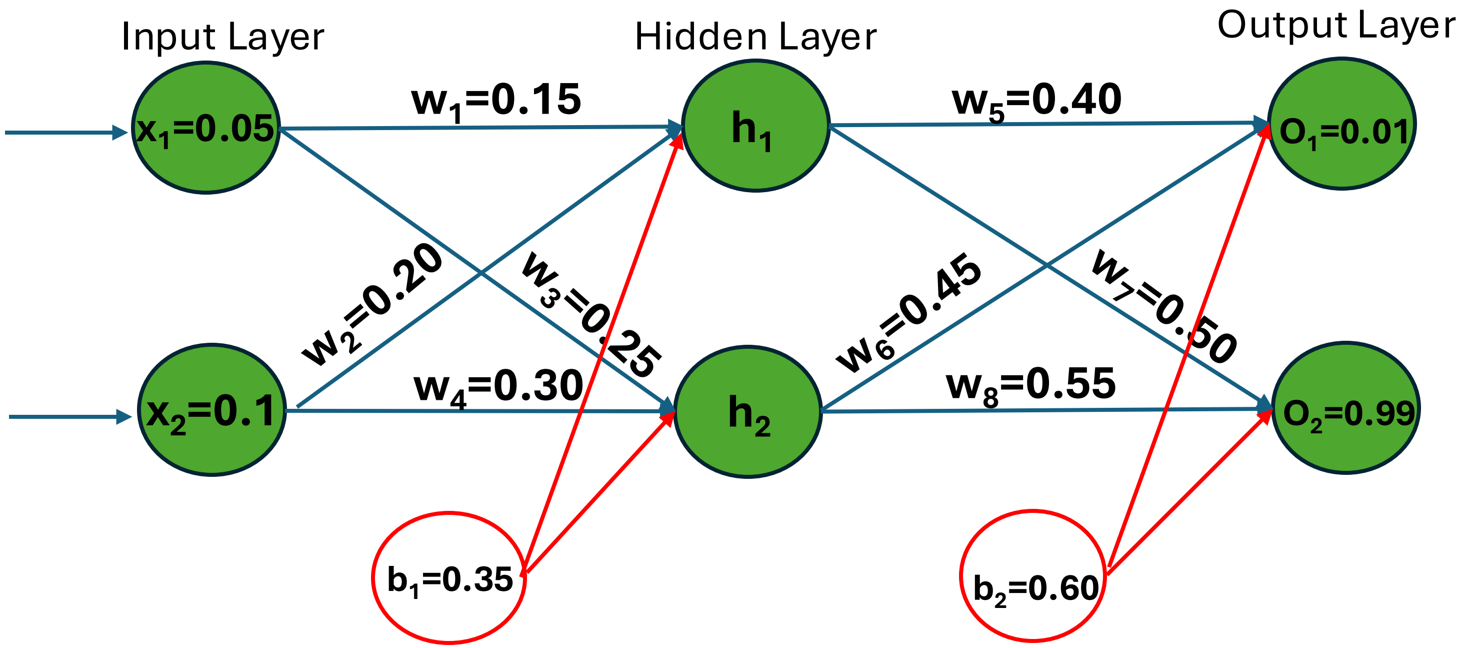
$$= 0.561370121$$

Example of Backpropagation



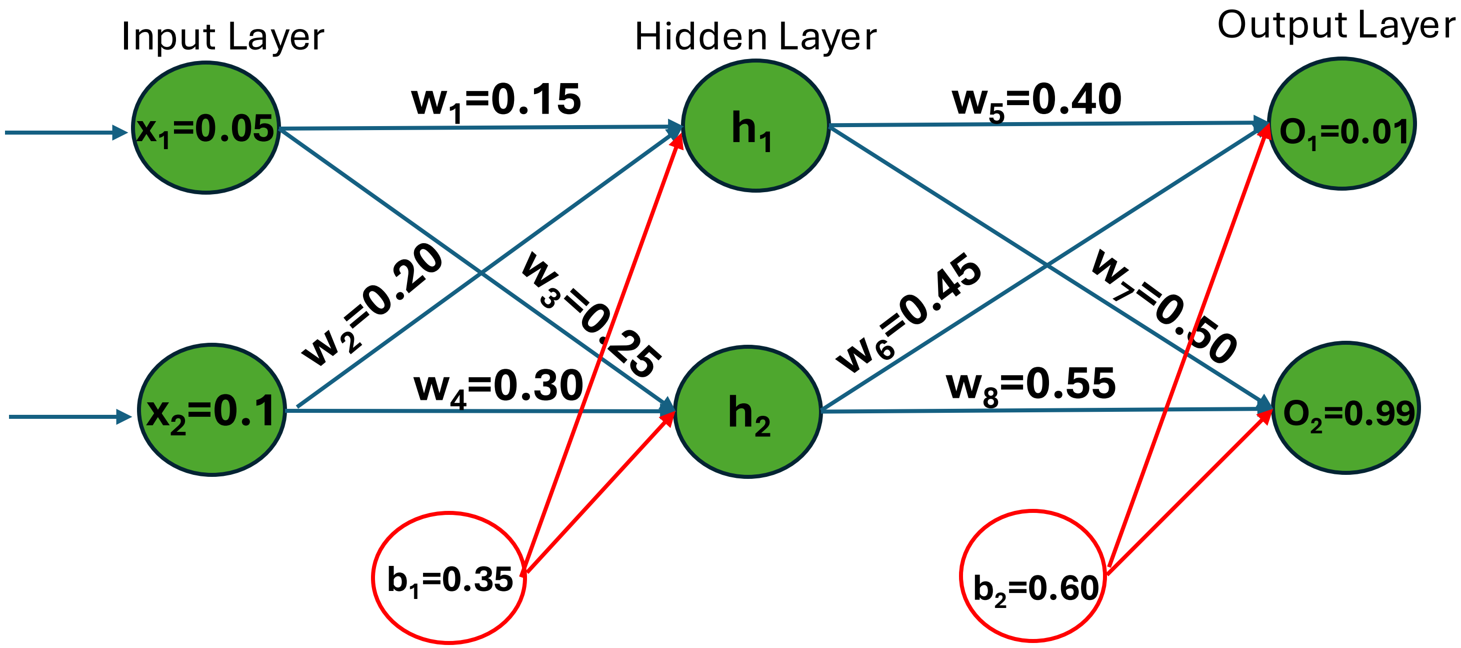
	Old value	New value		Old value	New value
w_5	0.40	0.35891648	w_7	0.50	0.511301270
w_6	0.45	0.408666186	w_8	0.55	0.561370121

Example of Backpropagation



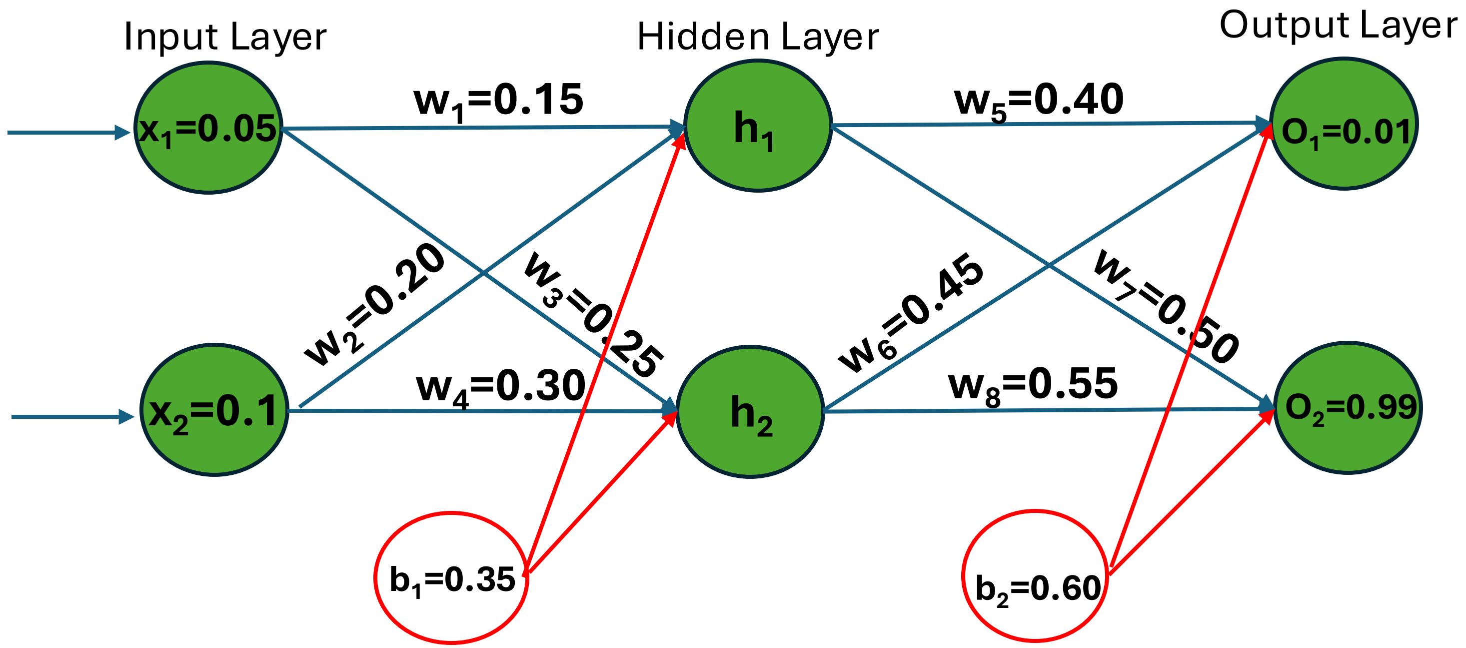
- **The Backpropagation Algorithm (Hidden Layer):**
  - Now, we continue the backwards pass by calculating new values for w_1, w_2, w_3, and w_4.
  - We use similar process as the output layer.
  - However, considering that the output of each hidden layer neuron contributes to the output (and therefore error) of multiple output neurons.
  - i.e. out_{h1} affects both out_{o1}, and out_{o2}.

Example of Backpropagation



- **The Backpropagation Algorithm (Hidden Layer):**
  - It is stated as $\frac{\partial E_{total}}{\partial w_1}$, and read as the partial derivative of E_{total} with respect to w_1.
  - As, there is no direct connection from the equation between the E_{total} and w_1, we use the chain rule.

Example of Backpropagation



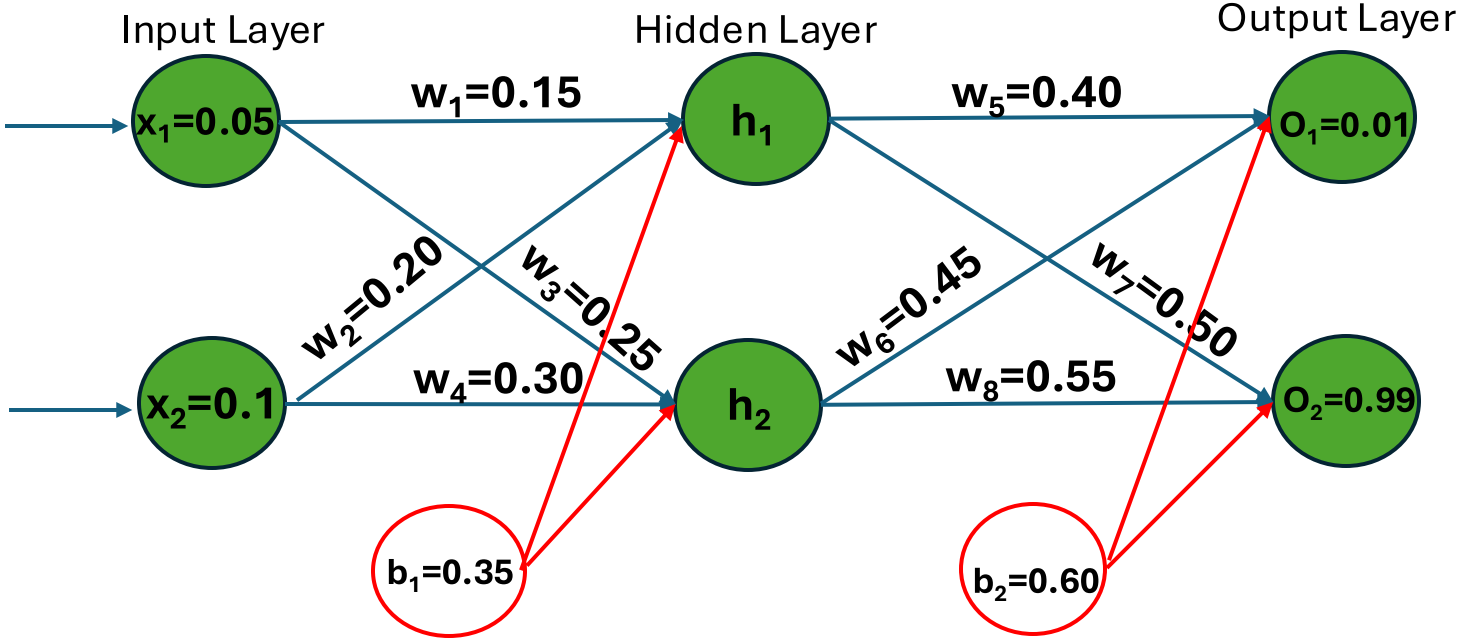
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_1} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_1}$$, where;

$$\frac{\partial Eo_1}{\partial net_{o1}} = \frac{\partial Eo_1}{\partial outo_1} * \frac{\partial outo_1}{\partial net_{o1}}$$

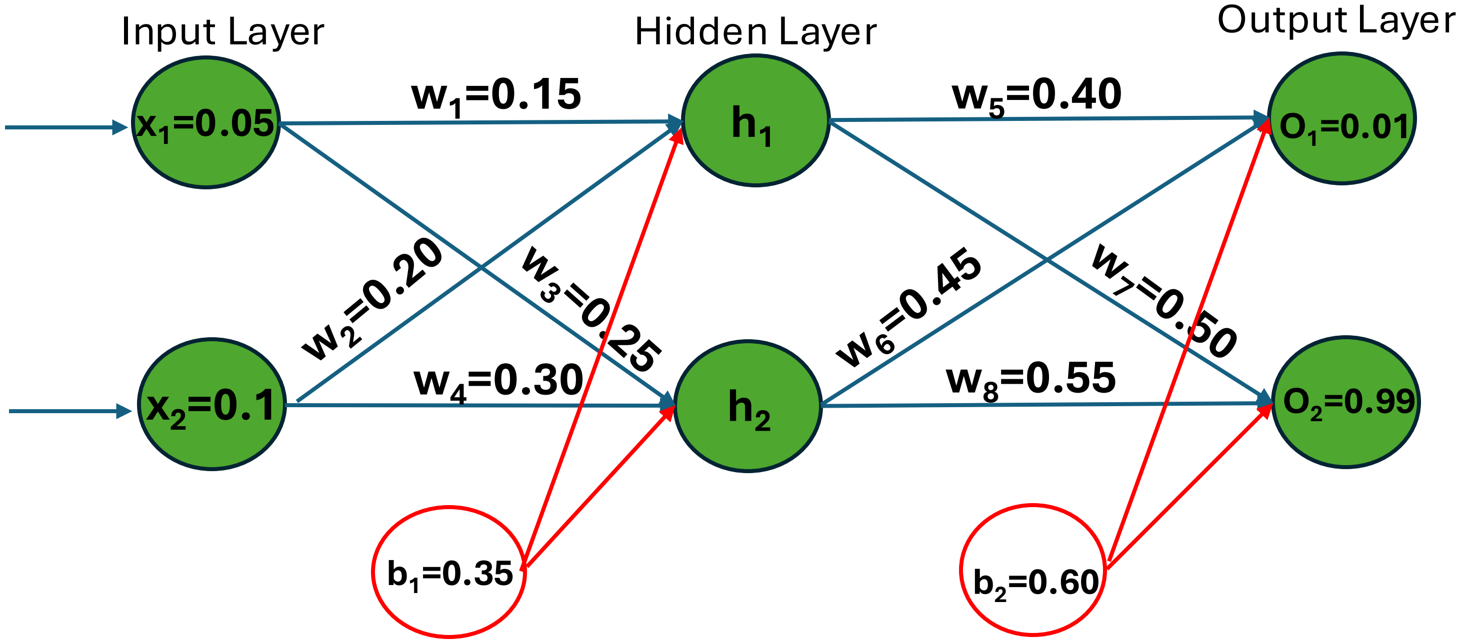
$= (out_{o1} - target_{o1}) * (out_{o1} * (1 - out_{o1}))$

$= (0.75136507 - 0.01) * (0.75136507 * (1- 0.75136507))$

$= 0.74136507 * 0.186815602$

$= 0.138498562$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

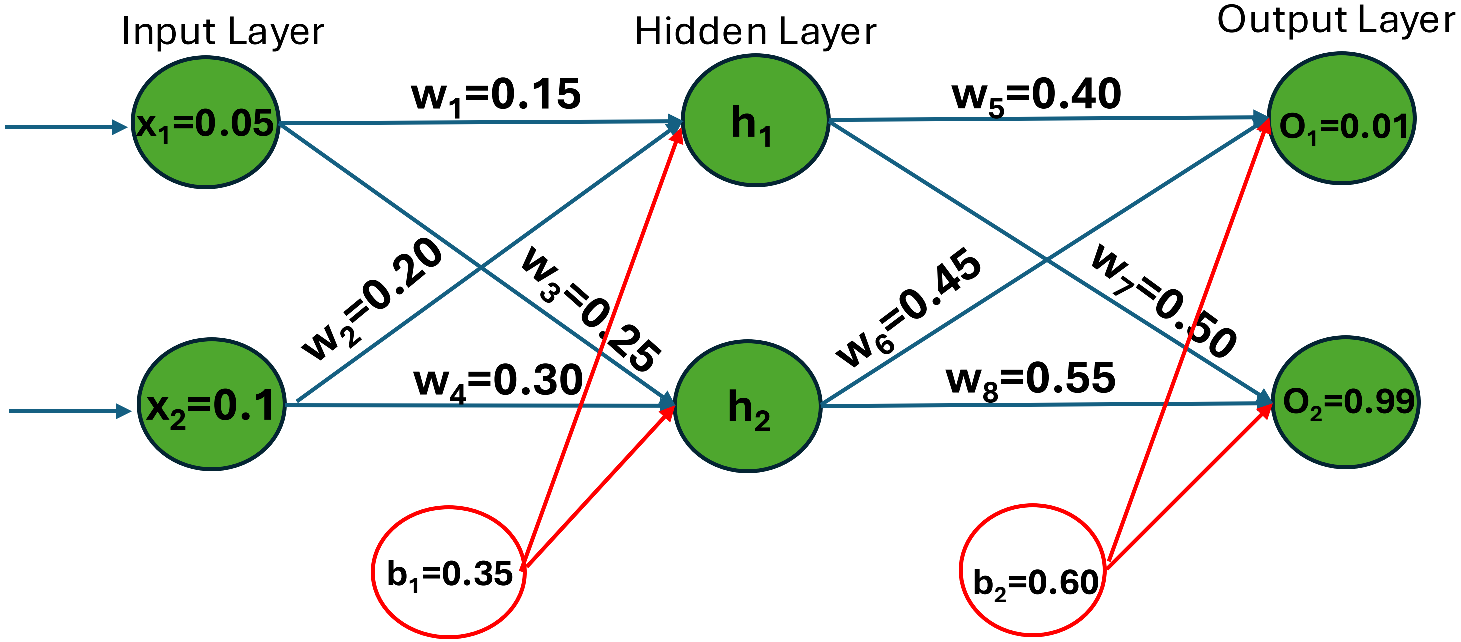
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1},$$

$$\frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation

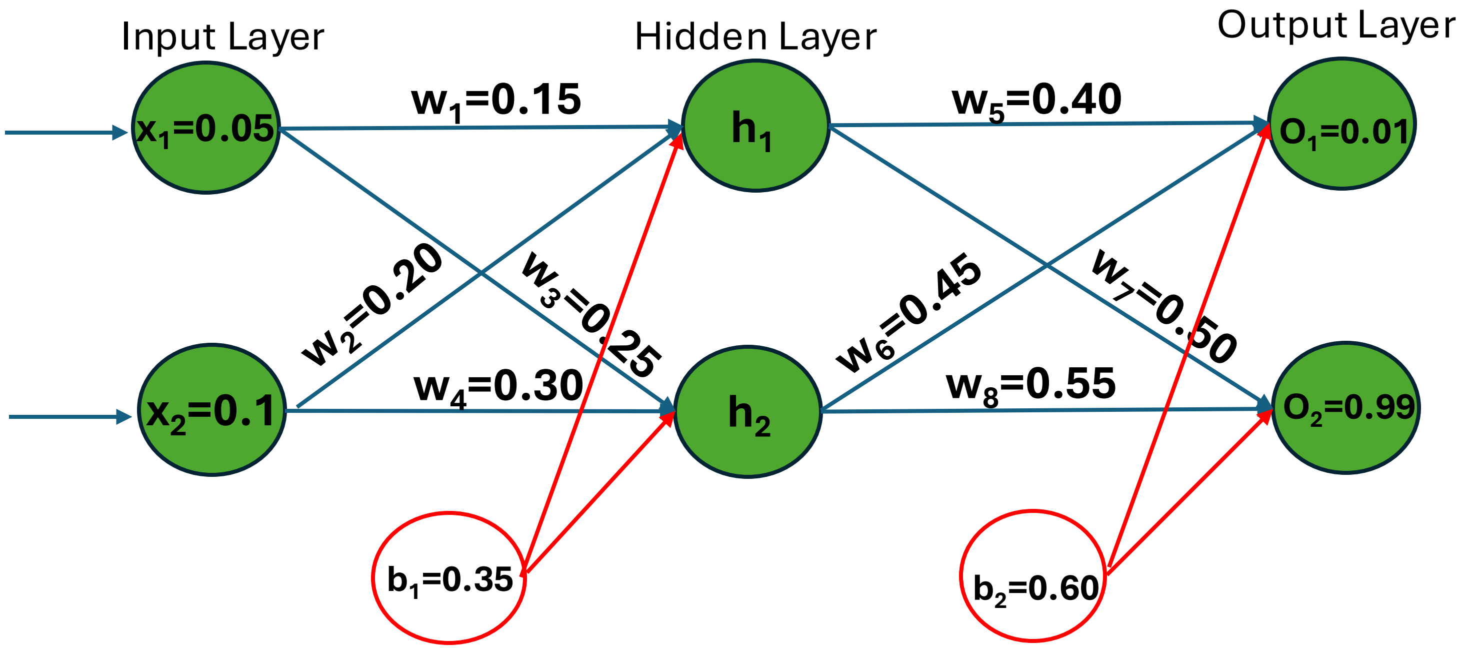


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_1} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_1}$$, where;

$$\frac{\partial neto_1}{\partial outh_1} = w_5 = 0.40$$

Example of Backpropagation



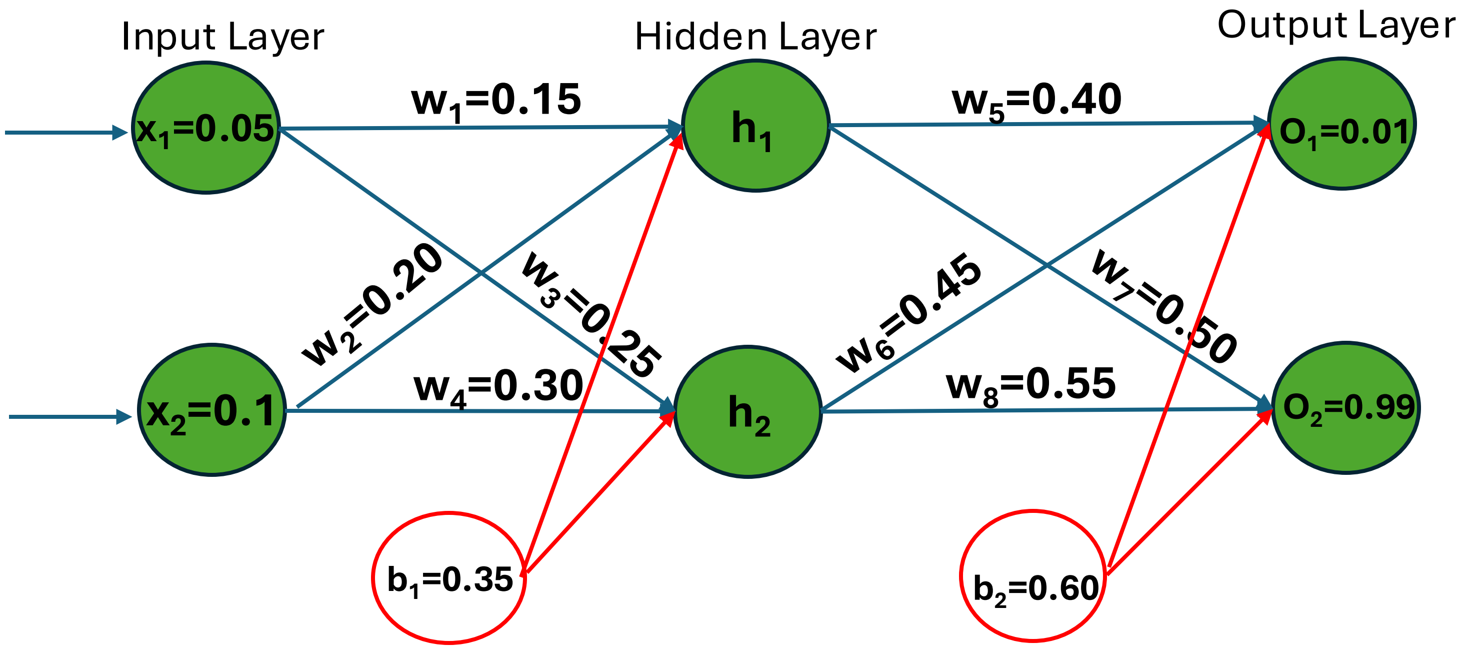
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

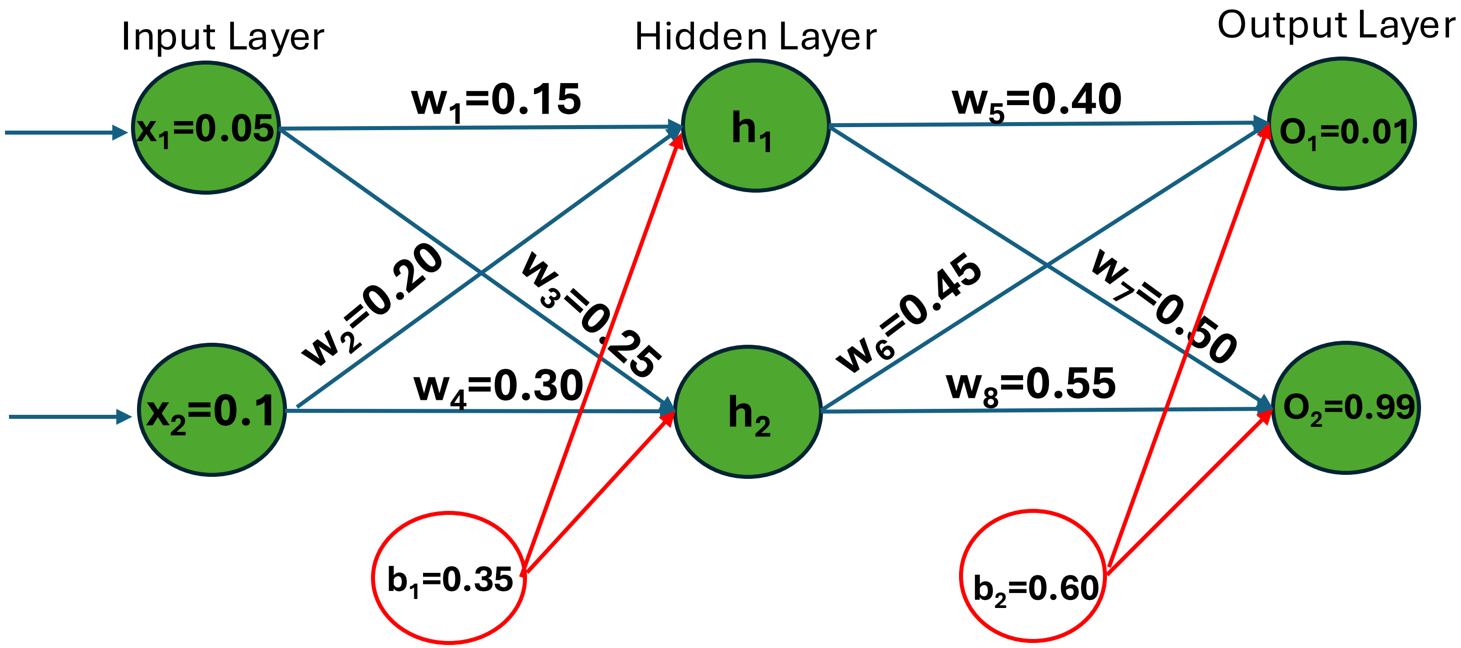
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = 0.138498562 * 0.40$$

$$= 0.055399425$$

Example of Backpropagation



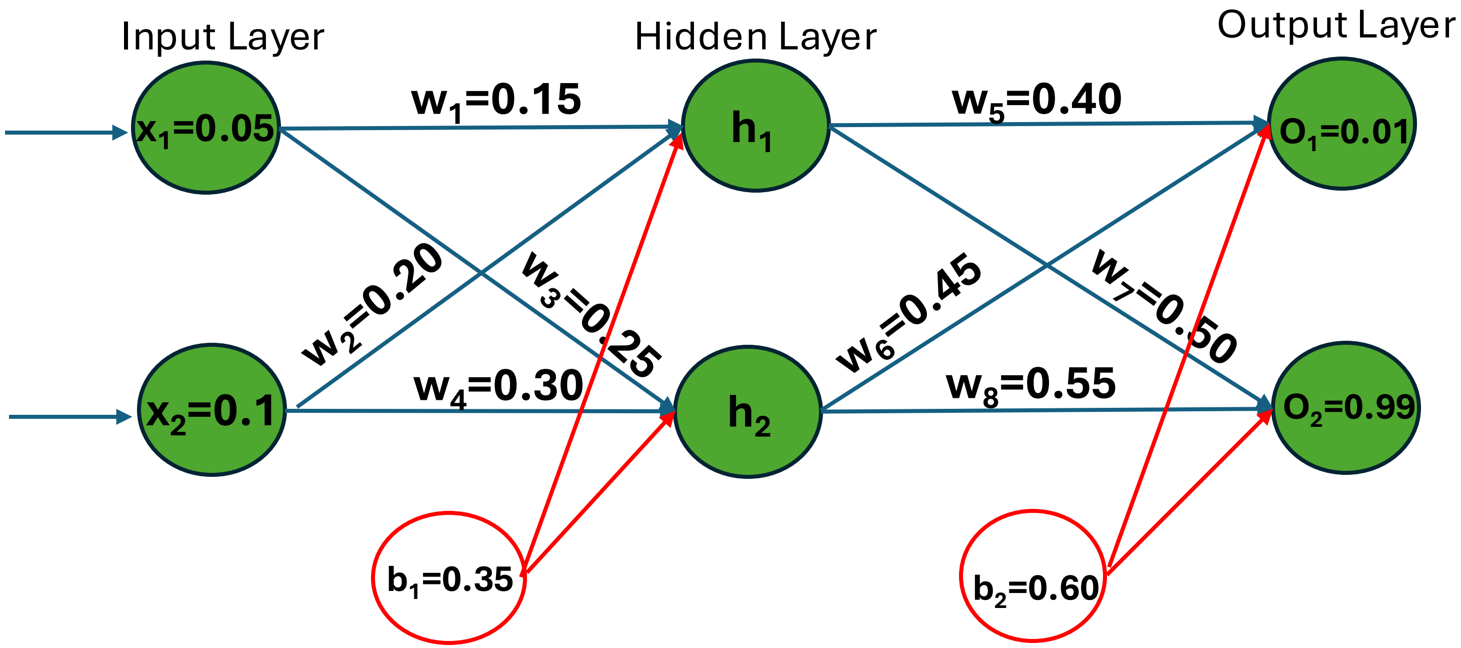
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1},\qquad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_1} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_1}$$, where;

$$\frac{\partial Eo_2}{\partial net_{o2}} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial net_{o2}}$$

