Introduction to Artificial Intelligence

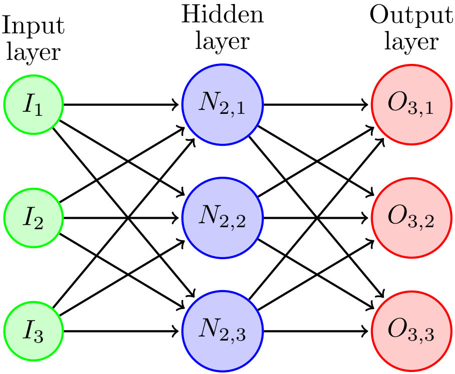
Artificial Neural Networks ANN

Ph.D. Gerardo Marx Chávez-Campos

Instituto Tecnológico de Morelia

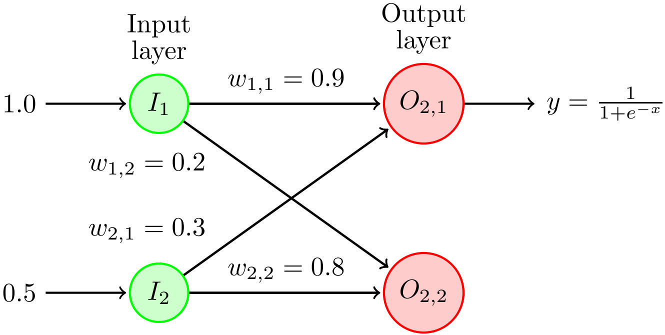
Introduction

- Inputs and outputs
- Neuron and its activation function
- Weights and bias



Signals in the Neural Network

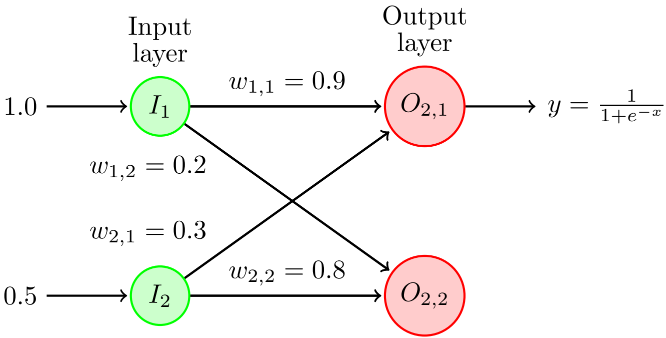
Considers the next example and compute first the output $O_{2,1}$



$$x = I_1 \cdot w_{1,1} + I_2 \cdot w_{2,1}$$

Signals in the Neural Network

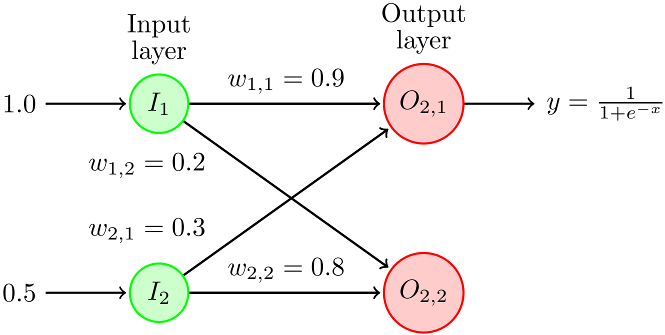
Considers the next example and compute first the output $O_{2,1}$



$$x = I_1 \cdot w_{1,1} + I_2 \cdot w_{2,1}$$
$$x = 1.0 \cdot 0.9 + 0.5 \cdot 0.3$$
$$x = 1.05$$

Signals in the Neural Network

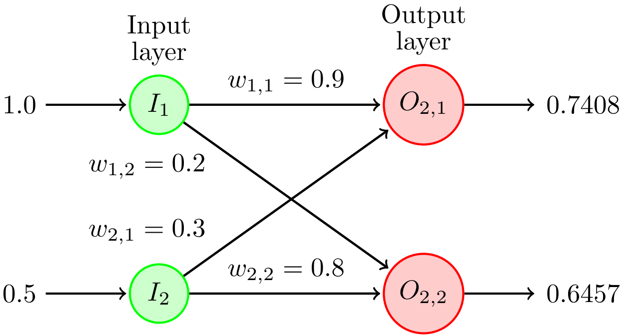
Considers the next example and compute first the output $O_{2,1}$



$$x = I_1 \cdot w_{1,1} + I_2 \cdot w_{2,1}$$
$$x = 1.0 \cdot 0.9 + 0.5 \cdot 0.3$$
$$x = 1.05$$
$$y = \frac{1}{1 + 0.3499} = 0.7407$$

Signals in the Neural Network

Results



Matrix Multiplication

Then, W is the matrix of weights, I is the matrix of inputs, and X is the resulting matrix of combined moderated signals into layer 2.

$$W \cdot I = X \tag{1}$$

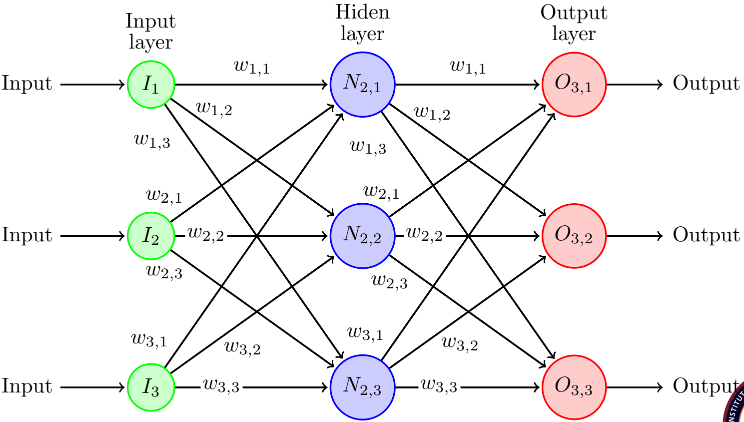
$$\begin{bmatrix} w_{1,1} & w_{2,1} \\ w_{1,2} & w_{2,2} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} (I_1 w_{1,1}) + (I_2 w_{2,1}) \\ (I_1 w_{1,2}) + (I_2 w_{2,2}) \end{bmatrix} \tag{2}$$

Finally, the output of the layer is:

$$O = \text{sigmoid}(X) \tag{3}$$