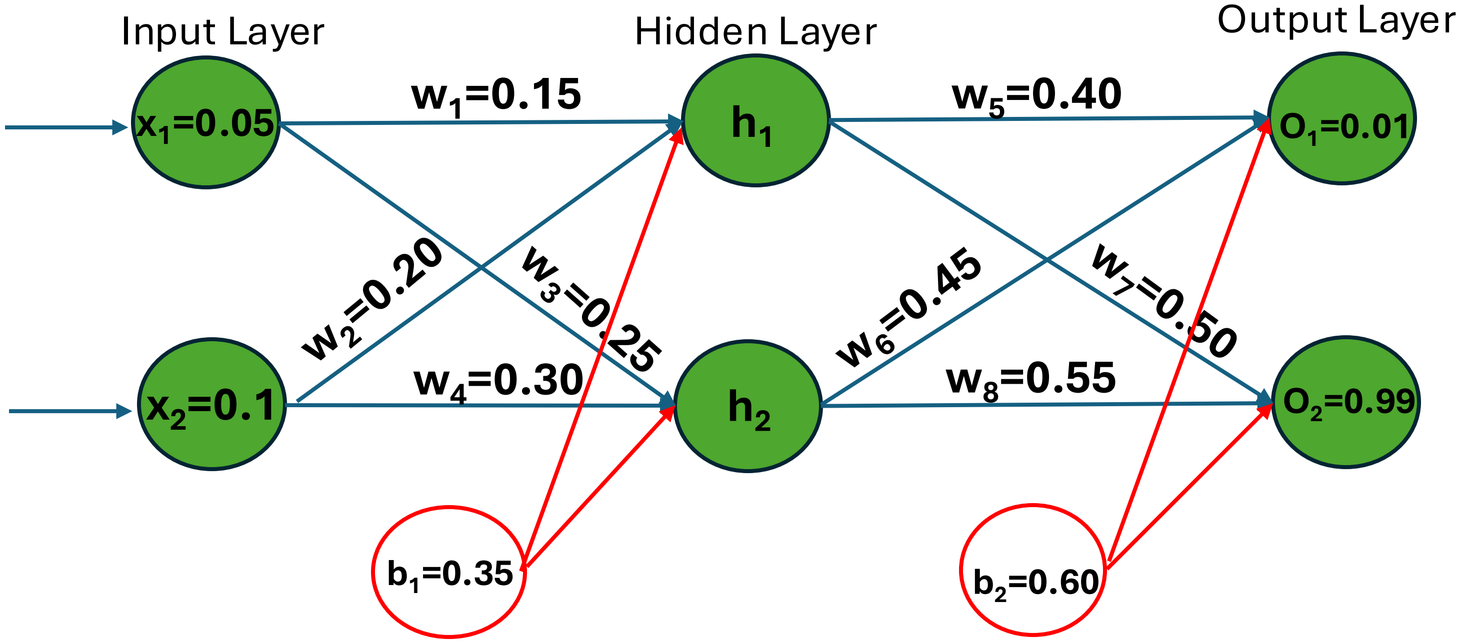
$= (out_{o2} - target_{o2}) * (out_{o2} * (1 - out_{o2}))$

$= (0.772928465 - 0.99) * (0.772928465 * (1- 0.772928465))$

$= -0.217071535 * 0.17551005299$

$= -0.03809823661$

Example of Backpropagation



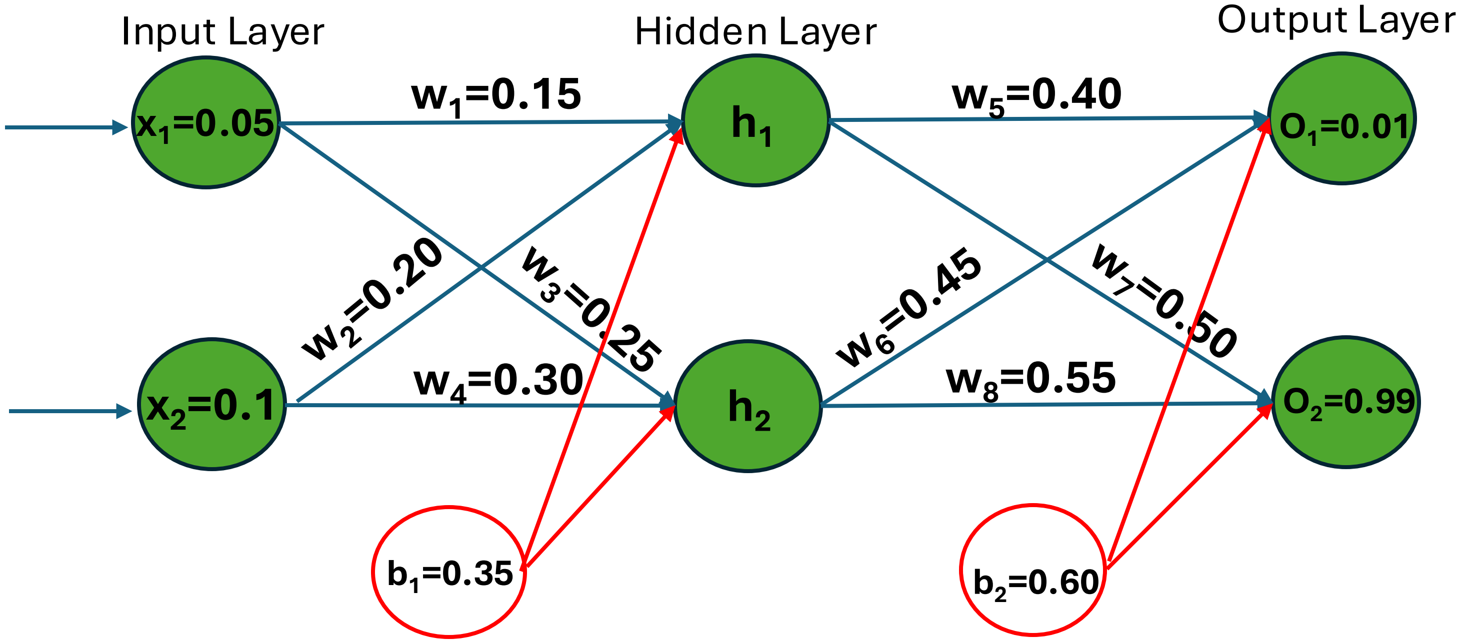
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation

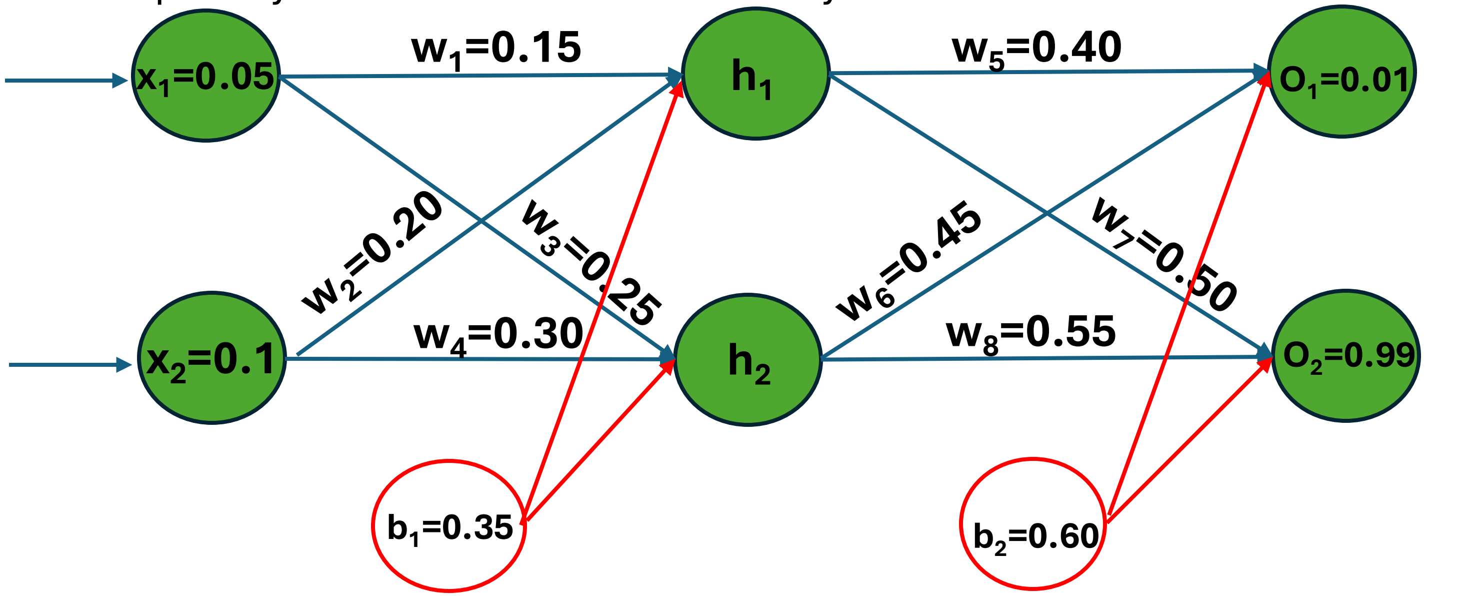


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_1} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_1}$$, where;

$\frac{\partial neto_2}{\partial outh_1} = w_7 = 0.50$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

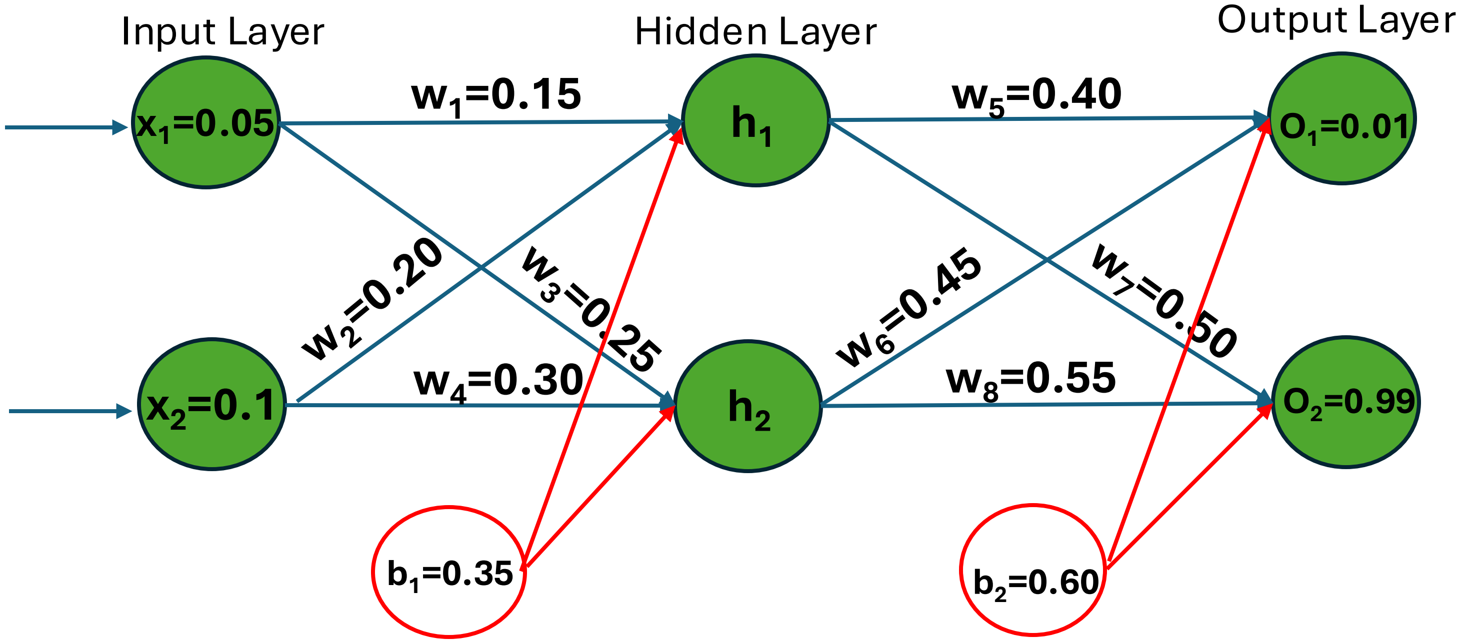
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial_{Etotal}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

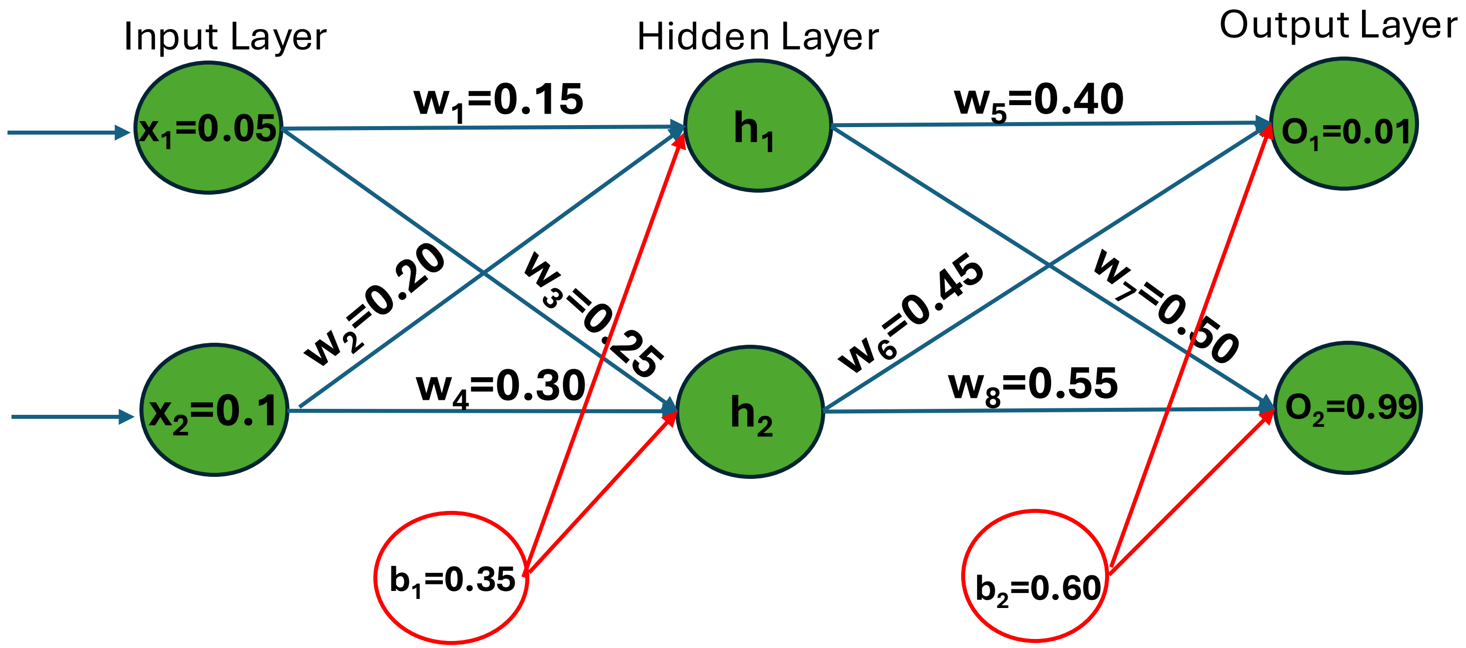
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o2}}{\partial out_{h1}} = -0.03809823661 * 0.50$$

$$= -0.019049119$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

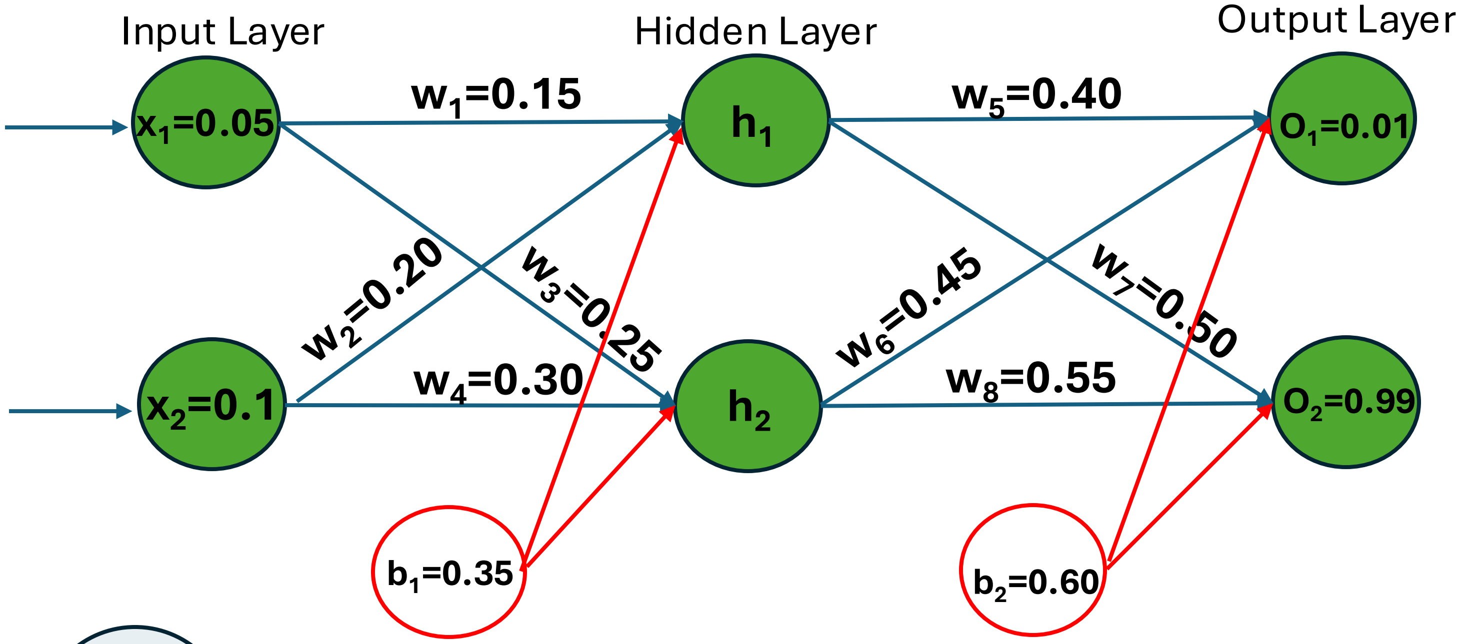
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

1) $\frac{\partial_{Etotal}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

= 0.055399425 + (-0.019049119)

= 0.036350306

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

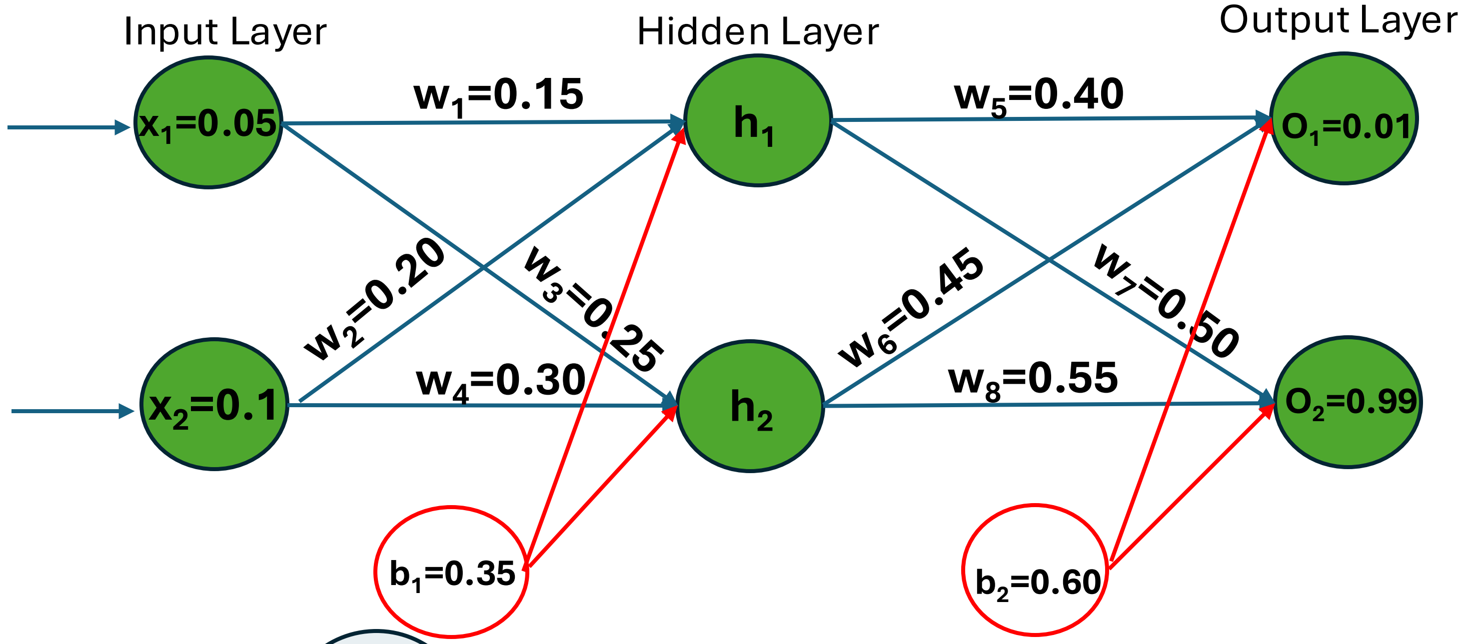
$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

2) $\frac{\partial outh_1}{\partial net_{h1}} = out_{h1} * (1 - out_{h1})$

$= 0.593269992 * (1 - 0.593269992)$

$= 0.241300709$

Example of Backpropagation

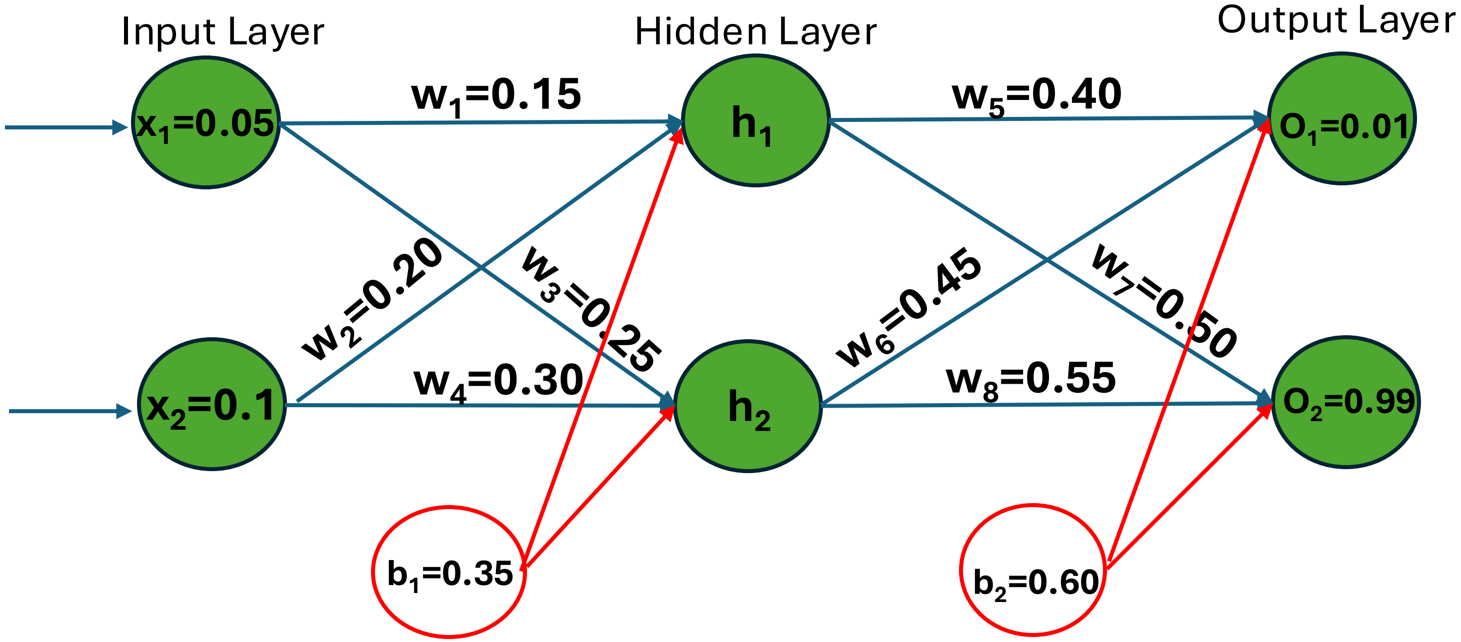


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

2) $\frac{\partial net_{h1}}{\partial w_1} = x_1 = 0.05$

Example of Backpropagation



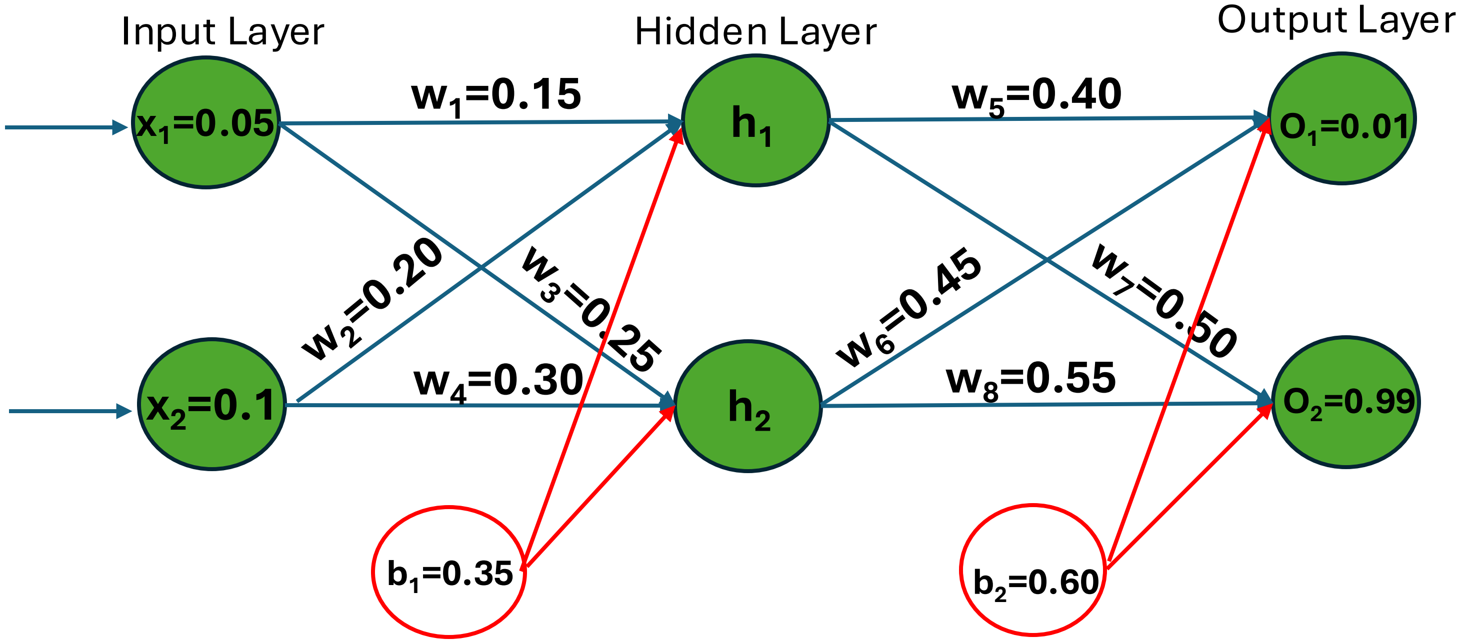
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1};$$

= 0.036350306 * 0.241300709 * 0.05

= 0.00043856773

Example of Backpropagation



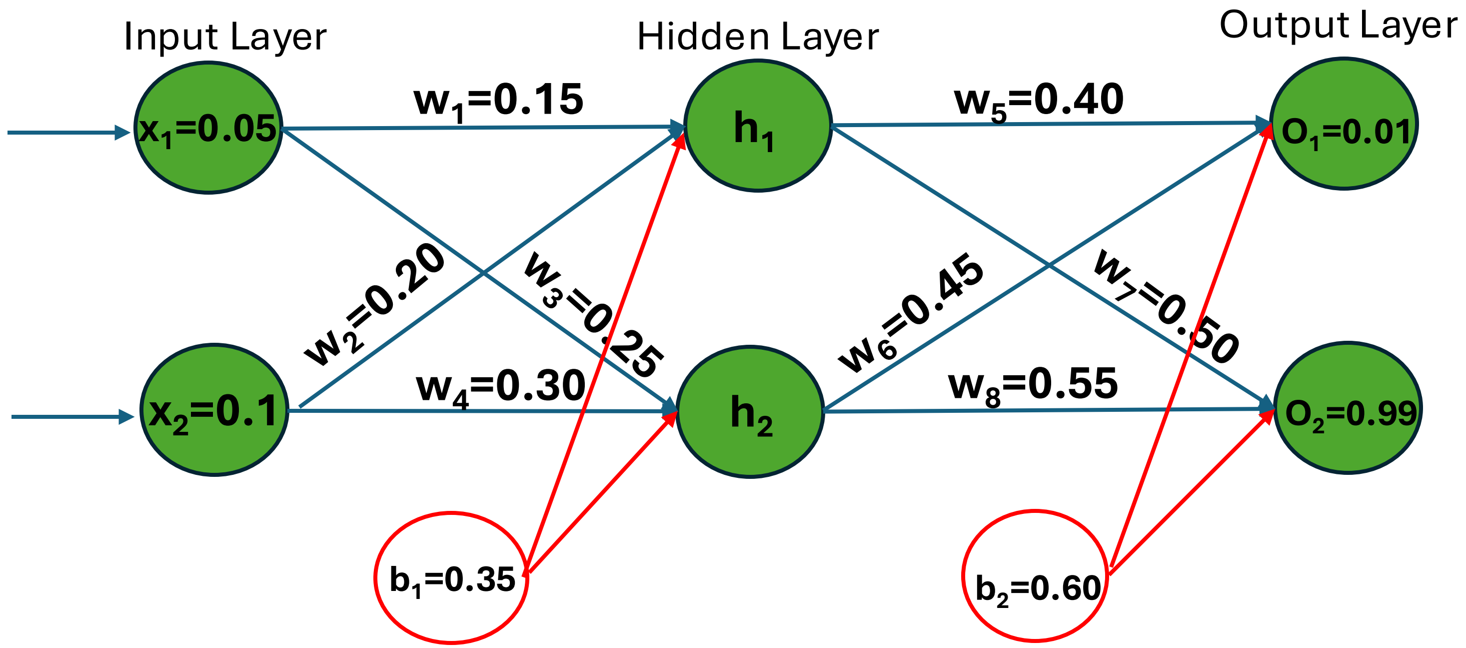
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$w_1^+ = w_1 - \eta * \frac{\partial Etotal}{\partial w_1}$$

$$= 0.15 - 0.5 * 0.00043856773$$

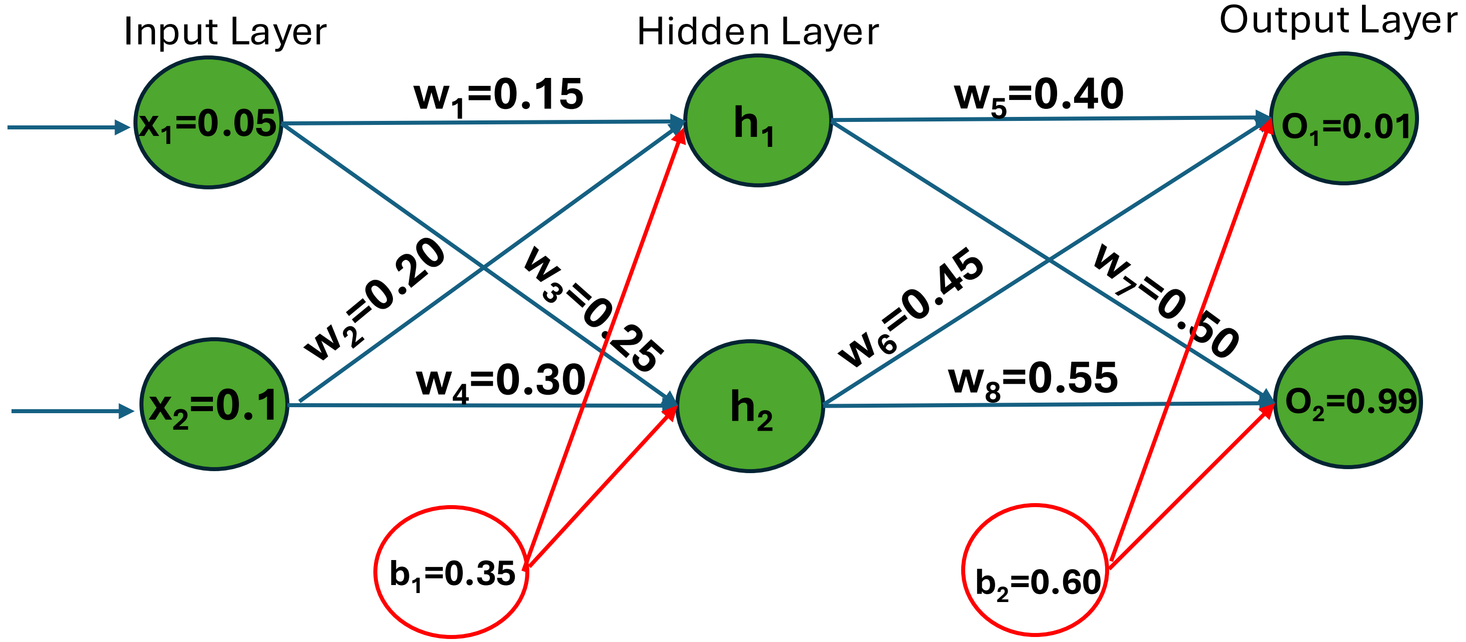
$$= 0.149780716$$

Example of Backpropagation



- **The Backpropagation Algorithm (Hidden Layer):**
  - After finding the new value of w_1, the value of w_2 is calculated as follow:
  - It is stated as $\frac{\partial E_{total}}{\partial w_2}$, and read as the partial derivative of E_{total} with respect to w_2.
  - As, there is no direct connection from the equation between the E_{total} and w_2, we use the chain rule.

Example of Backpropagation



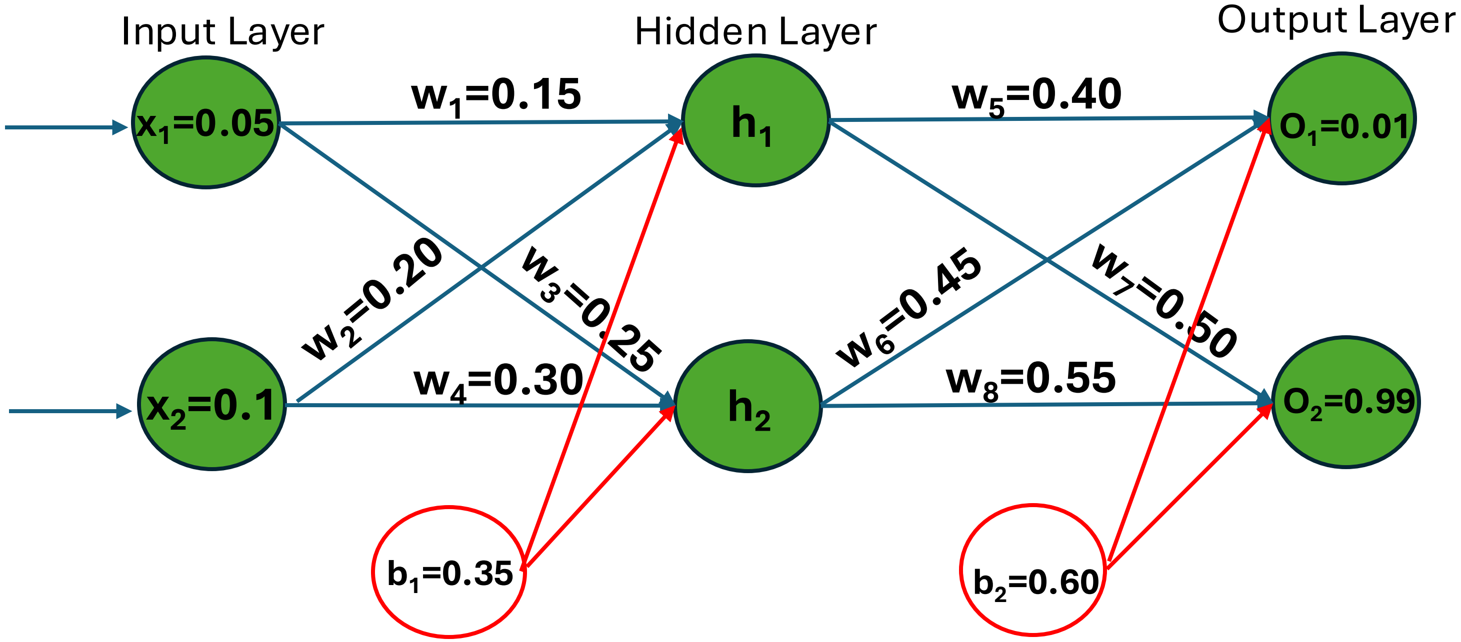
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial E_{o2}}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_2} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_2}$$, where;

$$\frac{\partial Eo_1}{\partial net_{o1}} = \frac{\partial Eo_1}{\partial outo_1} * \frac{\partial outo_1}{\partial net_{o1}}$$

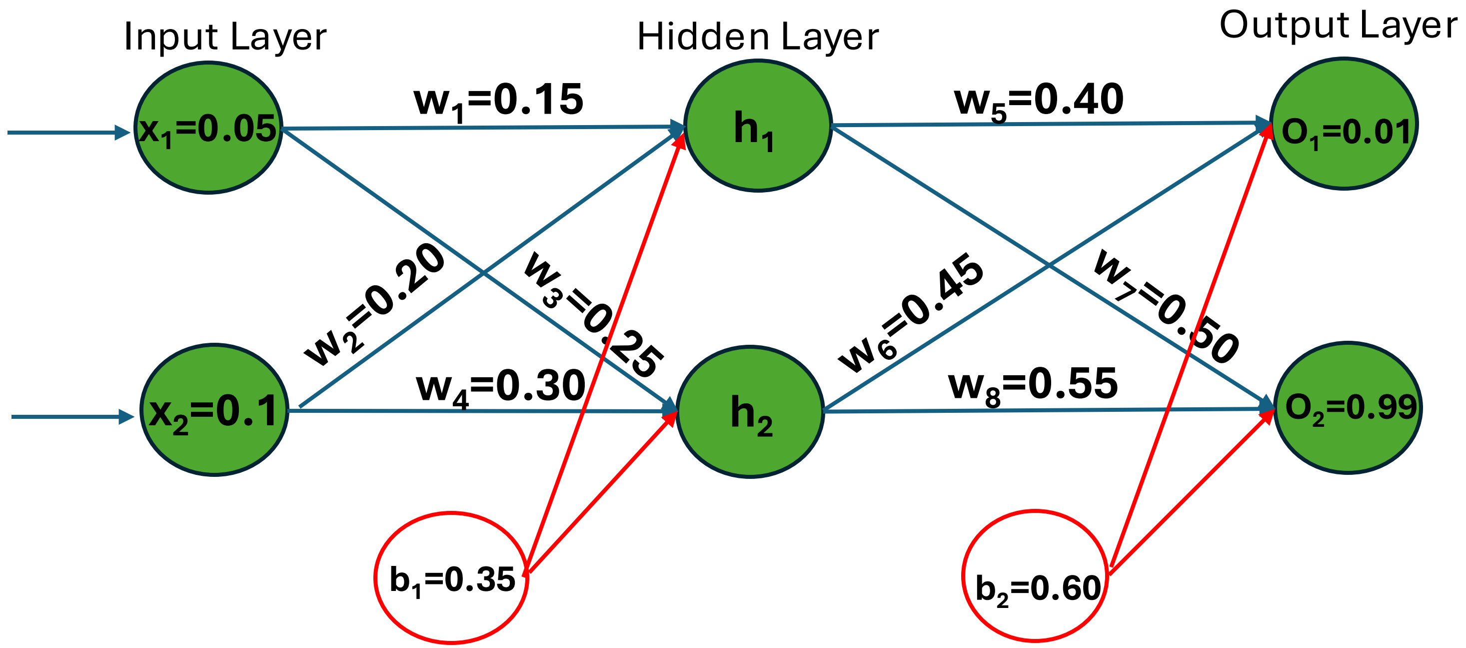
$= (out_{o1} - target_{o1}) * (out_{o1} * (1 - out_{o1}))$

$= (0.75136507 – 0.01) * (0.75136507 * (1- 0.75136507))$

$= 0.74136507 * 0.186815602$

$= 0.138498562$

Example of Backpropagation



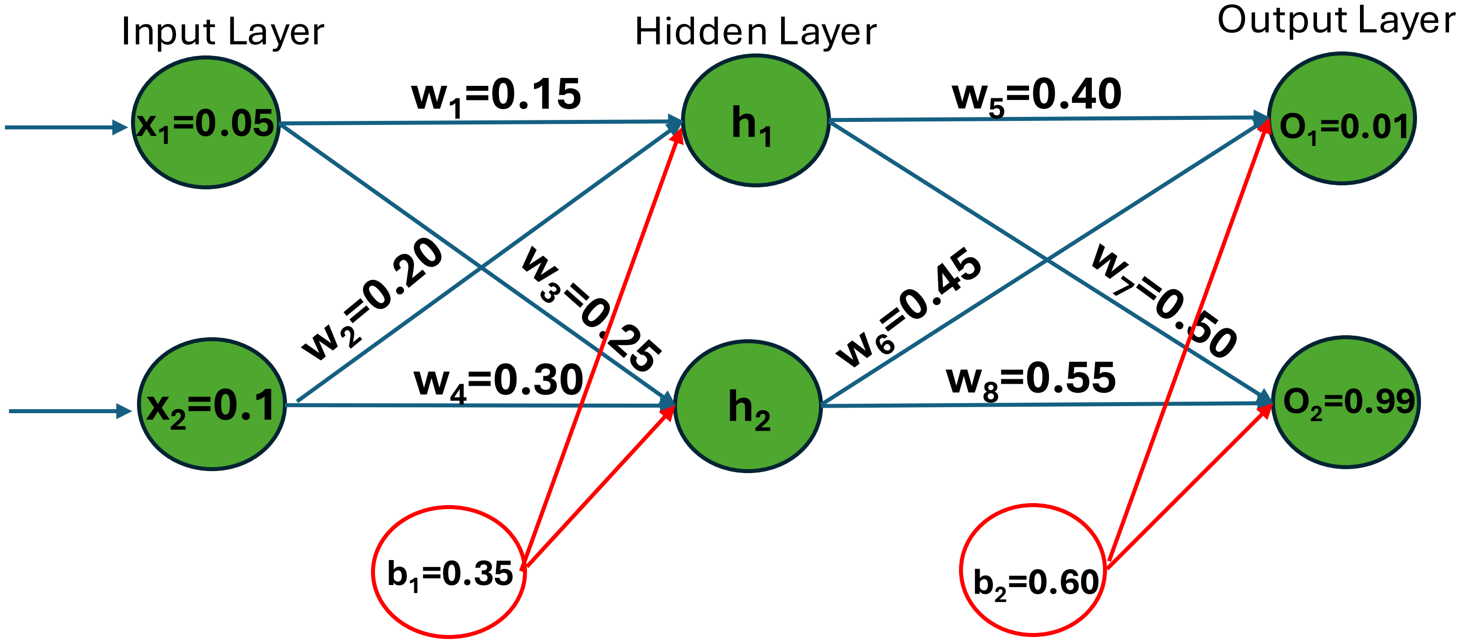
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation

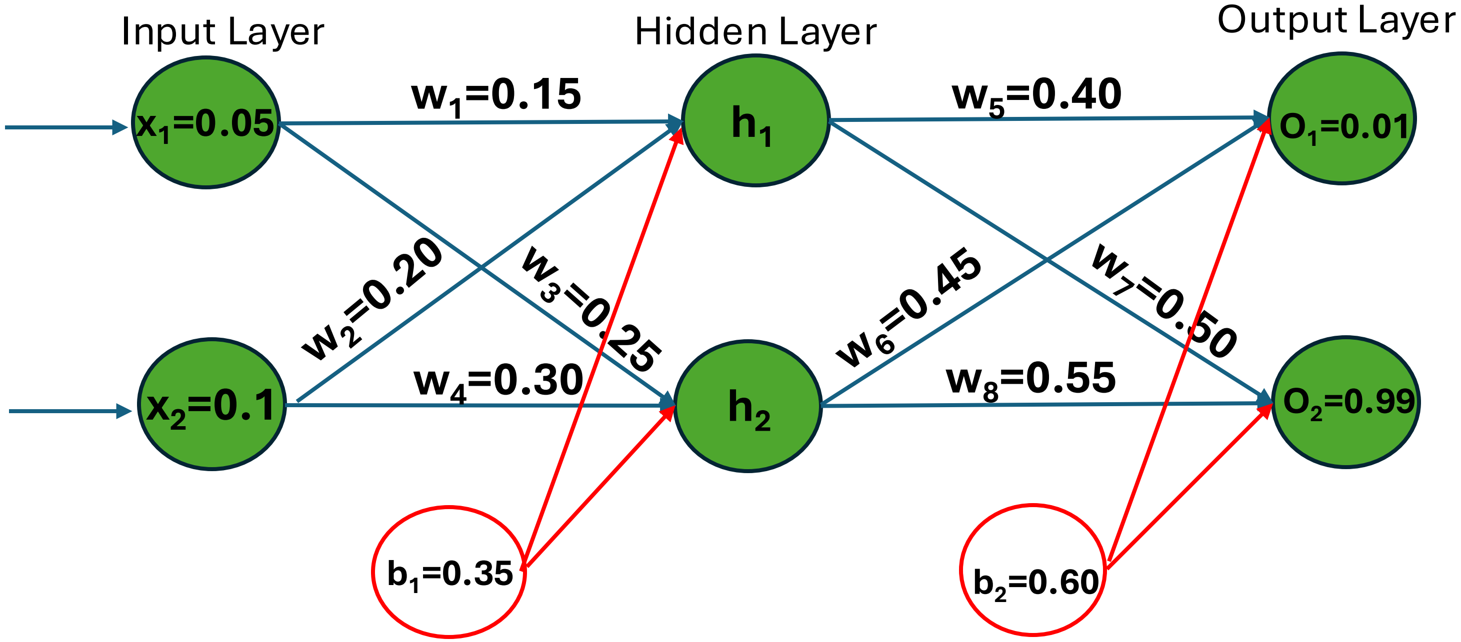


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_2} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_2}$$, where;

$\frac{\partial neto_1}{\partial outh_1} = w_5 = 0.40$

Example of Backpropagation



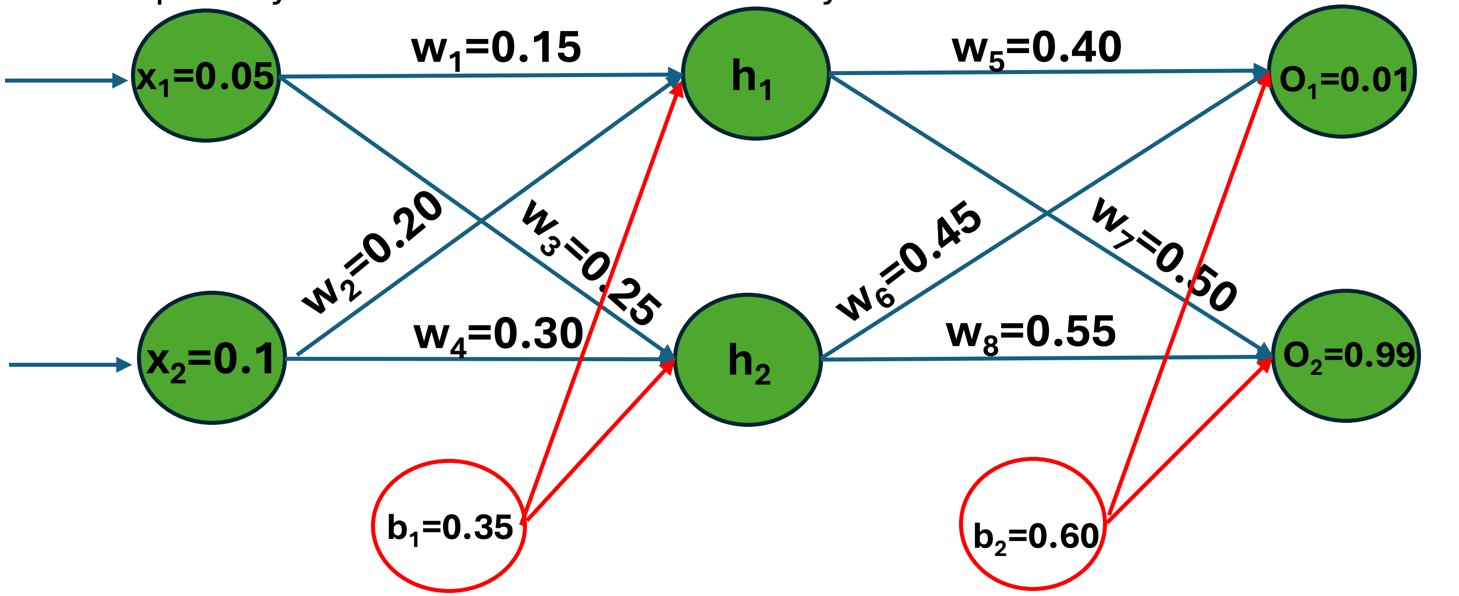
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

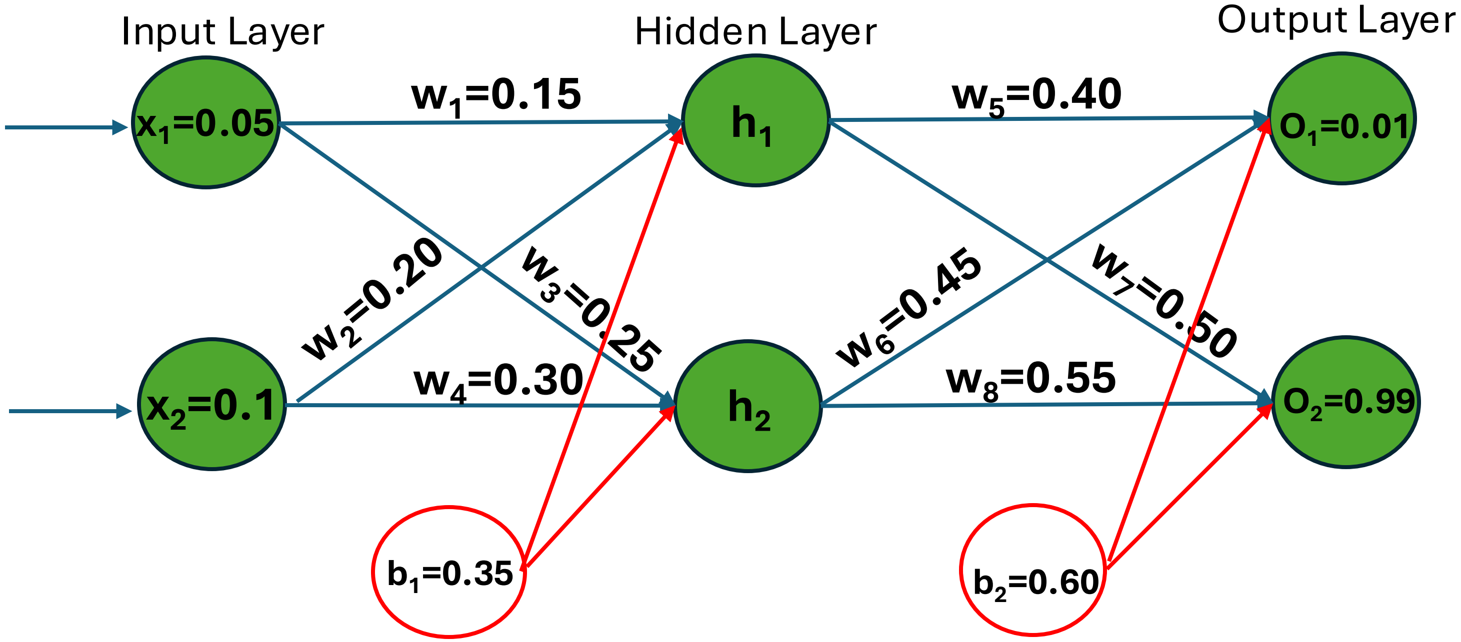
$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = 0.138498562 * 0.40$$

$$= 0.055399425$$

Example of Backpropagation



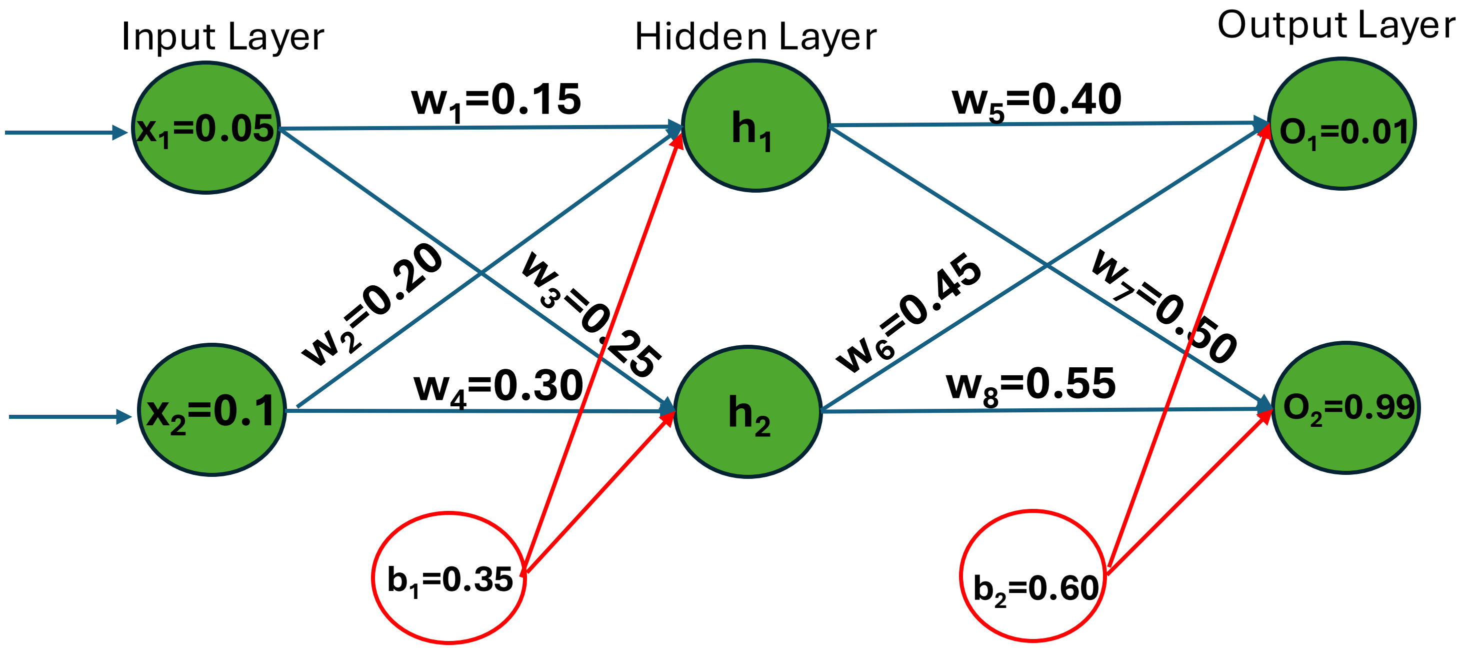
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_2} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_2}$$, where;

$$\frac{\partial Eo_2}{\partial net_{o2}} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial net_{o2}}$$

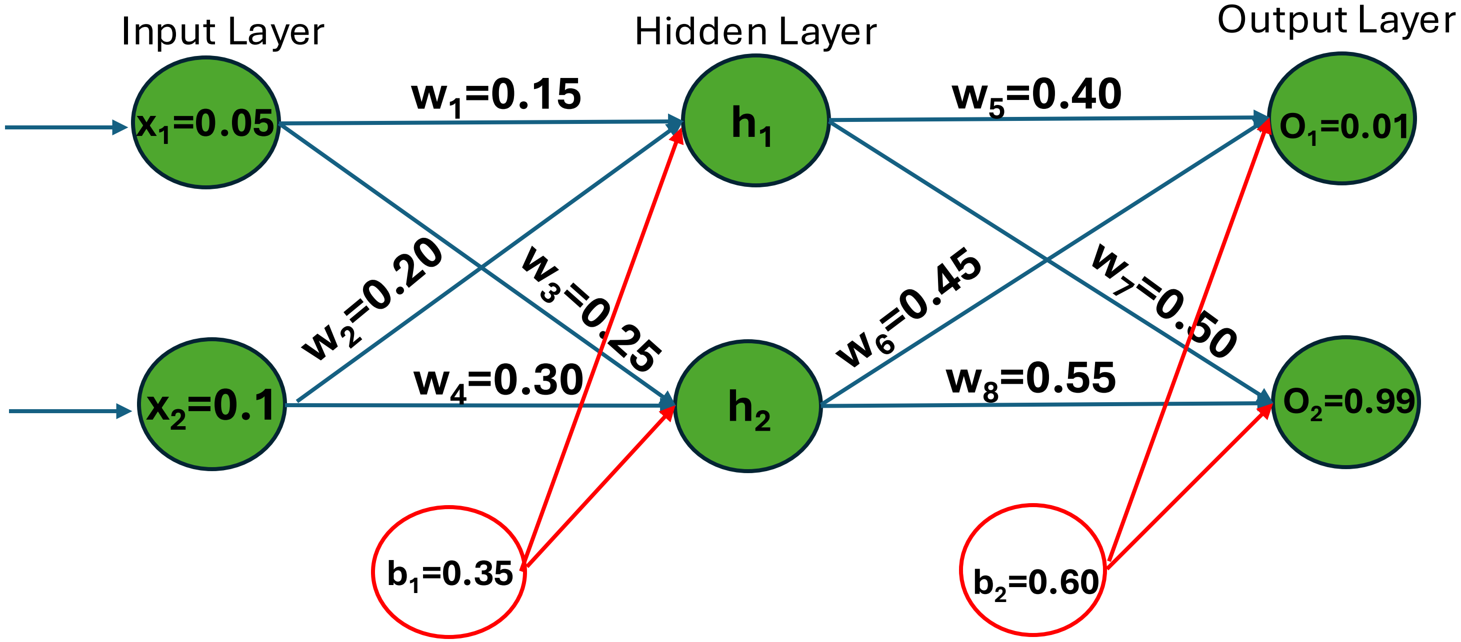
$= (out_{o2} - target_{o2}) * (out_{o2} * (1 - out_{o2}))$

$= (0.772928465 - 0.99) * (0.772928465 * (1- 0.772928465))$

$= -0.217071535 * 0.17551005299$

$= -0.03809823661$

Example of Backpropagation



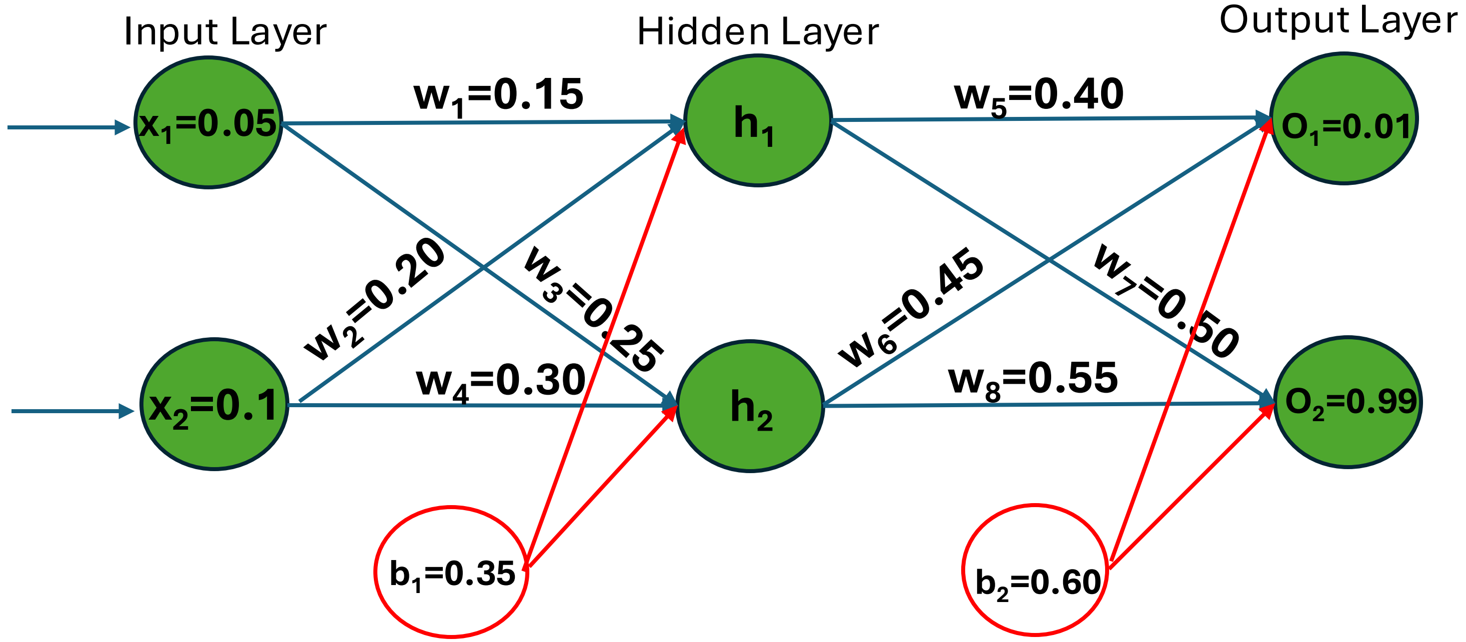
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation

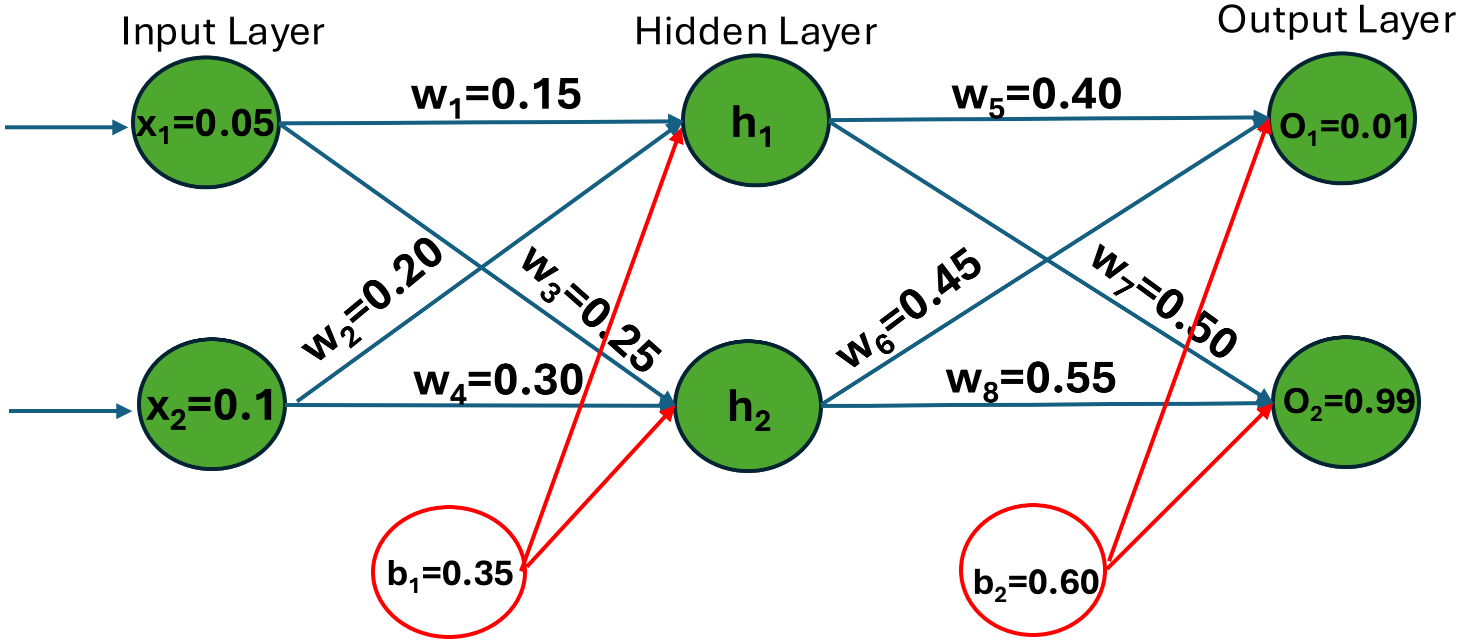


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_2} = \frac{\partial Etotal}{\partial outh_1} * \frac{\partial outh_1}{\partial neth_1} * \frac{\partial neth_1}{\partial w_2}$$, where;

$\frac{\partial neto_2}{\partial outh_1} = w_7 = 0.50$

Example of Backpropagation



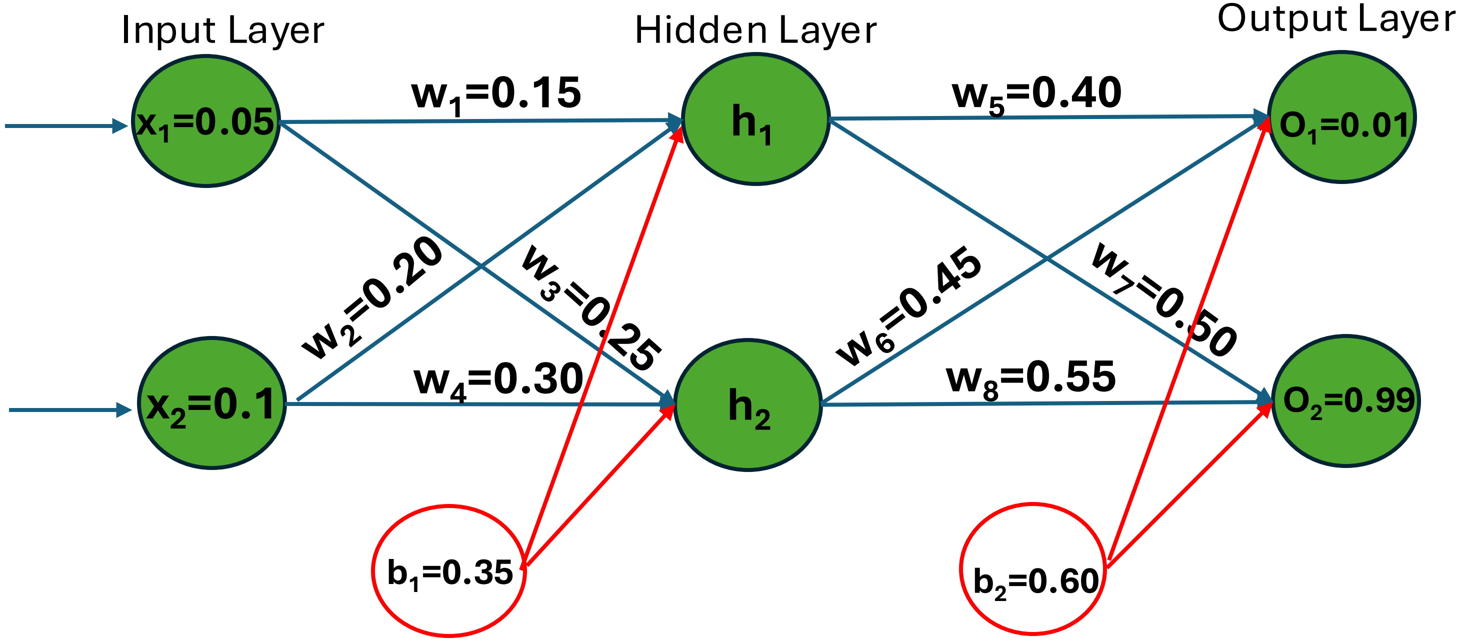
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_1}, \quad \frac{\partial Eo_2}{\partial outh_1} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_1}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

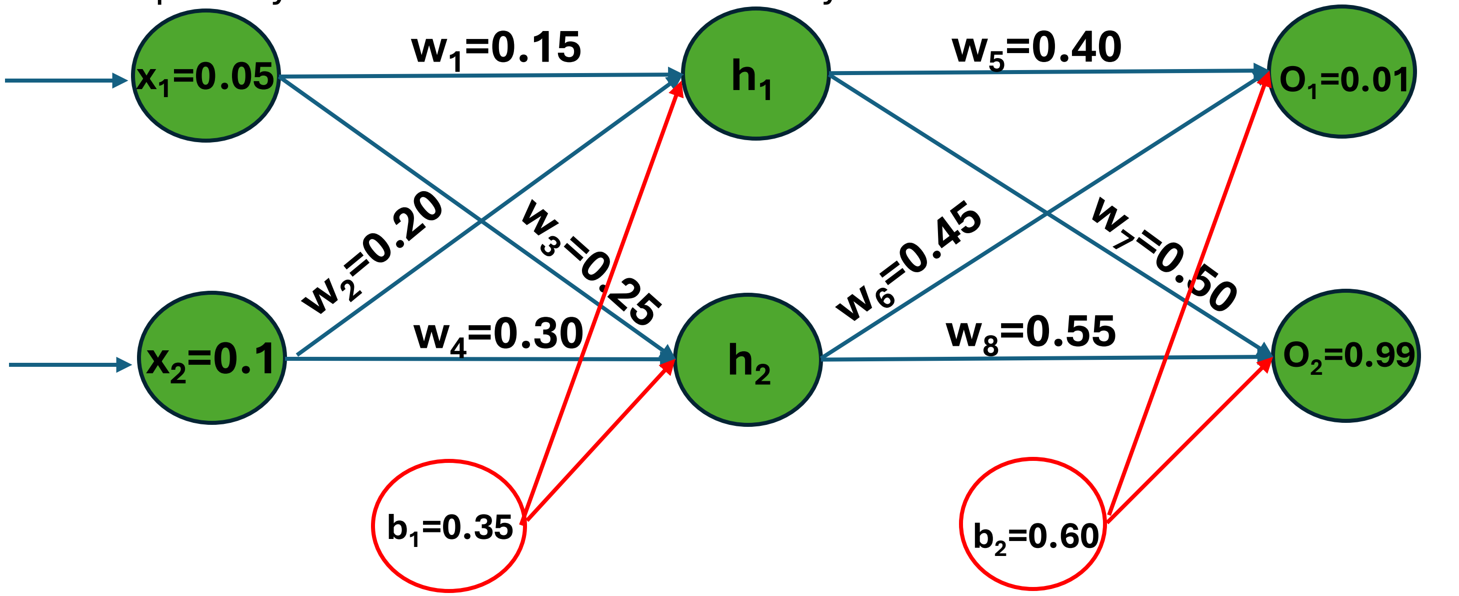
$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial Etotal}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o2}}{\partial out_{h1}} = -0.03809823661 * 0.50$$

$$= -0.019049119$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

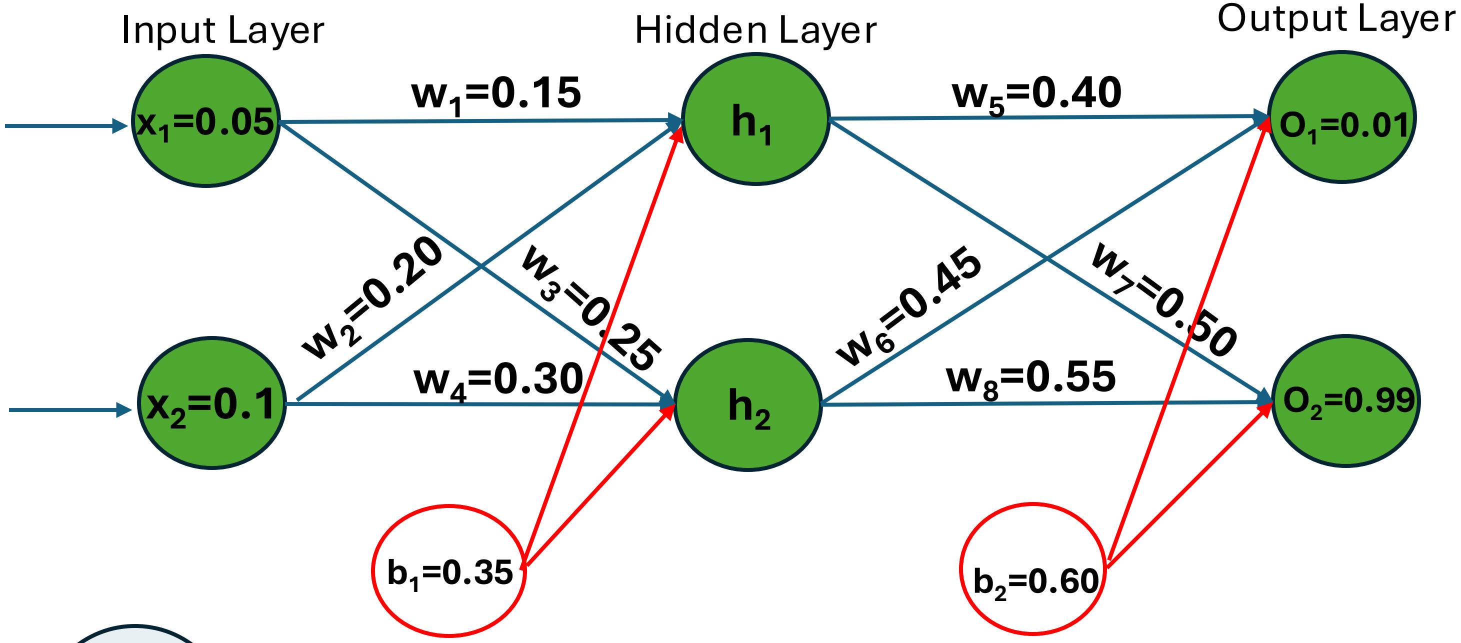
$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_1 has direct affect on both o_1 and o_2.

= 0.055399425 + (-0.019049119)

= 0.036350306

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

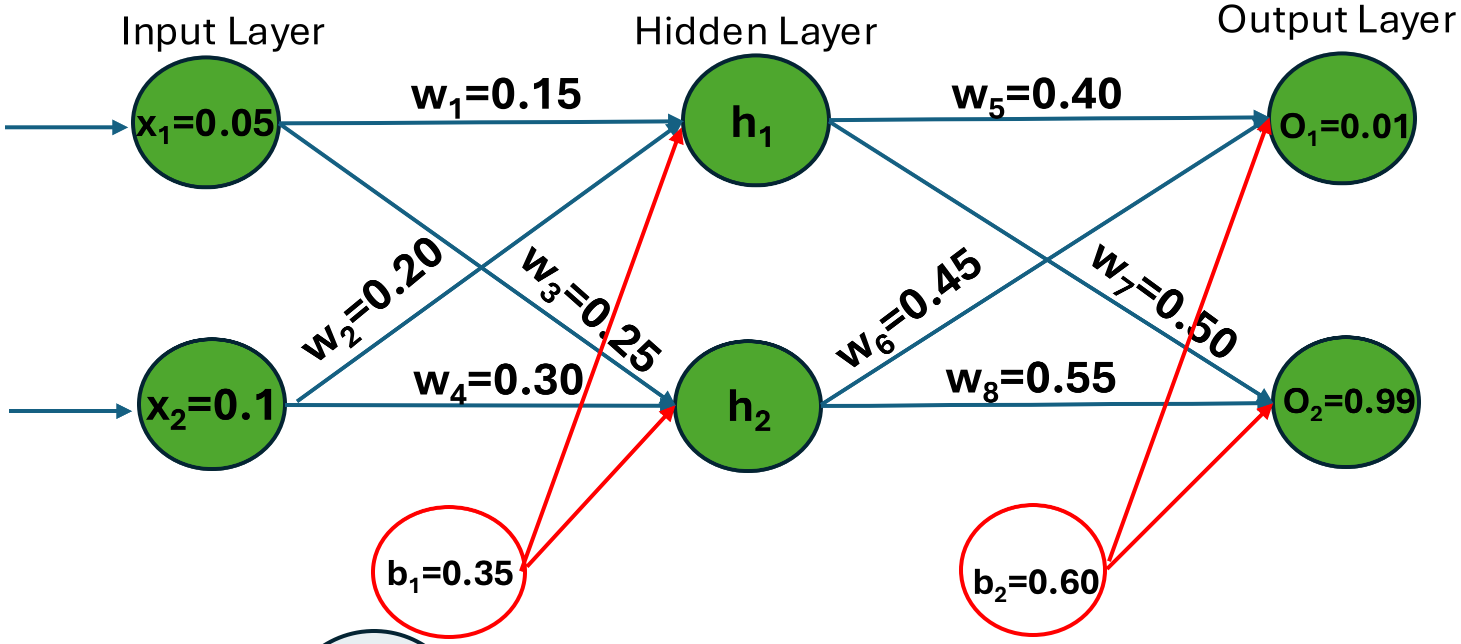
$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

2) $\frac{\partial outh_1}{\partial net_{h1}} = out_{h1} * (1 - out_{h1})$

= 0.593269992 * (1 - 0.593269992)

= 0.241300709

Example of Backpropagation

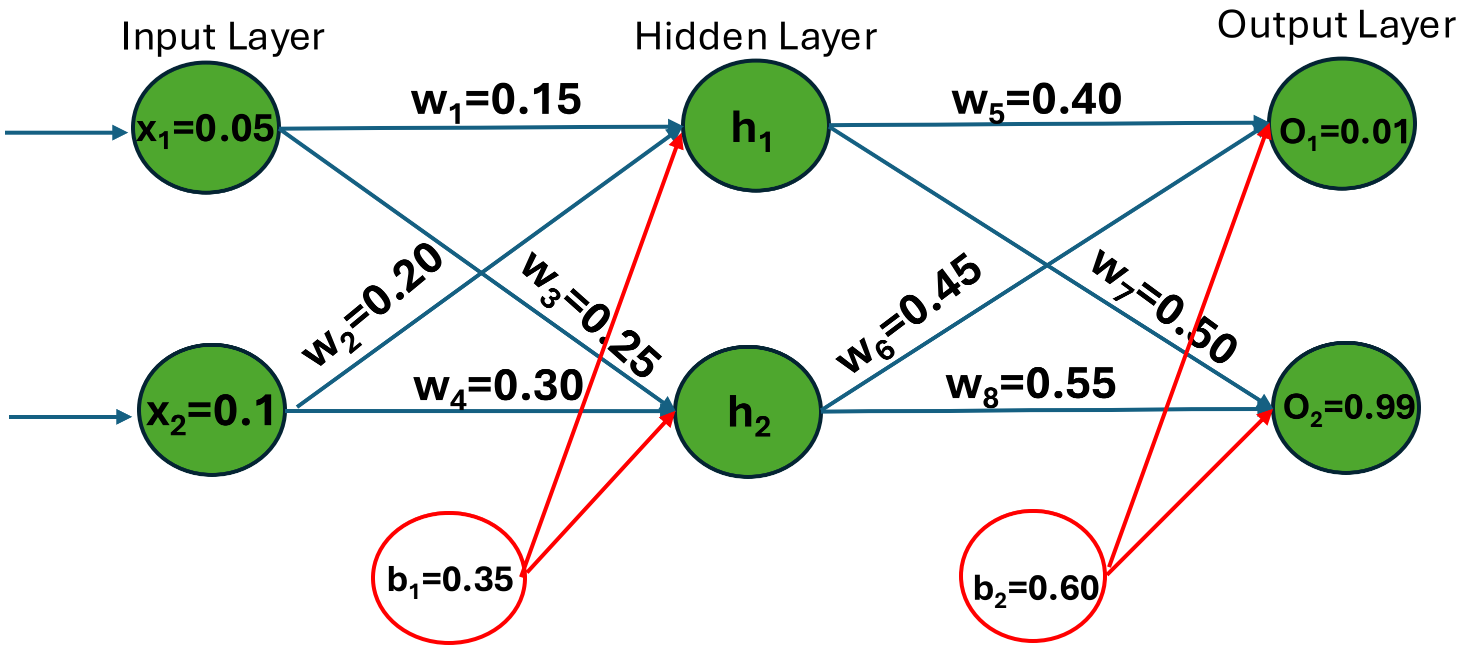


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

2) $\frac{\partial net_{h1}}{\partial w_2} = x_2 = 0.1$

Example of Backpropagation



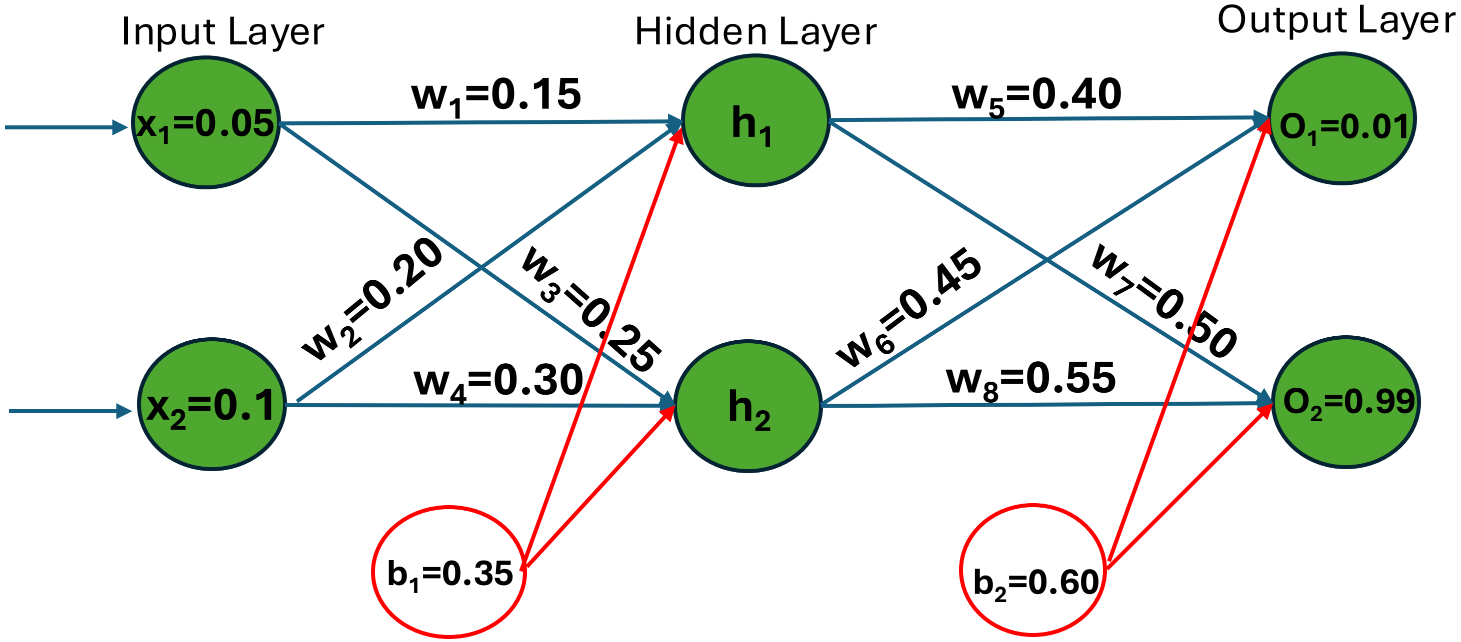
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial_{Etotal}}{\partial out_{h1}} * \frac{\partial outh_1}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_2};$$

= 0.036350306 * 0.241300709 * 0.1

= 0.00087713546

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

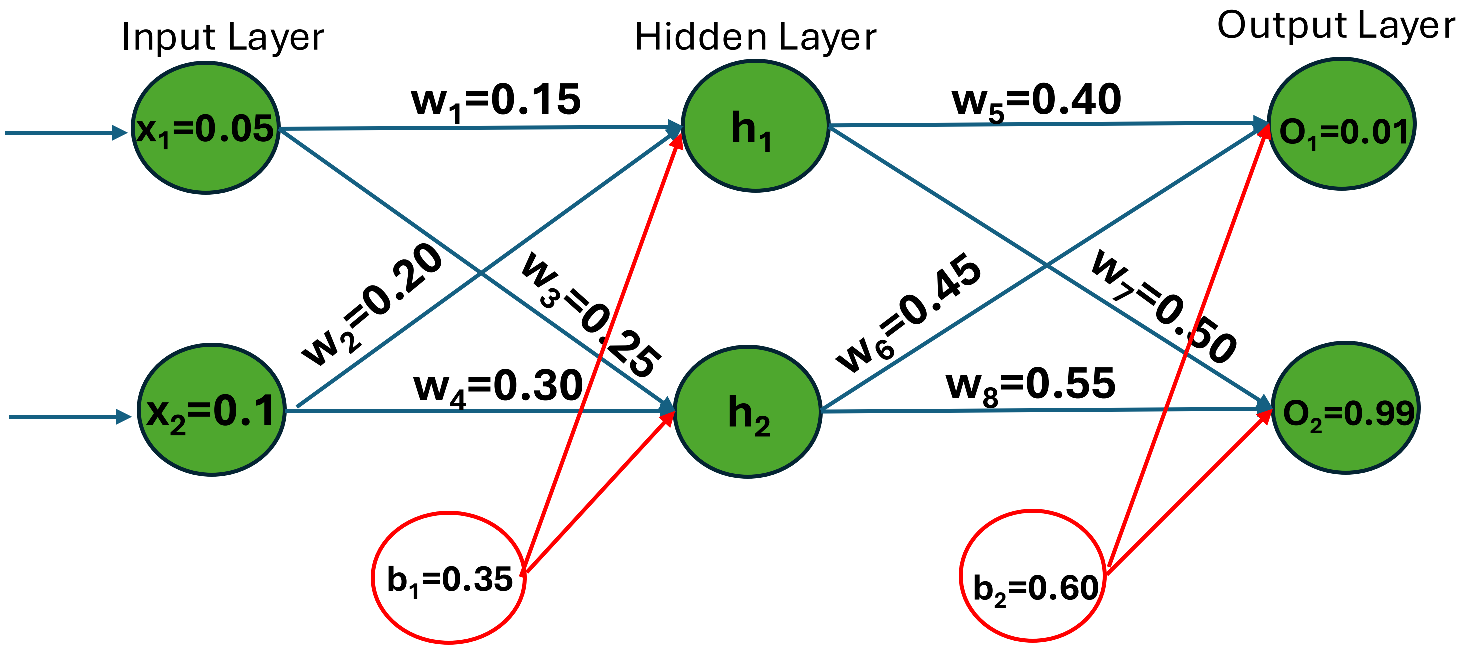
$$w_2^{+} = w_2 - \eta * \frac{\partial Etotal}{\partial w_2}$$

$$= 0.20 - 0.5 * 0.00087713546$$

$$= 0.19956143$$

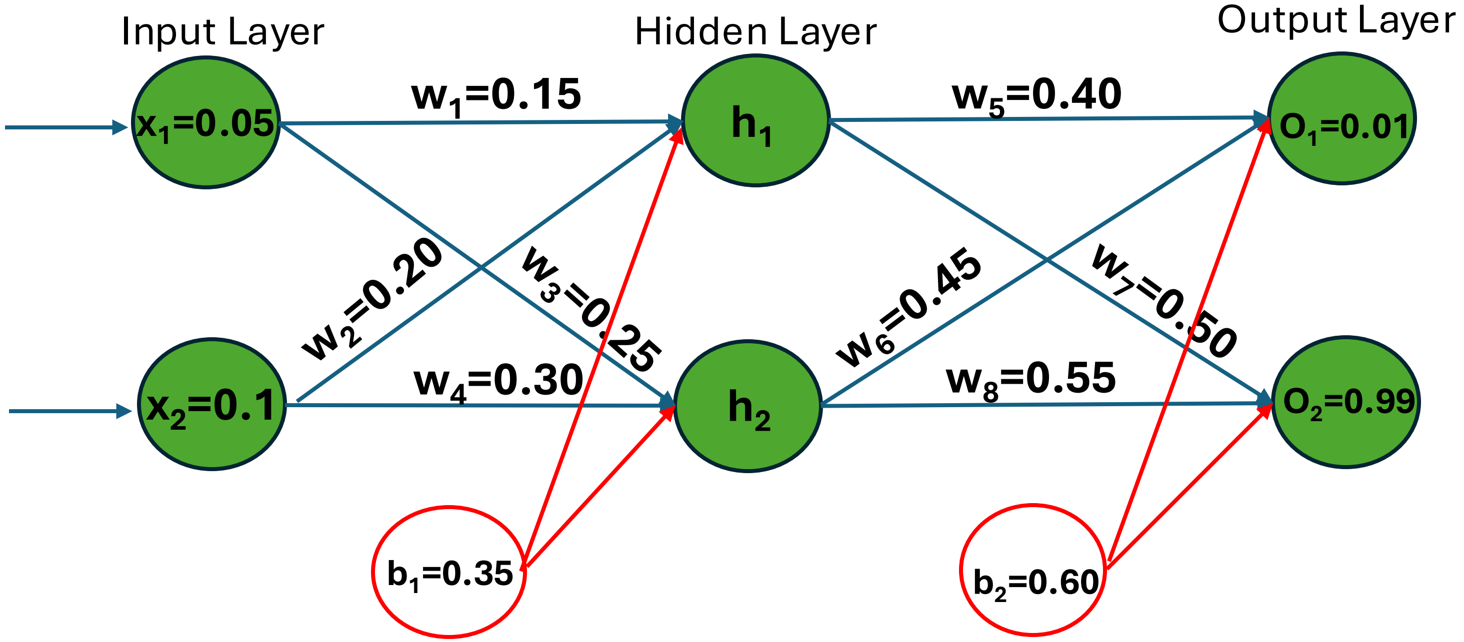
Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

Example of Backpropagation



- **The Backpropagation Algorithm (Hidden Layer):**
  - After finding the new value of w_2, the value of w_3 is calculated as follow:
  - It is stated as $\frac{\partial E_{total}}{\partial w_3}$, and read as the partial derivative of E_{total} with respect to w_3.
  - As, there is no direct connection from the equation between the E_{total} and w_3, we use the chain rule.

Example of Backpropagation



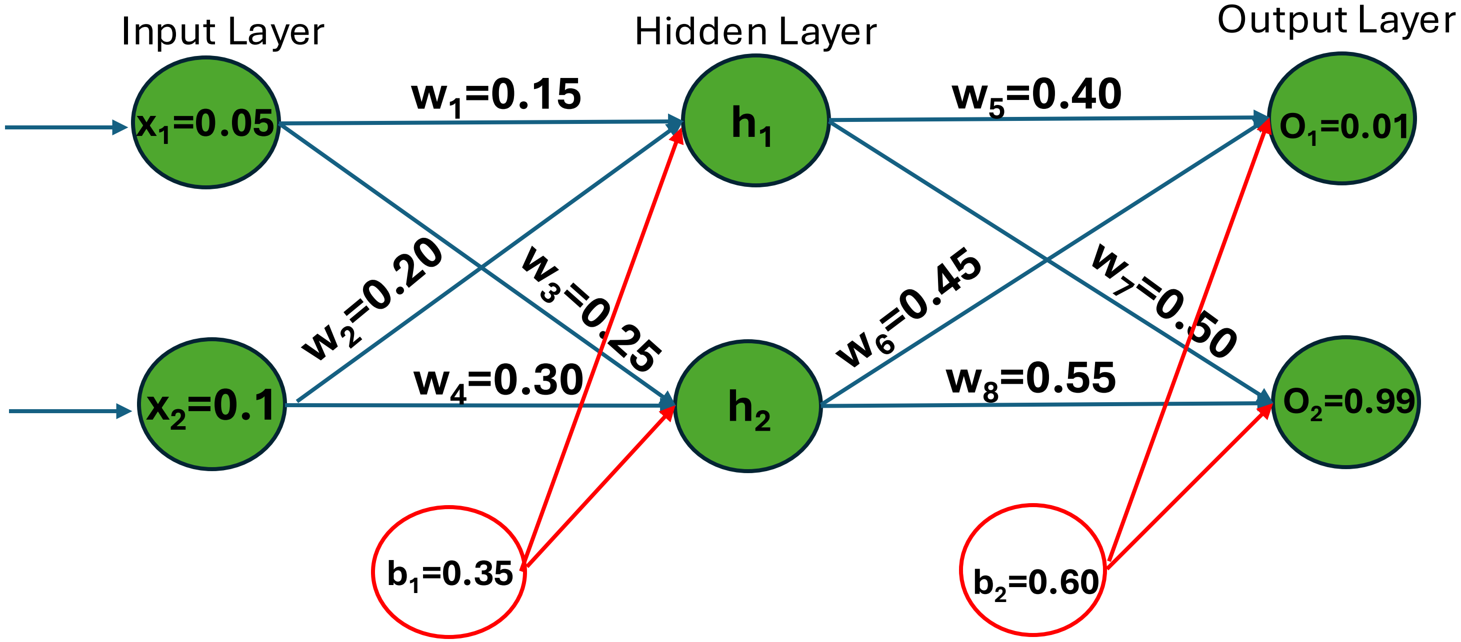
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_3} = \frac{\partial Etotal}{\partial out_{h2}} * \frac{\partial outh_2}{\partial net_{h2}} * \frac{\partial net_{h2}}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

$$\frac{\partial Eo_1}{\partial neto_1} = \frac{\partial Eo_1}{\partial outo_1} * \frac{\partial outo_1}{\partial net_{o1}}$$

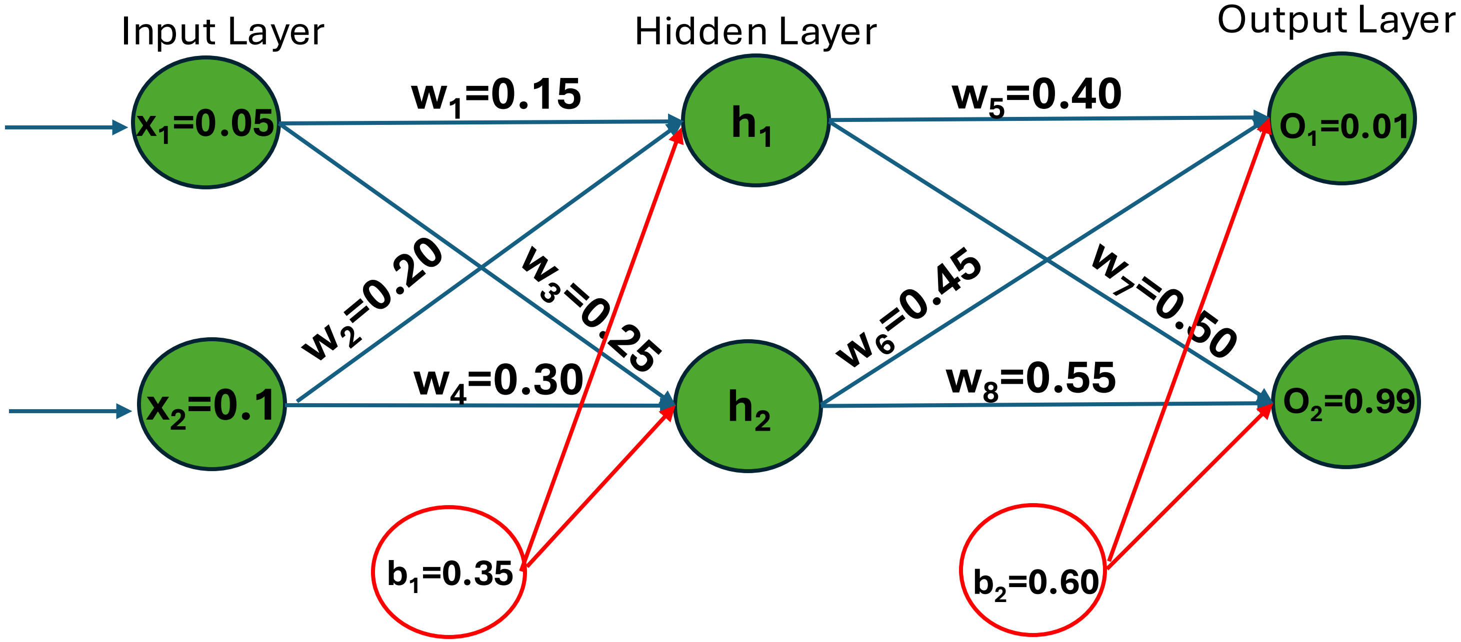
$= (out_{o1} - target_{o1}) * (out_{o1} * (1 - out_{o1}))$

$= (0.75136507 – 0.01) * (0.75136507 * (1- 0.75136507))$

$= 0.74136507 * 0.186815602$

$= 0.138498562$

Example of Backpropagation



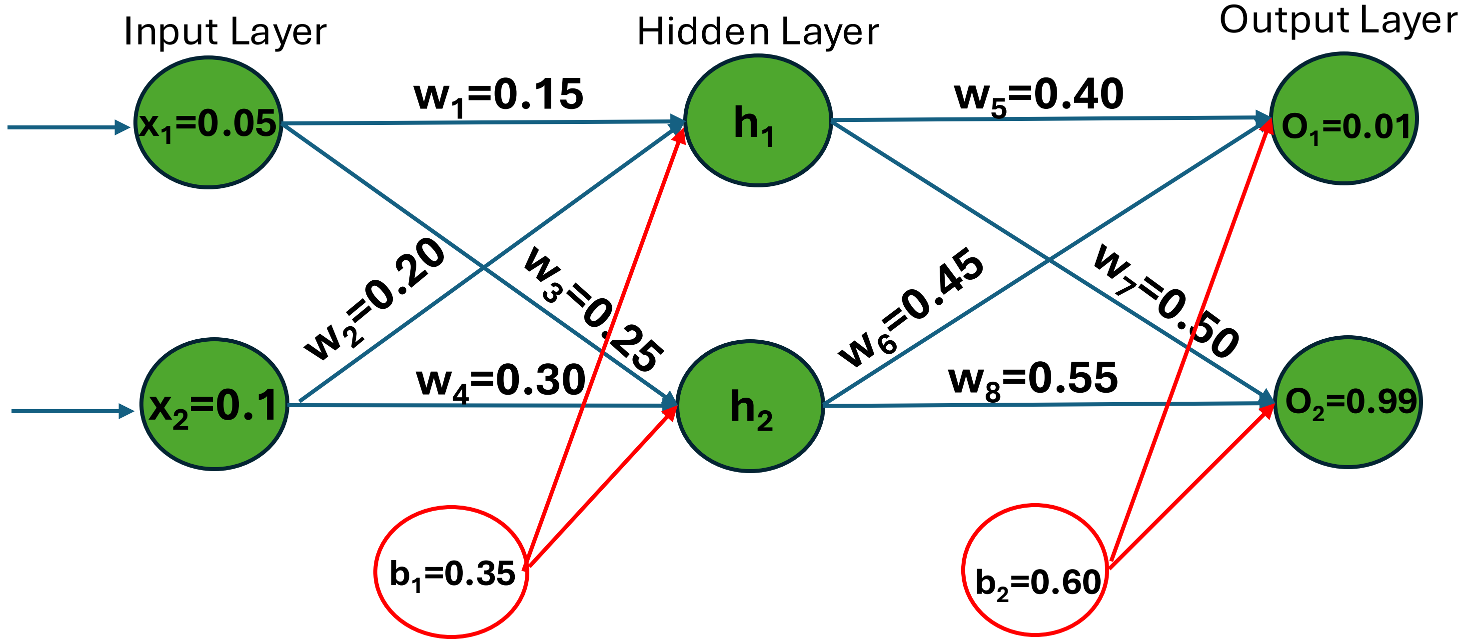
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation

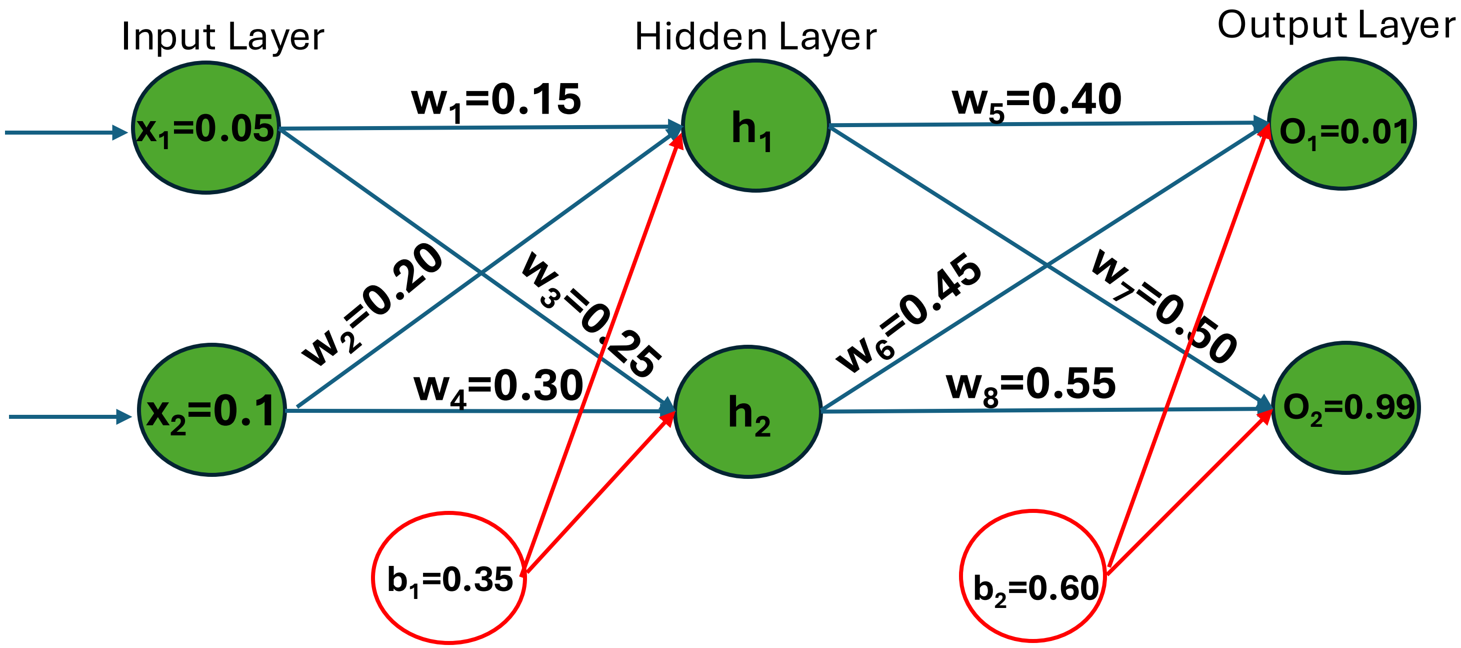


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

$\frac{\partial neto_1}{\partial outh_2} = w_6 = 0.45$

Example of Backpropagation



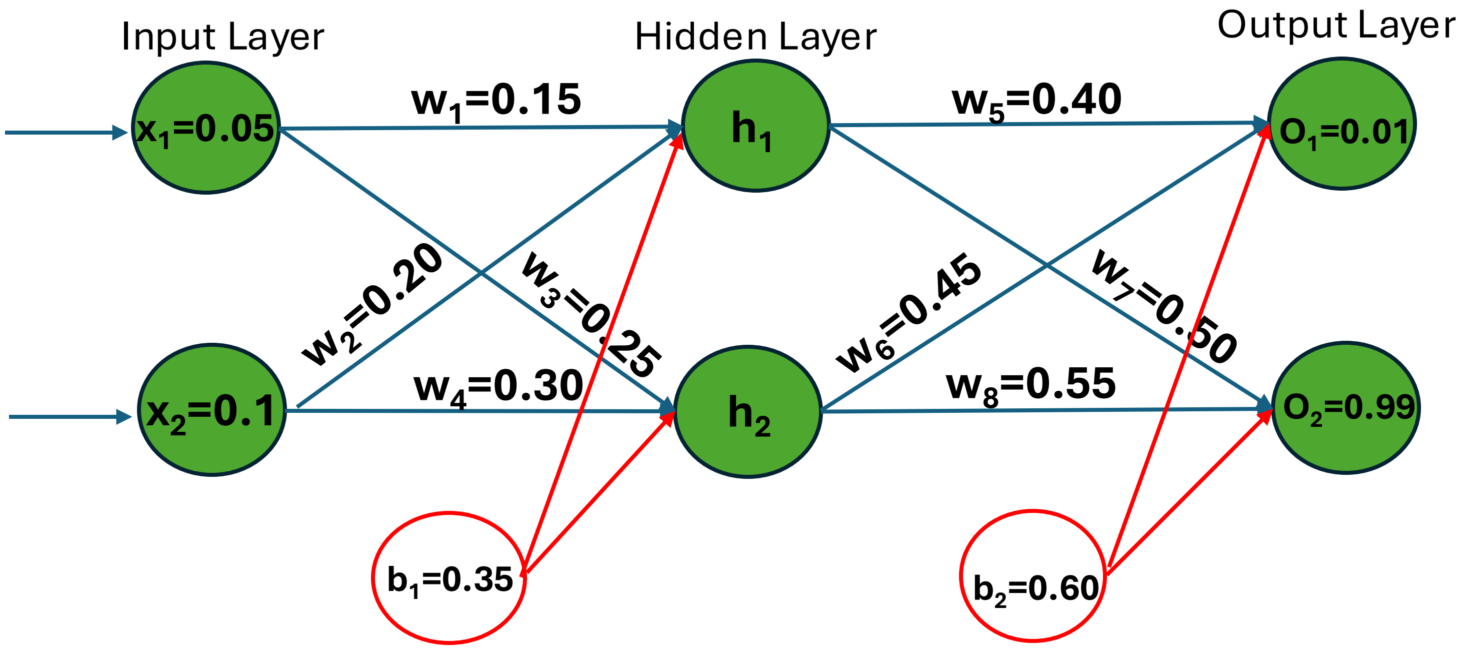
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

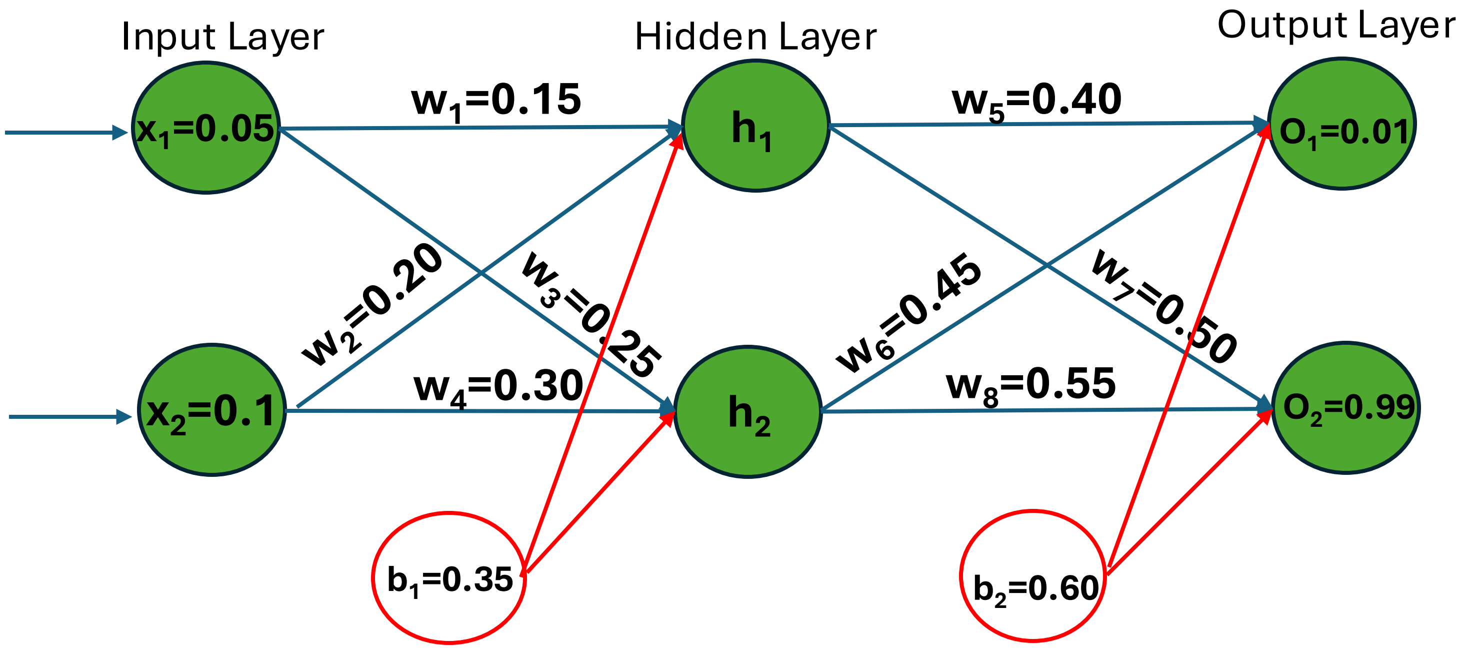
$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h2}} = 0.138498562 * 0.45$$

$$= 0.0623243529$$

Example of Backpropagation



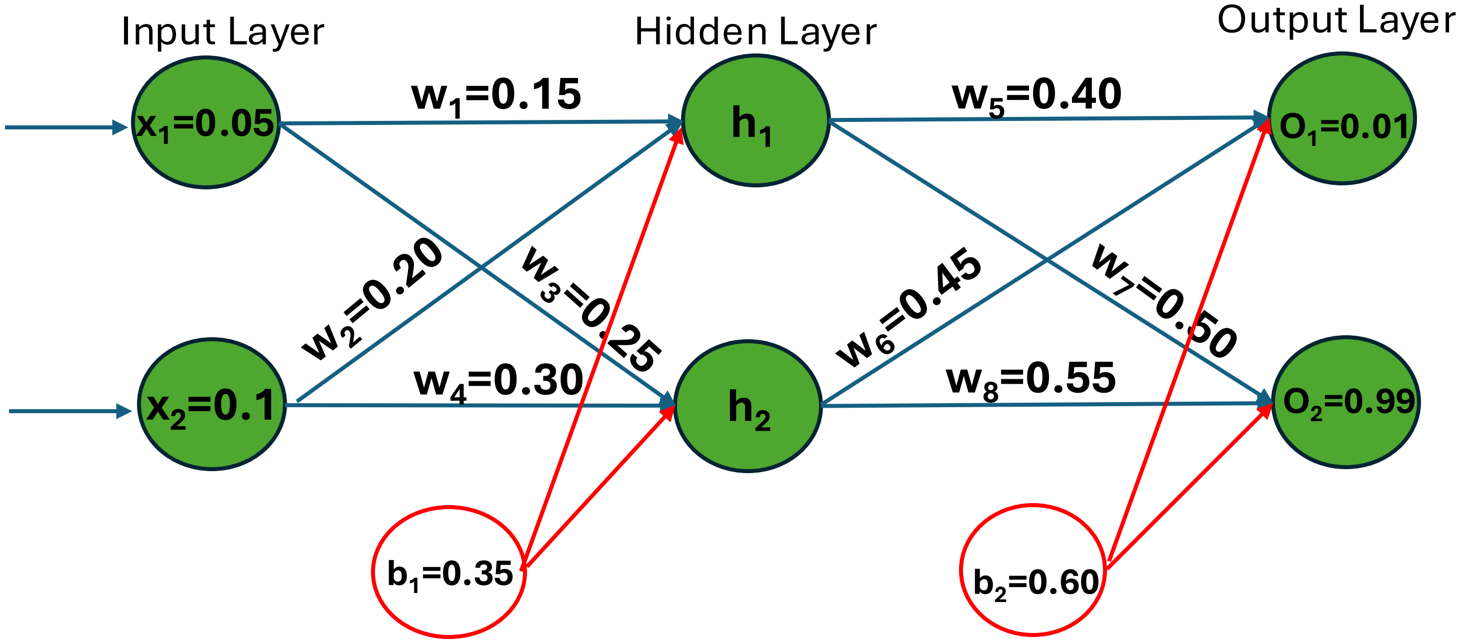
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

$$\frac{\partial Eo_2}{\partial net_{o2}} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial net_{o2}}$$

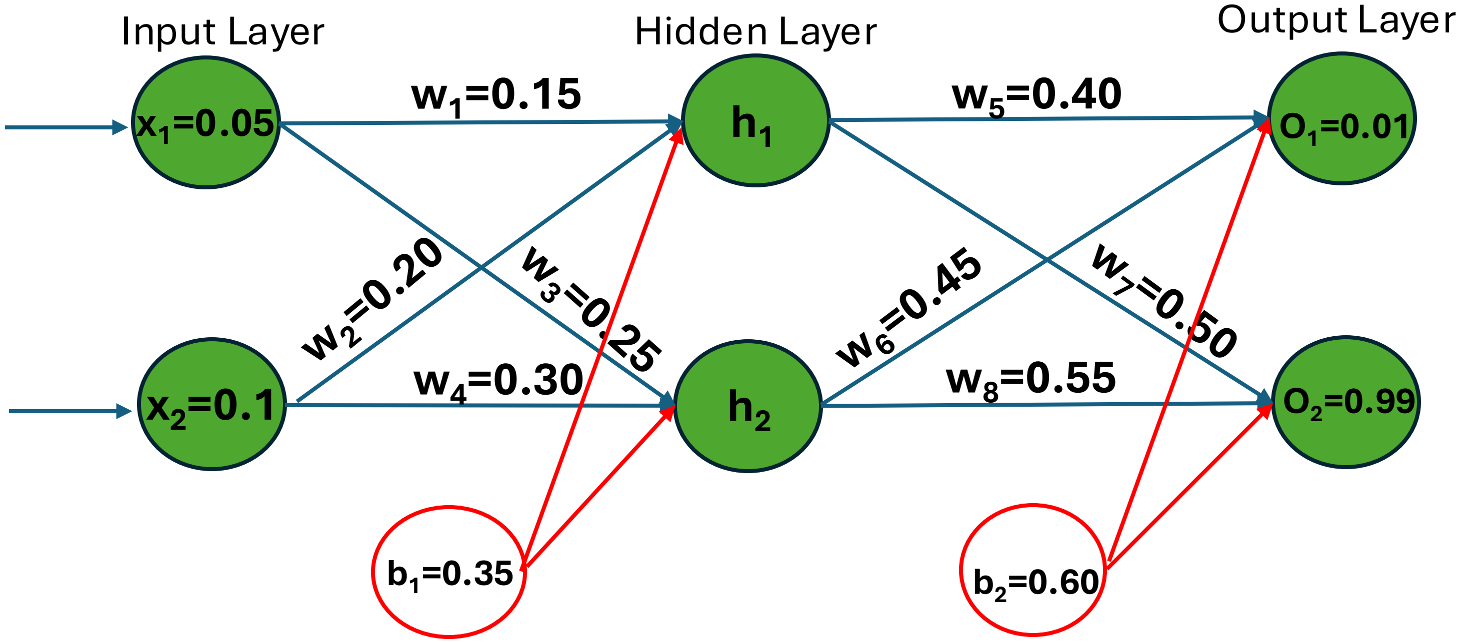
$= (out_{o2} - target_{o2}) * (out_{o2} * (1 - out_{o2}))$

$= (0.772928465 - 0.99) * (0.772928465 * (1- 0.772928465))$

$= -0.217071535 * 0.17551005299$

$= -0.03809823661$

Example of Backpropagation



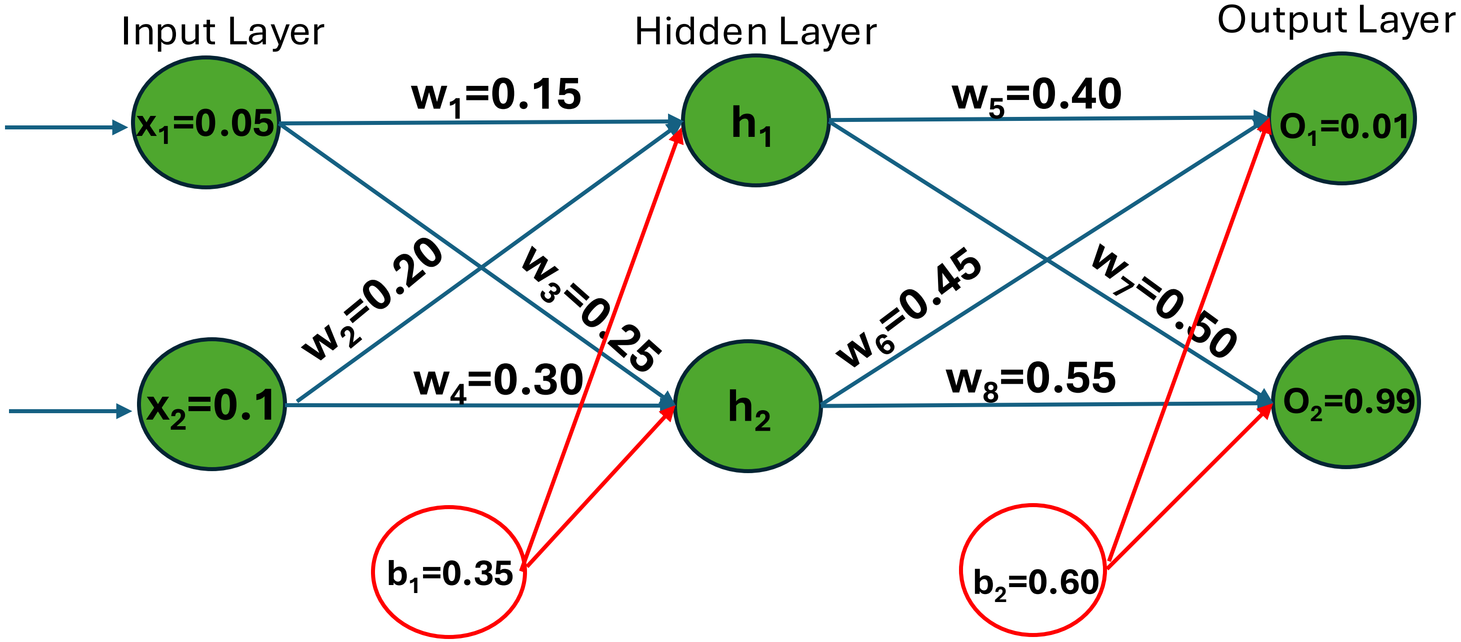
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation

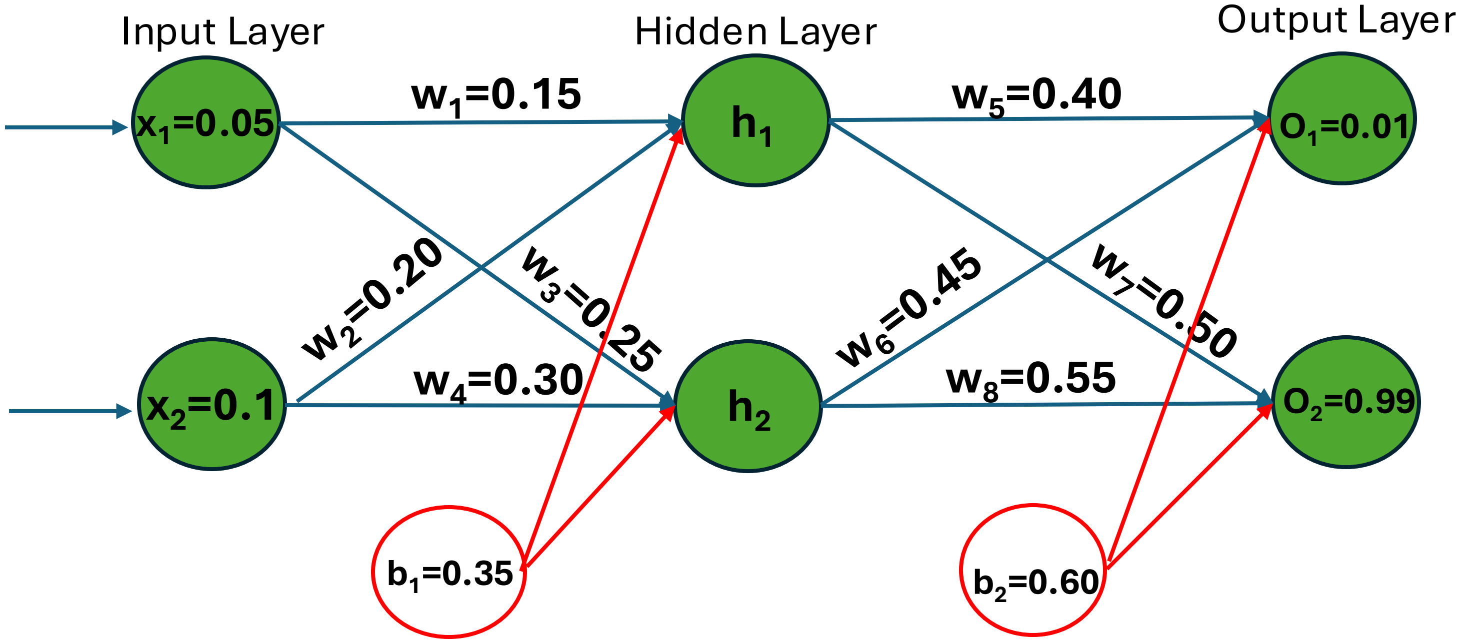


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

$\frac{\partial neto_2}{\partial outh_2} = w_8 = 0.55$

Example of Backpropagation



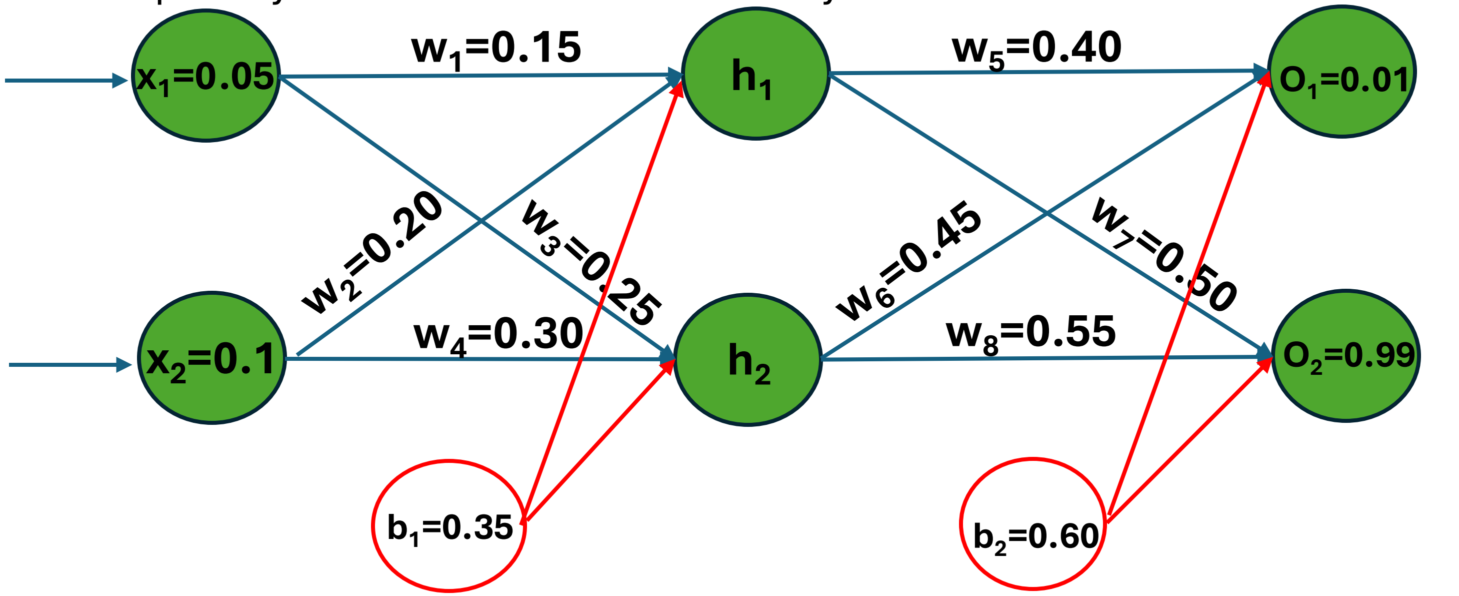
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

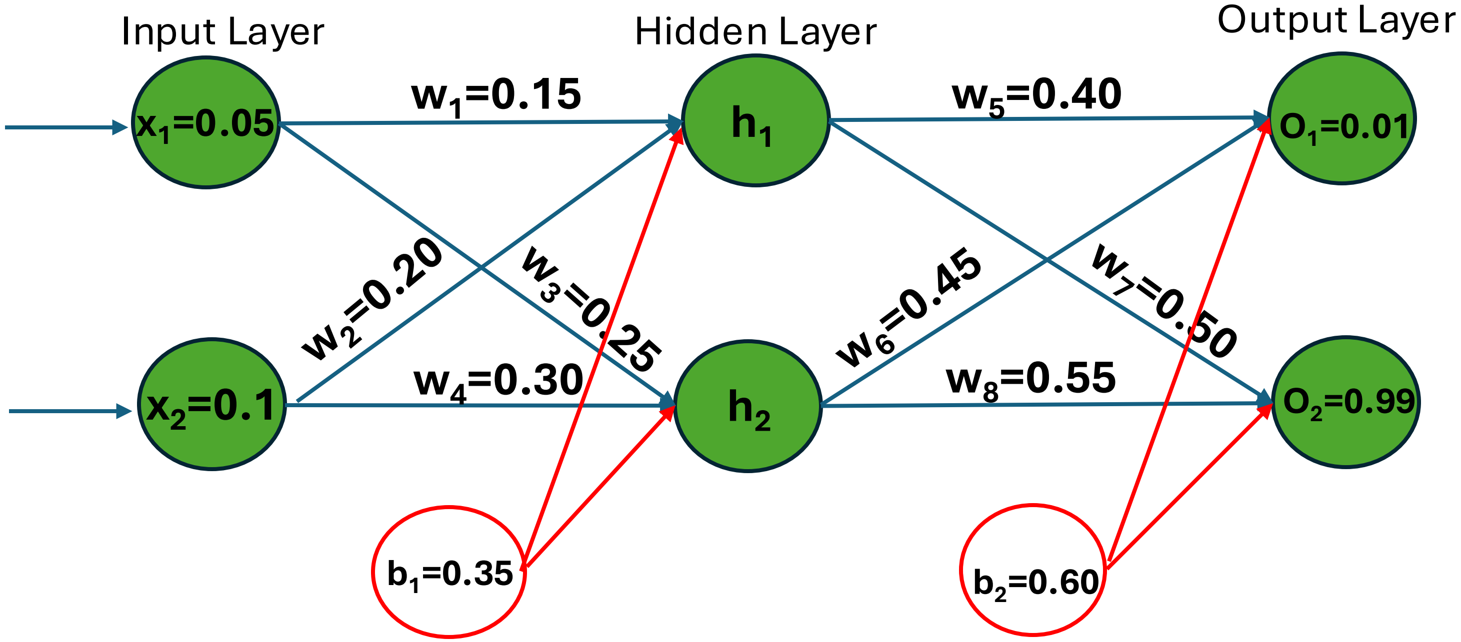
$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o2}}{\partial out_{h2}} = -0.03809823661 * 0.55$$

$$= -0.02095403013$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

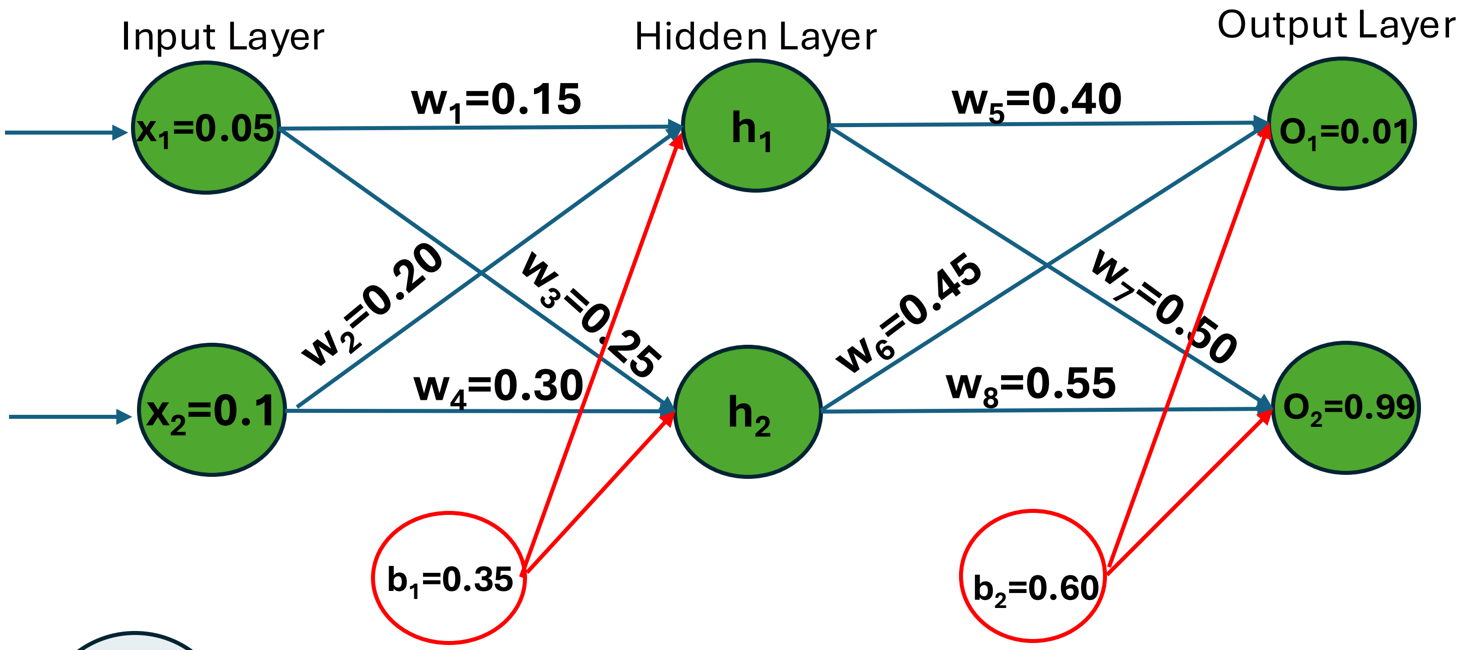
$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3};$$

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_1 has direct affect on both o_1 and o_2.

= 0.0623243529 + (-0.02095403013)

= 0.04137032277

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

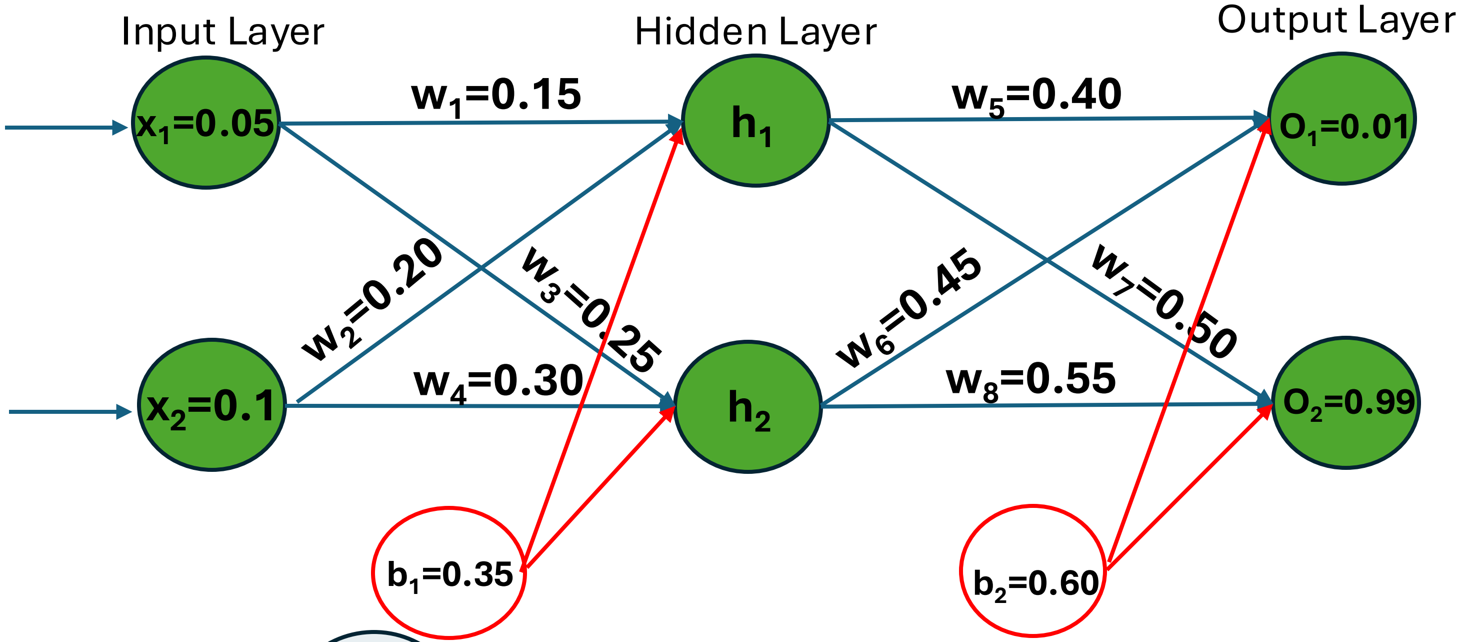
$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3};$$

2) $\frac{\partial outh_2}{\partial net_{h2}} = out_{h2} * (1 - out_{h2})$

$= 0.596884378 * (1 - 0.596884378)$

$= 0.2406134173$

Example of Backpropagation

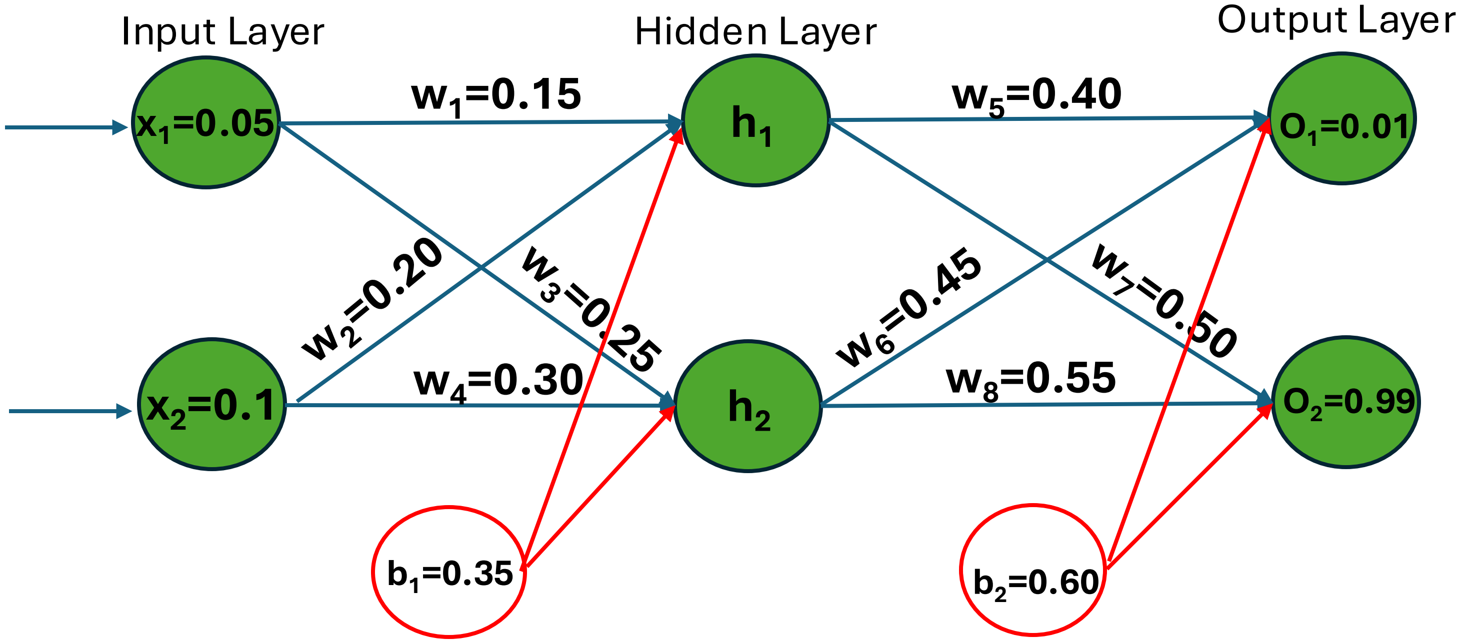


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3};$$

2) $\frac{\partial net_{h2}}{\partial w_3} = x_1 = 0.05$

Example of Backpropagation



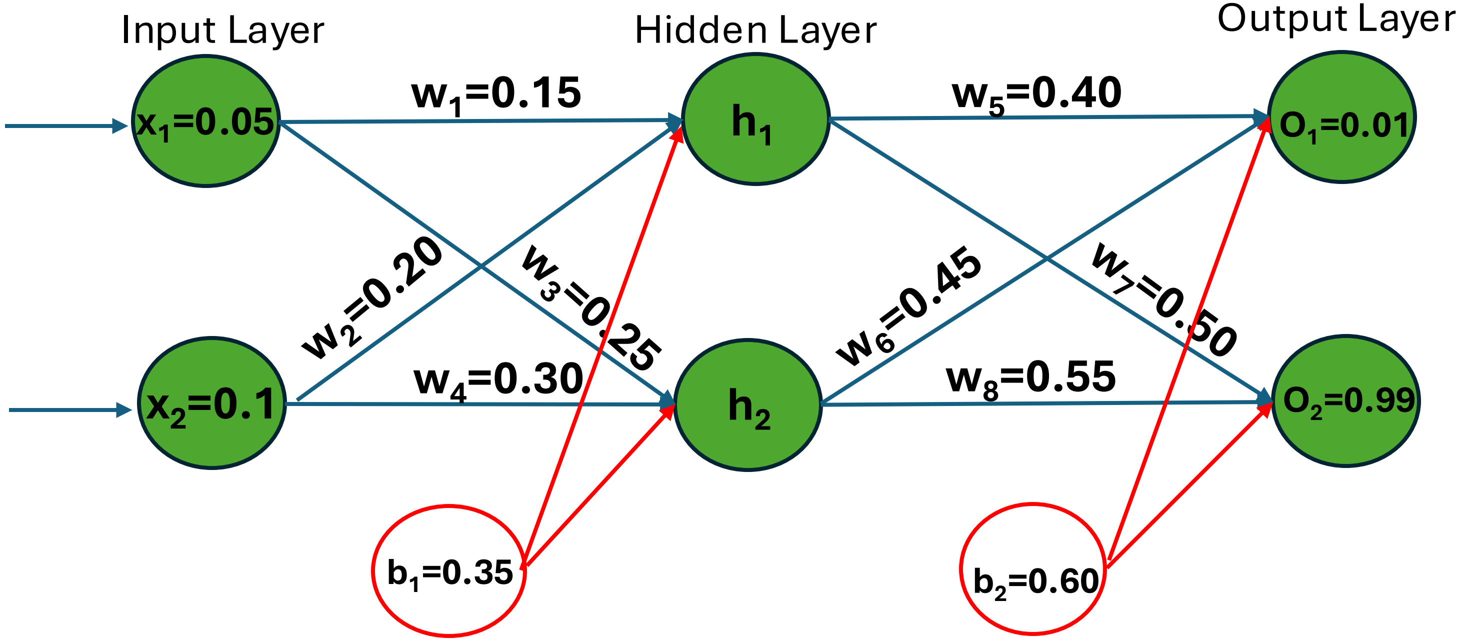
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_3} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_3};$$

$= 0.04137032277 * 0.2406134173 * 0.05$

$= 0.00049771273$

Example of Backpropagation



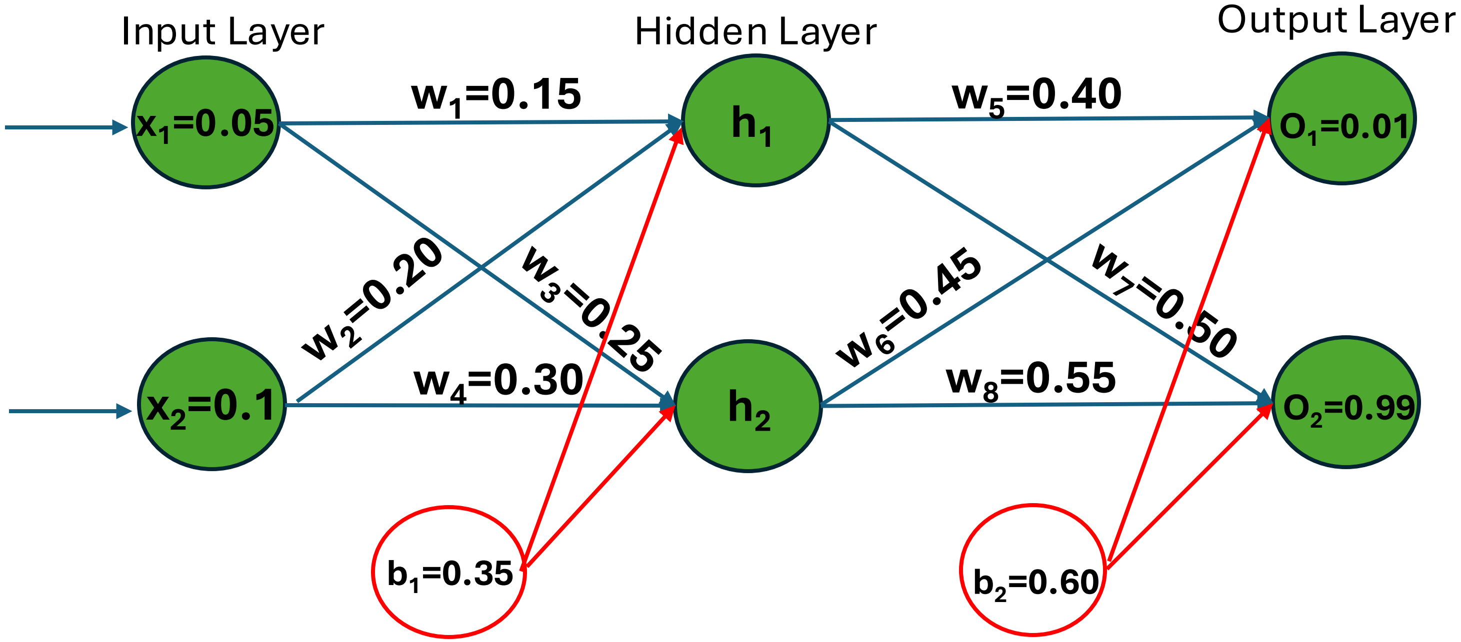
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$w_3^+ = w_3 - \eta * \frac{\partial Etotal}{\partial w_3}$$

$$= 0.25 - 0.5 * 0.00049771273$$

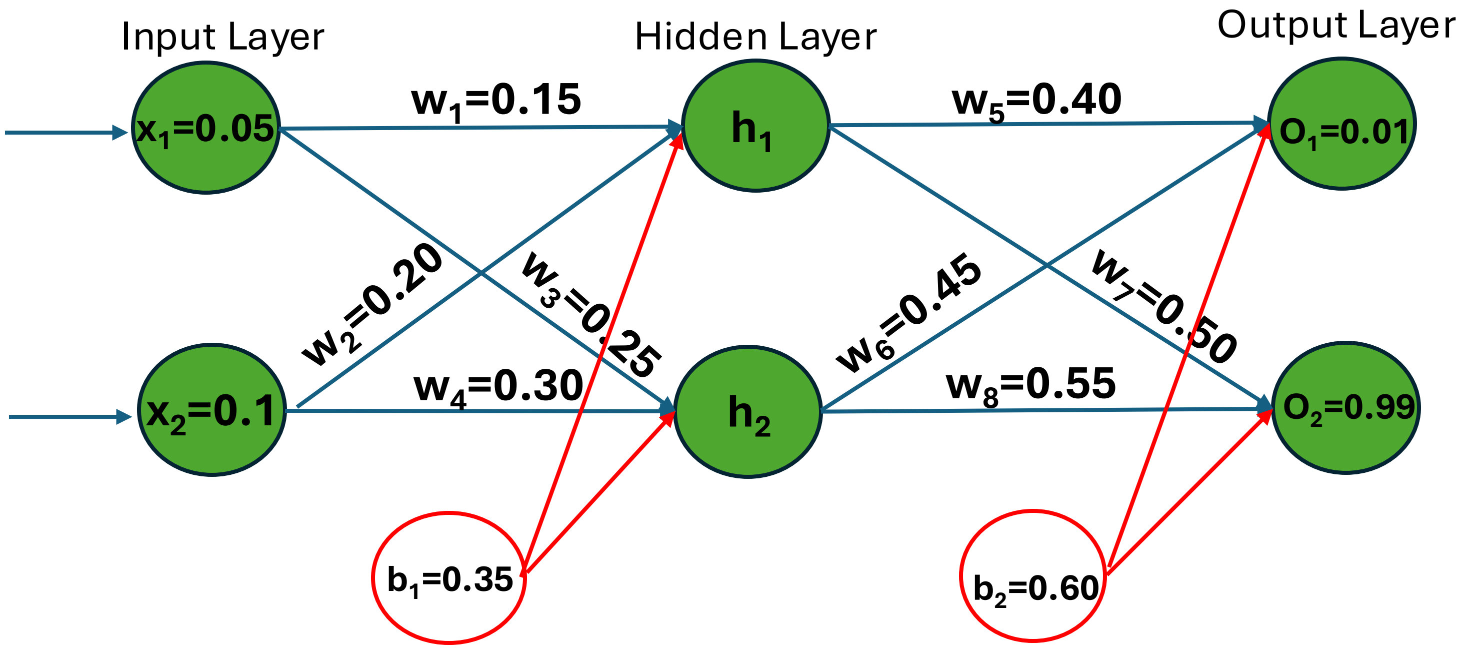
$$= 0.24975114$$

Example of Backpropagation



- **The Backpropagation Algorithm (Hidden Layer):**
  - After finding the new value of w_3, the value of w_4 is calculated as follow:
  - It is stated as $\frac{\partial E_{total}}{\partial w_4}$, and read as the partial derivative of E_{total} with respect to w_4.
  - As, there is no direct connection from the equation between the E_{total} and w_4, we use the chain rule.

Example of Backpropagation



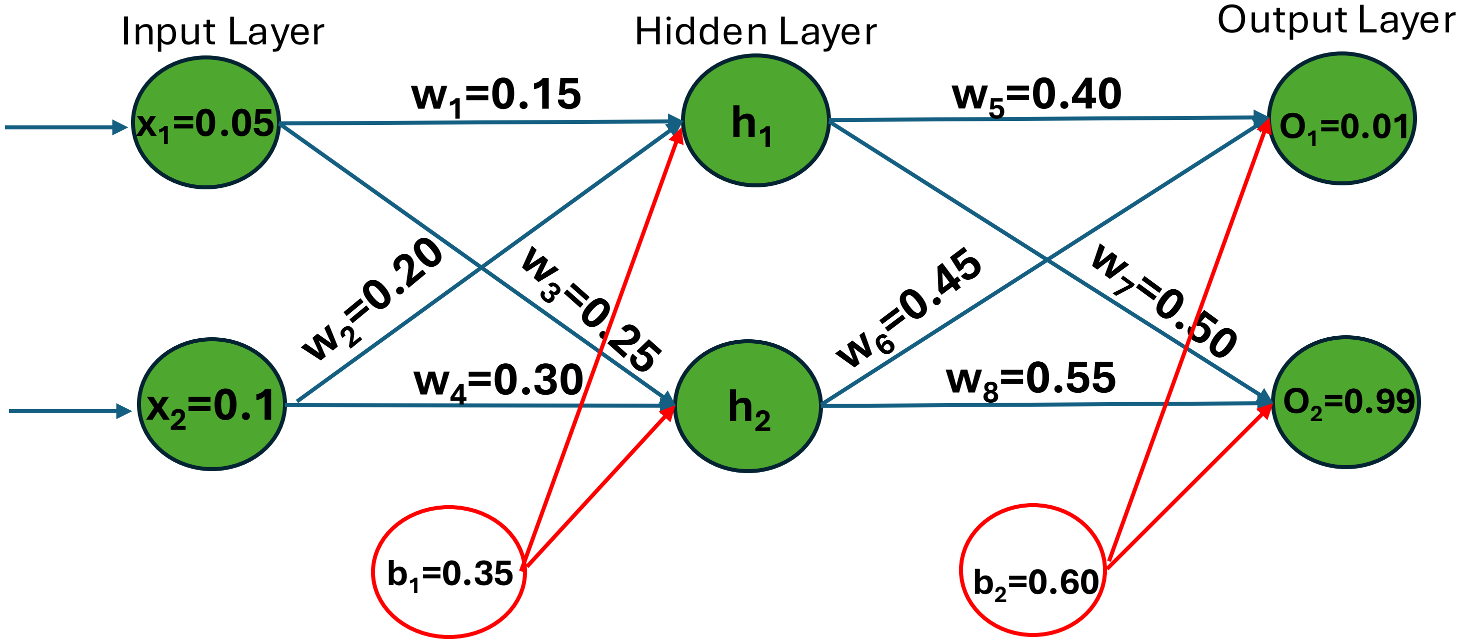
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial Etotal}{\partial out_{h2}} * \frac{\partial outh_2}{\partial net_{h2}} * \frac{\partial net_{h2}}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial net_{o1}} * \frac{\partial net_{o1}}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

$$\frac{\partial Eo_1}{\partial neto_1} = \frac{\partial Eo_1}{\partial outo_1} * \frac{\partial outo_1}{\partial net_{o1}}$$

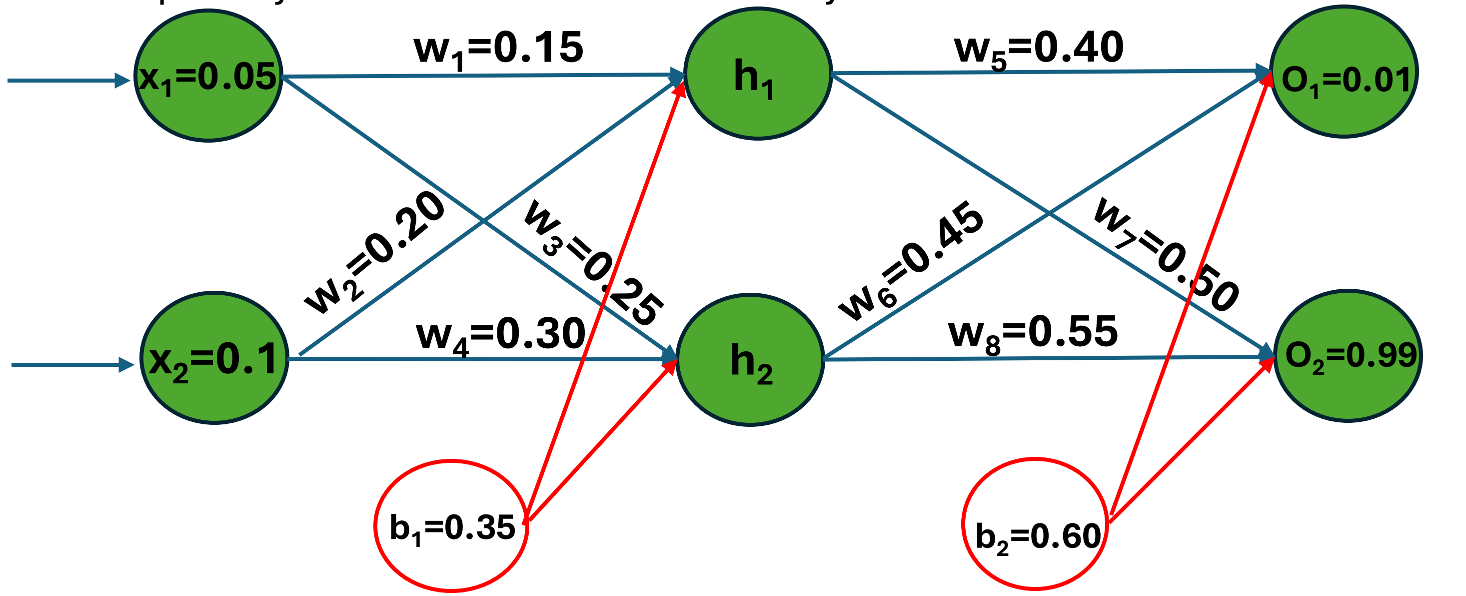
$= (out_{o1} - target_{o1}) * (out_{o1} * (1 - out_{o1}))$

$= (0.75136507 – 0.01) * (0.75136507 * (1- 0.75136507))$

$= 0.74136507 * 0.186815602$

$= 0.138498562$

Example of Backpropagation



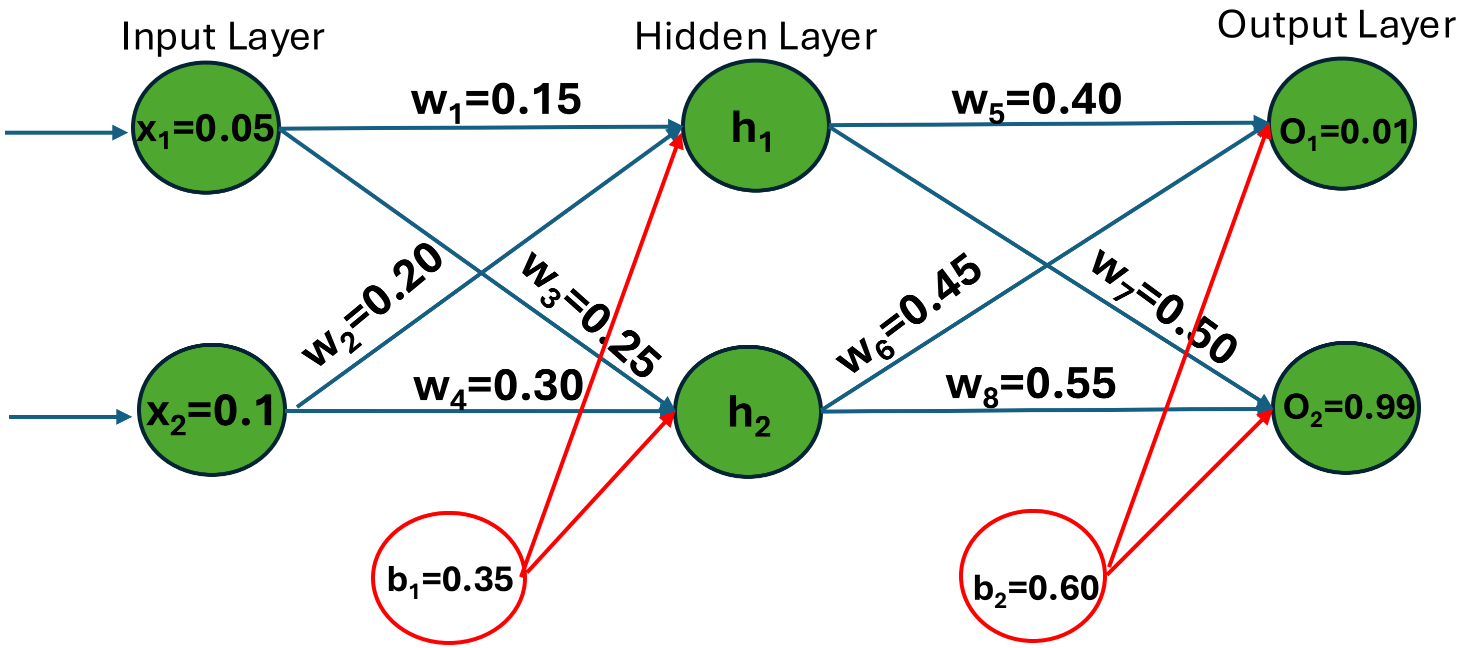
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation

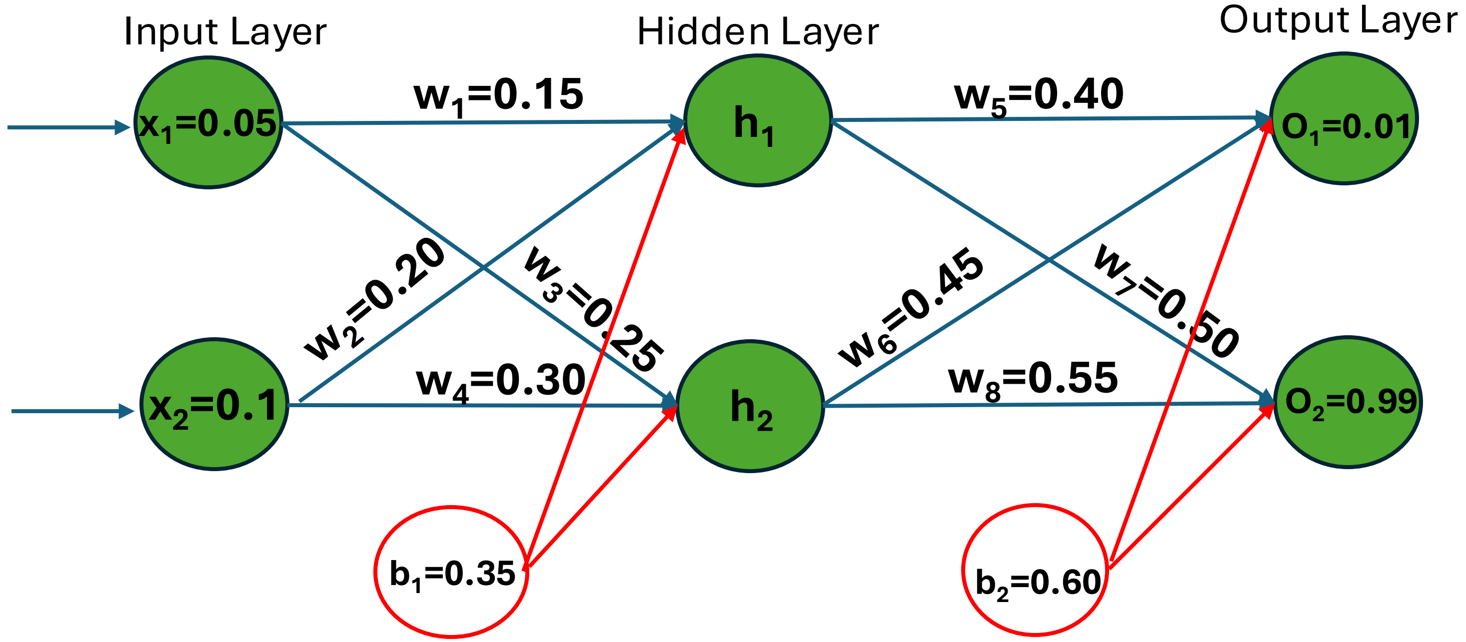


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

$\frac{\partial neto_1}{\partial outh_2} = w_6 = 0.45$

Example of Backpropagation



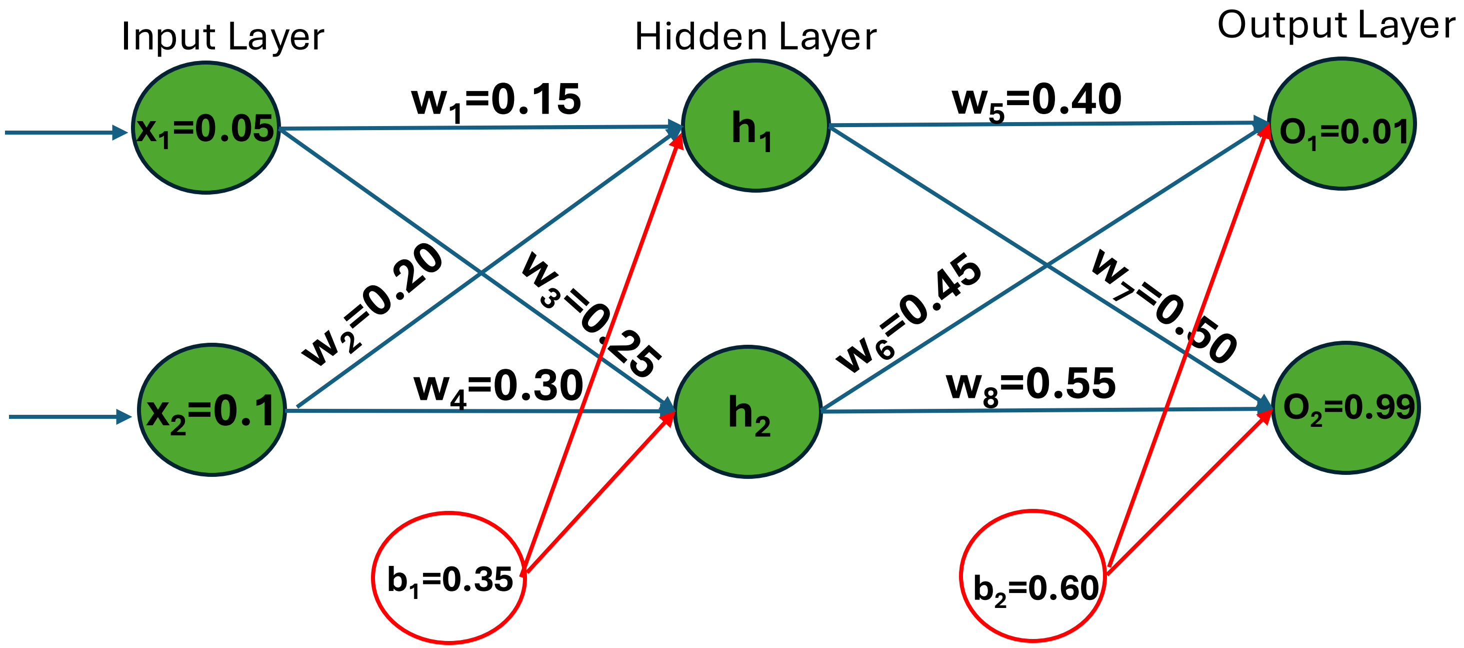
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h1}} = \frac{\partial Eo_1}{\partial outh_1} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \qquad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

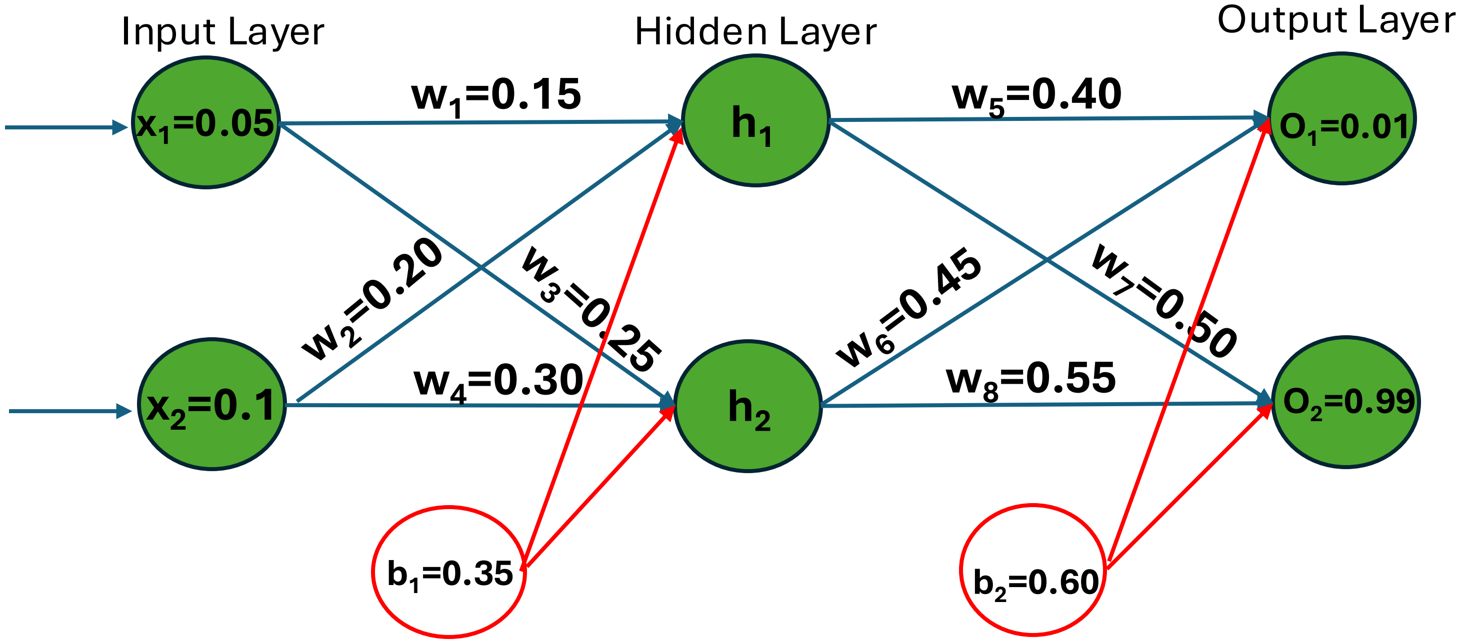
$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_1}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o1}}{\partial out_{h2}} = 0.138498562 * 0.45$$

$$= 0.0623243529$$

Example of Backpropagation



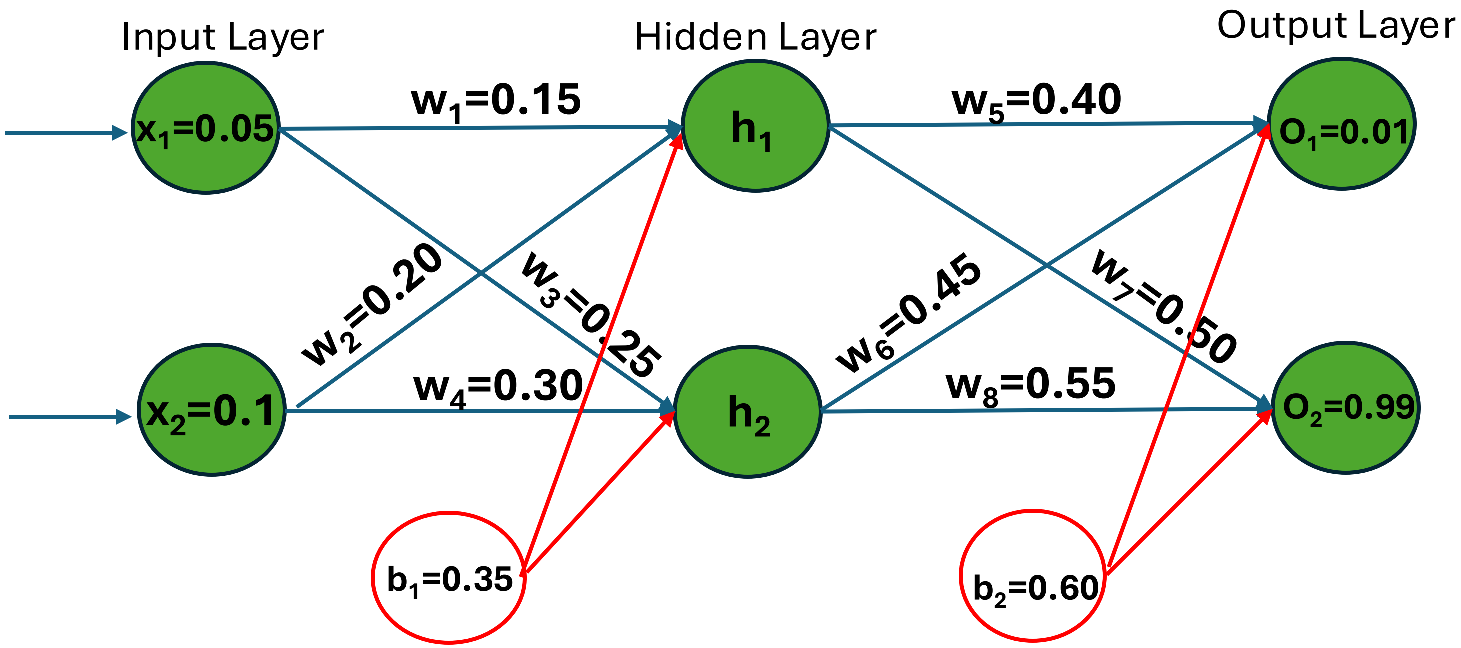
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

$$\frac{\partial Eo_2}{\partial net_{o2}} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial net_{o2}}$$

$= (out_{o2} - target_{o2}) * (out_{o2} * (1 - out_{o2}))$

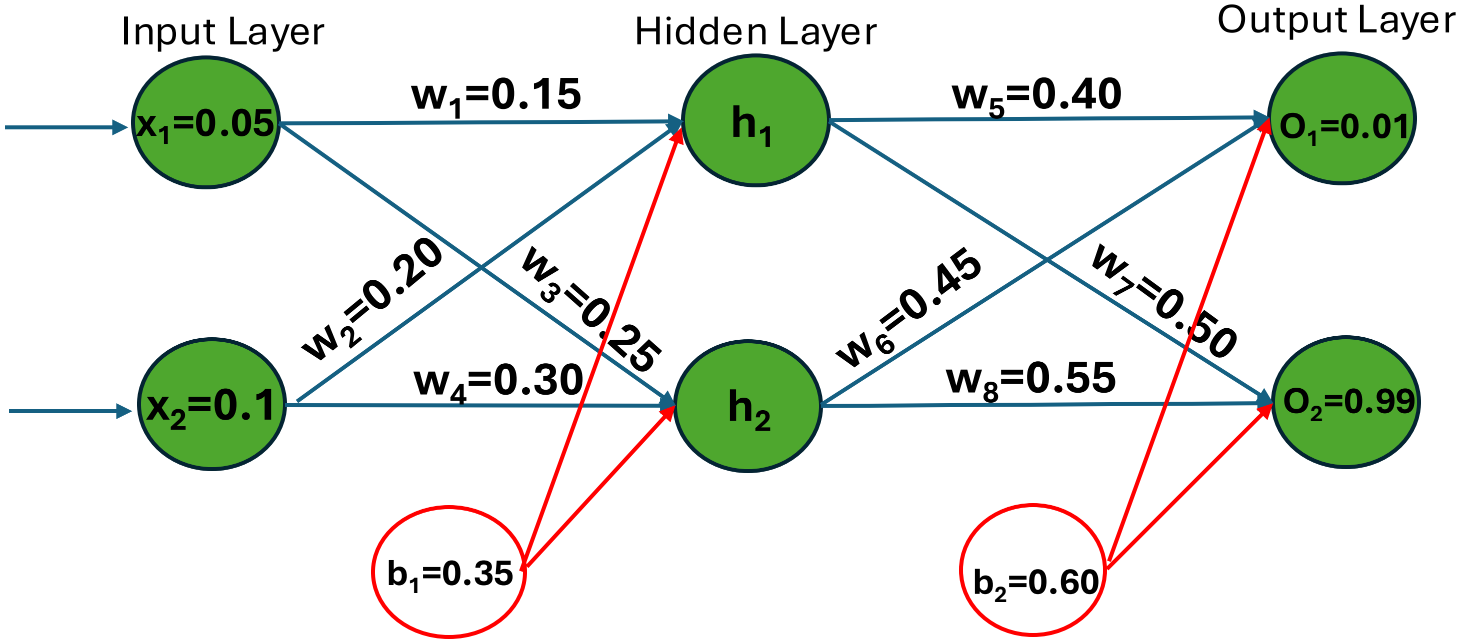
$= (0.772928465 – 0.99) * (0.772928465 * (1- 0.772928465))$

$= -0.217071535 * 0.17551005299$

$= -0.03809823661$

Prof. Mohammed A. Al Ghamdi, Machine Learning Course.

Example of Backpropagation



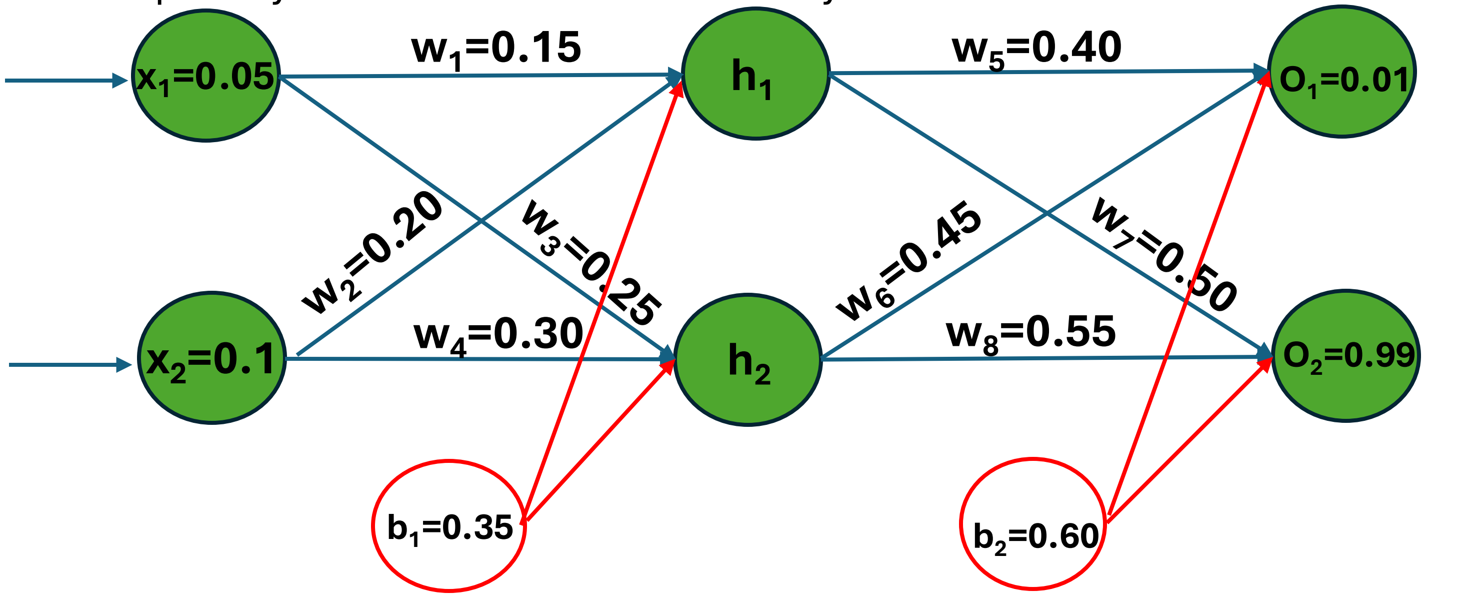
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \qquad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation

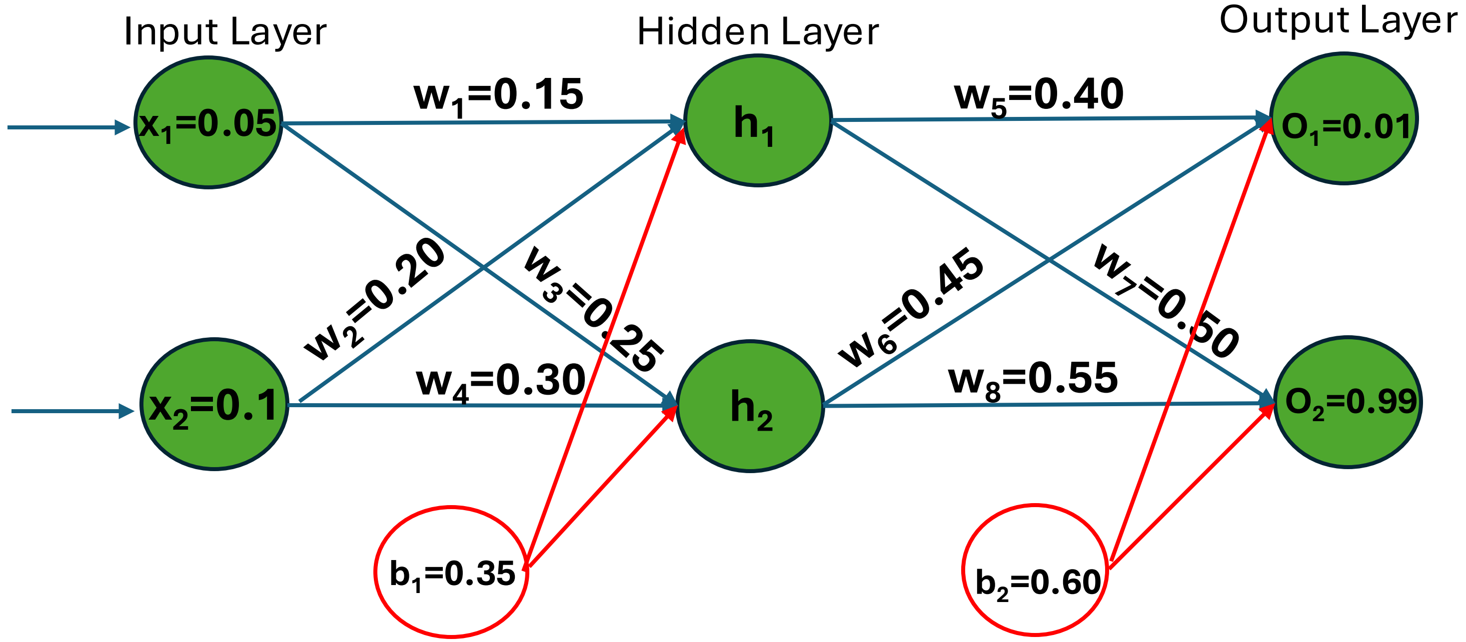


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

$\frac{\partial neto_2}{\partial outh_2} = w_8 = 0.55$

Example of Backpropagation



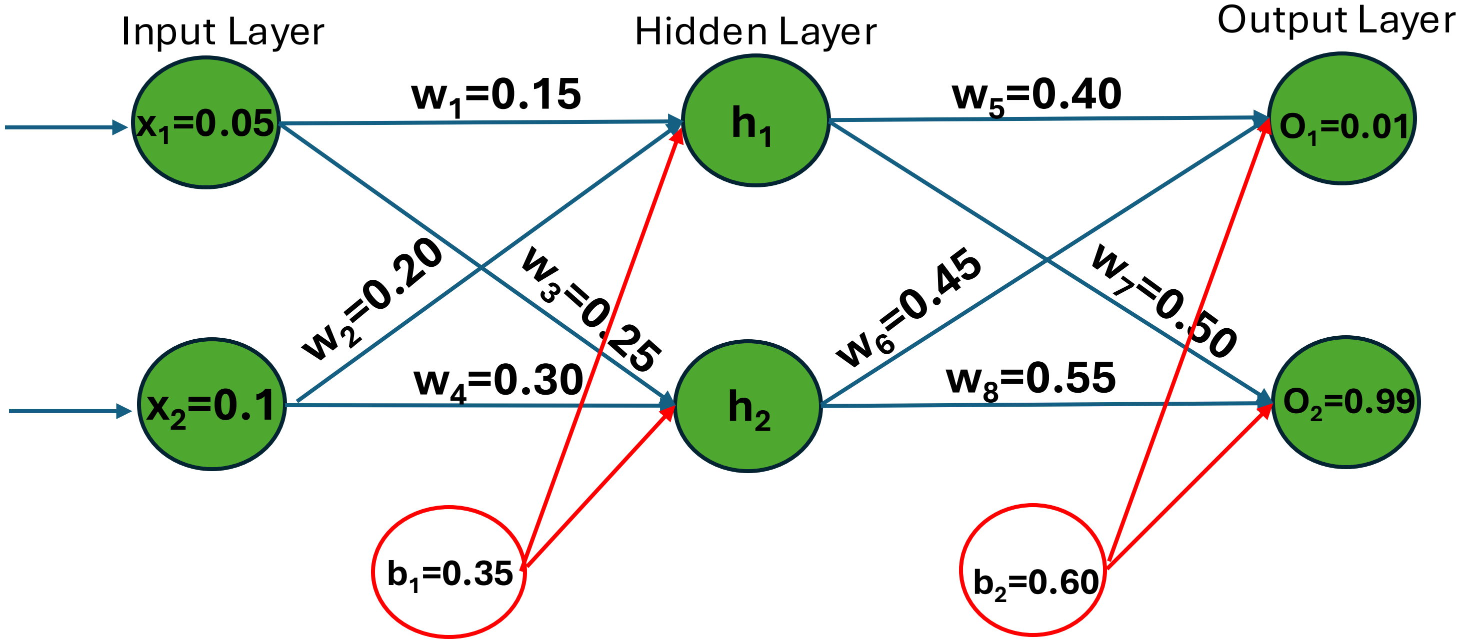
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial Eo_1}{\partial outh_2} = \frac{\partial Eo_1}{\partial neto_1} * \frac{\partial neto_1}{\partial outh_2}, \quad \frac{\partial Eo_2}{\partial outh_2} = \frac{\partial Eo_2}{\partial outo_2} * \frac{\partial outo_2}{\partial outh_2}$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

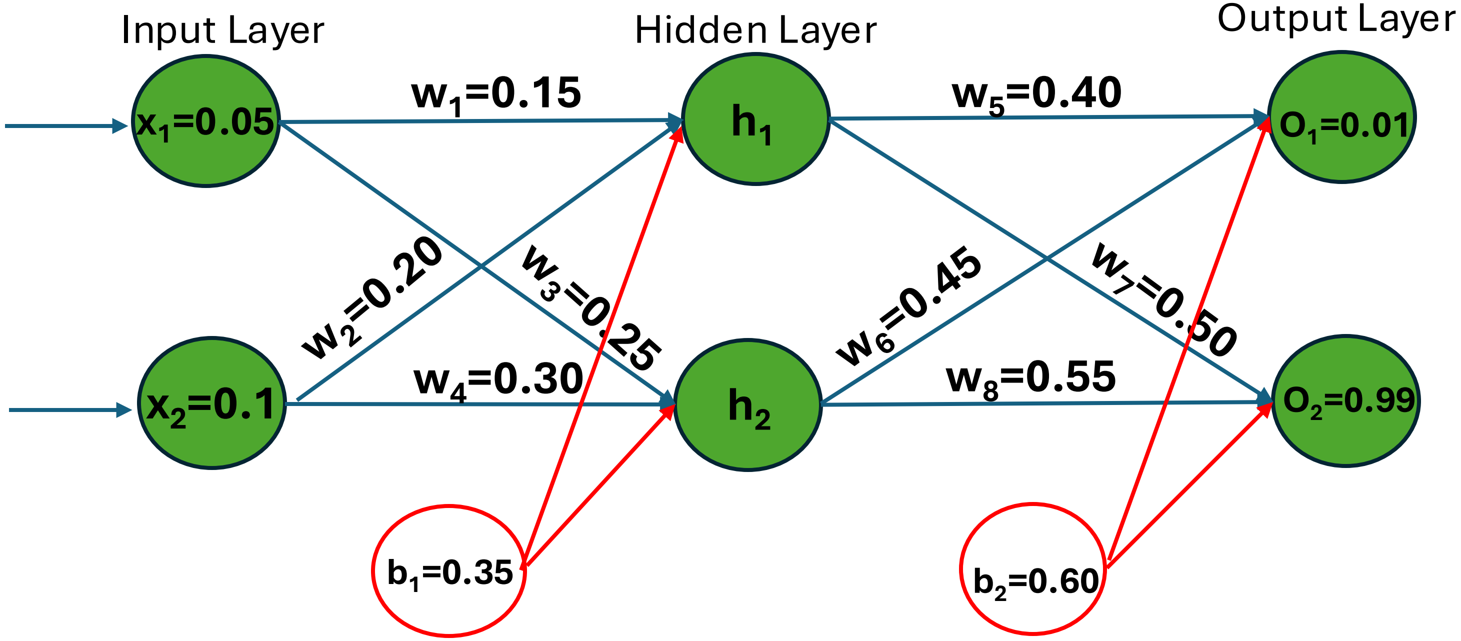
$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4}$$, where;

1) $\frac{\partial Etotal}{\partial out_{h2}} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

$$\frac{\partial E_{o2}}{\partial out_{h2}} = -0.03809823661 * 0.55$$

$$= -0.02095403013$$

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

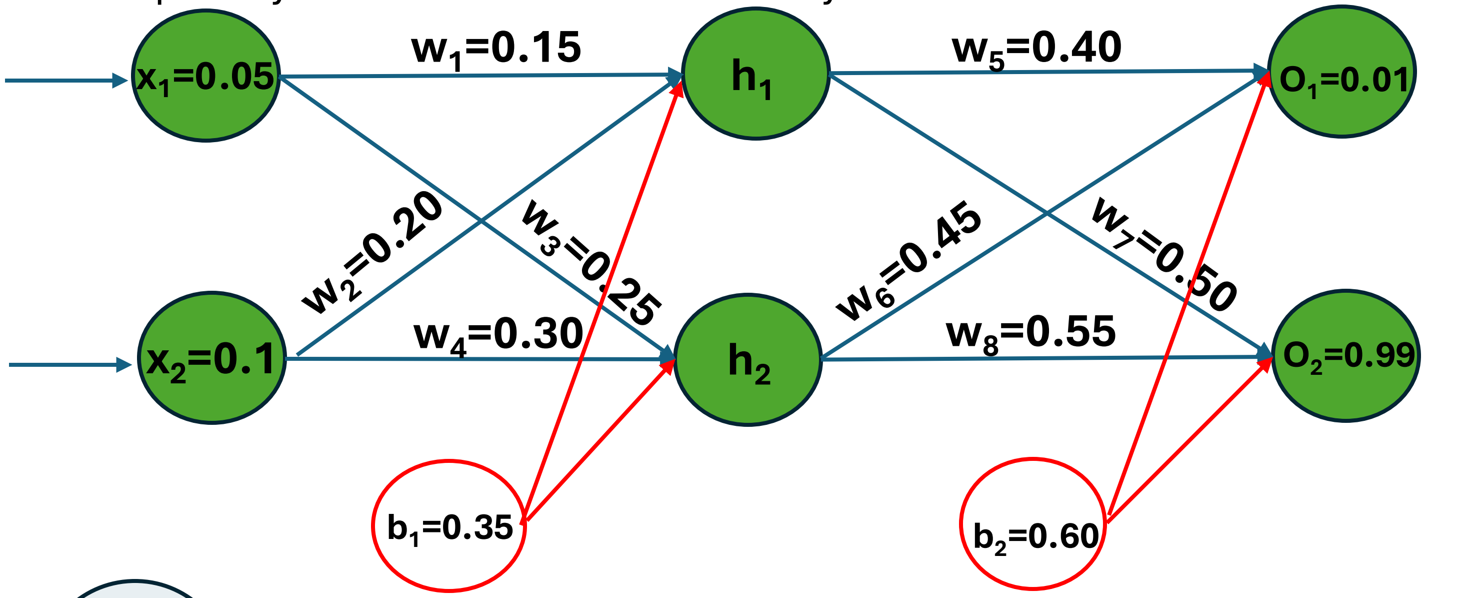
$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4};$$

1) $\frac{\partial Etotal}{\partial outh_2} = \frac{\partial Eo_1}{\partial outh_2} + \frac{\partial Eo_2}{\partial outh_2}$; as said h_2 has direct affect on both o_1 and o_2.

= 0.0623243529 + (-0.02095403013)

= 0.04137032277

Example of Backpropagation



o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

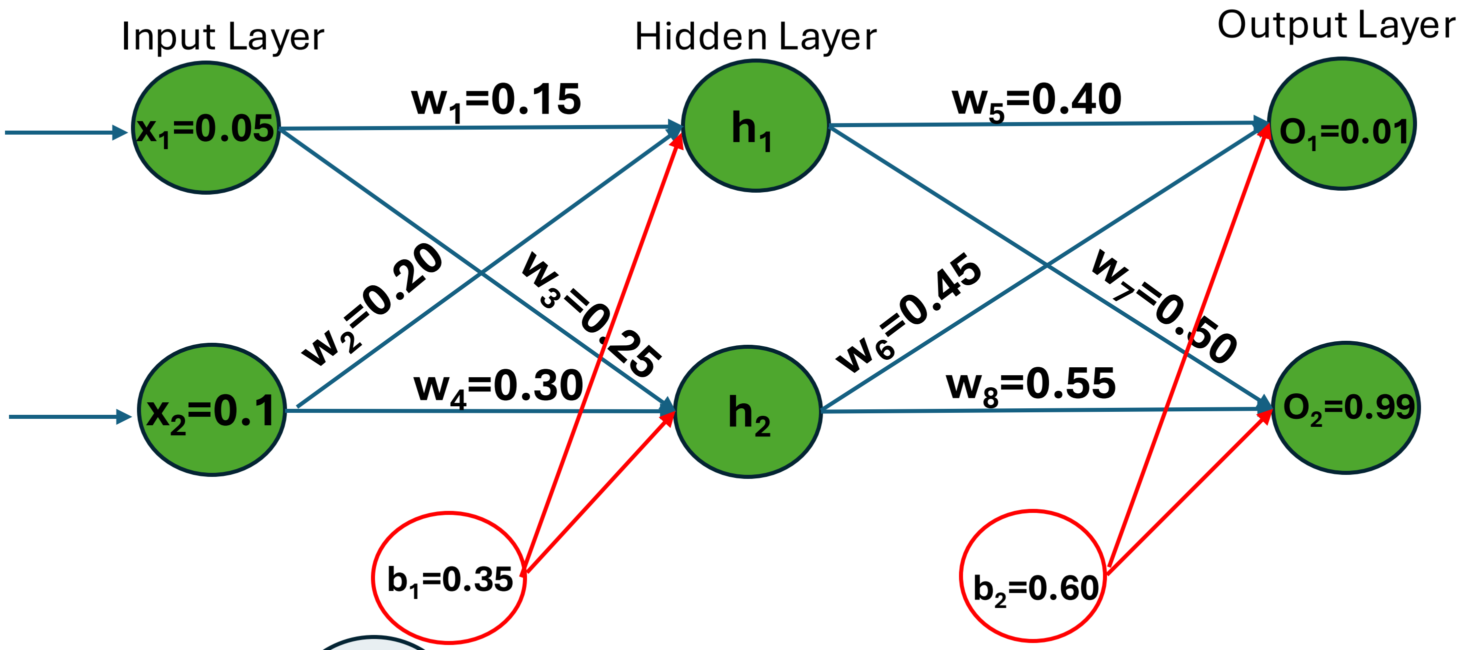
$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4};$$

2) $\frac{\partial outh_2}{\partial net_{h2}} = out_{h2} * (1 - out_{h2})$

$= 0.596884378 * (1 - 0.596884378)$

$= 0.2406134173$

Example of Backpropagation

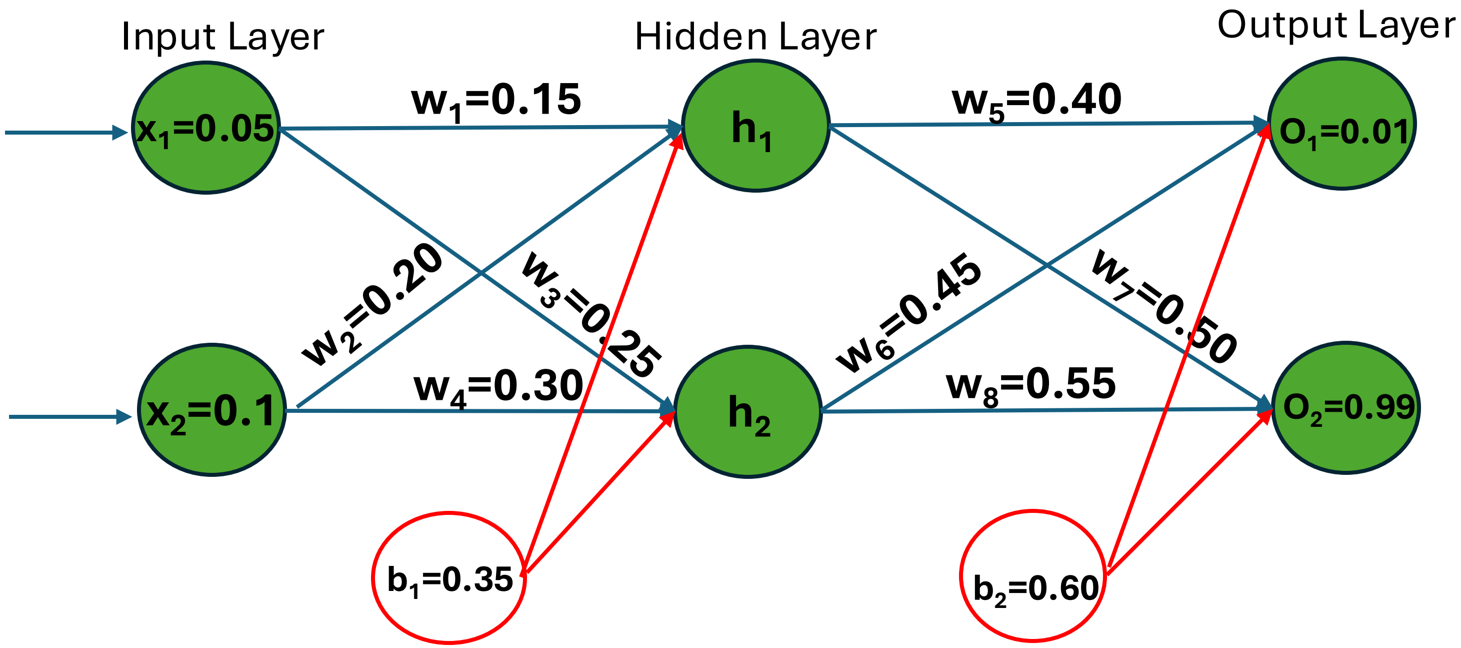


o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4};$$

2) $\frac{\partial net_{h2}}{\partial w_4} = x_2 = 0.1$

Example of Backpropagation



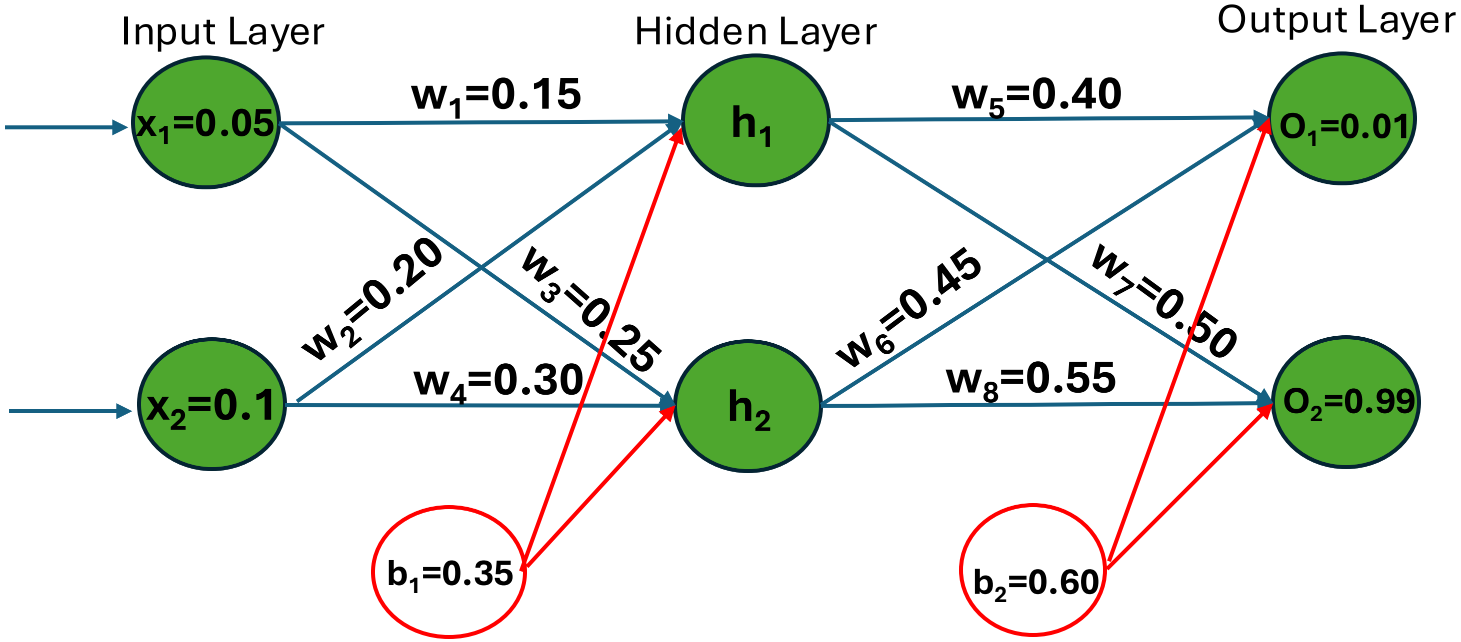
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$\frac{\partial Etotal}{\partial w_4} = \frac{\partial Etotal}{\partial outh_2} * \frac{\partial outh_2}{\partial neth_2} * \frac{\partial neth_2}{\partial w_4};$$

= 0.04137032277 * 0.2406134173 * 0.1

= 0.00099542547

Example of Backpropagation



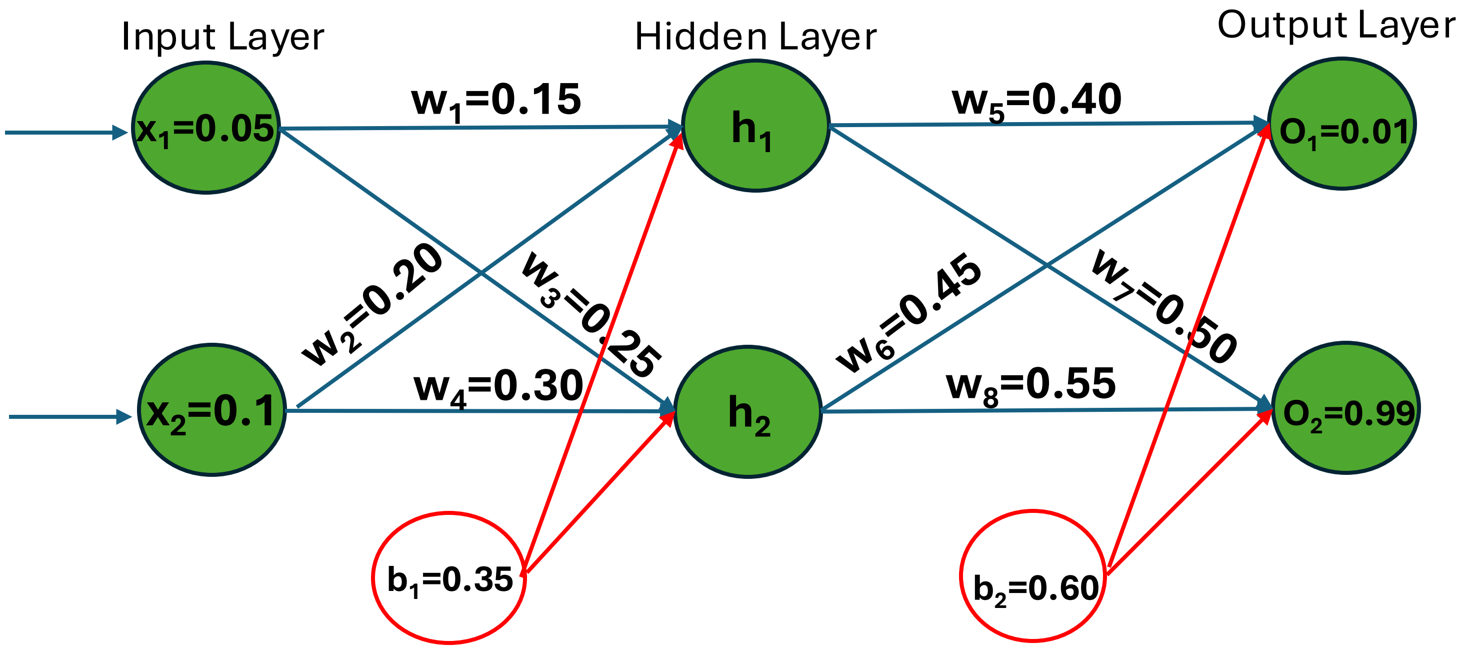
o_1	0.75136507
o_2	0.772928465
h_1	0.593269992
h_2	0.596884378

$$w_4^+ = w_4 - \eta * \frac{\partial Etotal}{\partial w_4}$$

$$= 0.30 - 0.5 * 0.00099542547$$

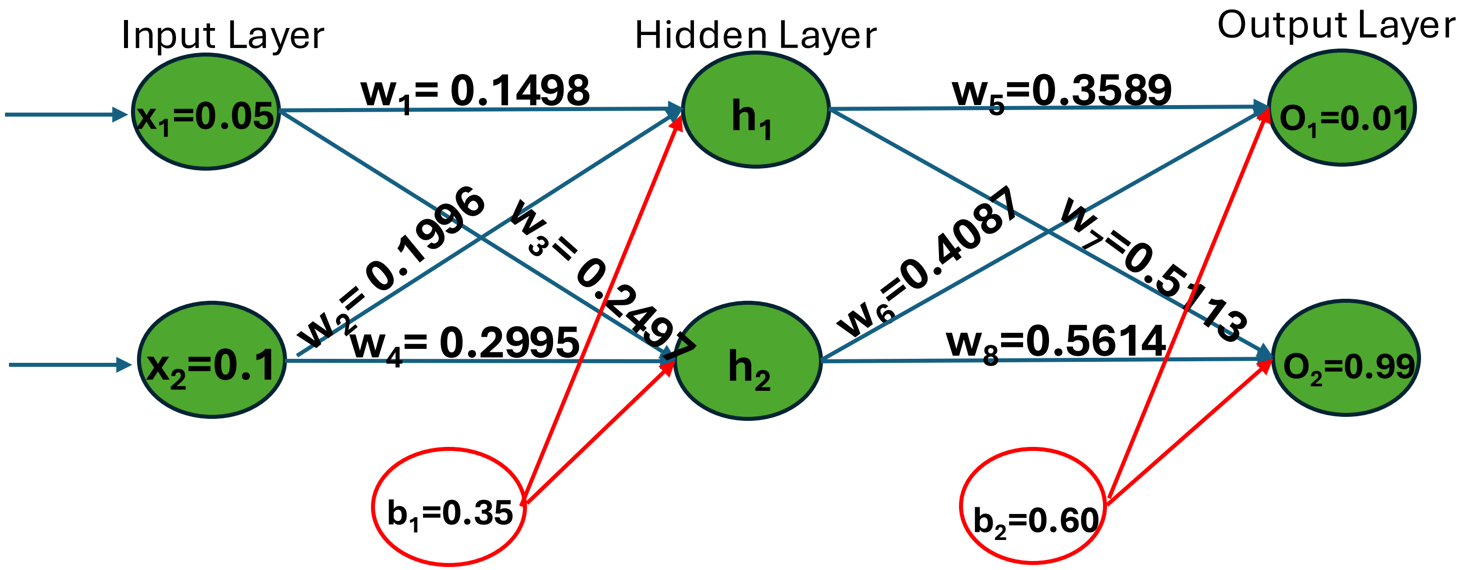
$$= 0.29950229$$

Example of Backpropagation



	Old value	New value		Old value	New value
w_1	0.15	0.149780716	w_3	0.25	0.24975114
w_2	0.20	0.19956143	w_4	0.30	0.29950229

Example of Backpropagation



- **The Backpropagation Algorithm:**
  - Finally, updating all of our weights.
  - When we fed forward the 0.05 and 0.1 inputs originally, the error on the network was 0.298371109.
  - After this first round of backpropagation, the total error is now down to 0.291027924.
  - It might not seem like much, but after repeating this process 10,000 times, for example, the error plummets to 0.0000351085.
  - At this point, when we feed forward 0.05 and 0.1, the two outputs neurons generate 0.015912196 (vs 0.01 target) and 0.984065734 (vs 0.99 target).

Thanks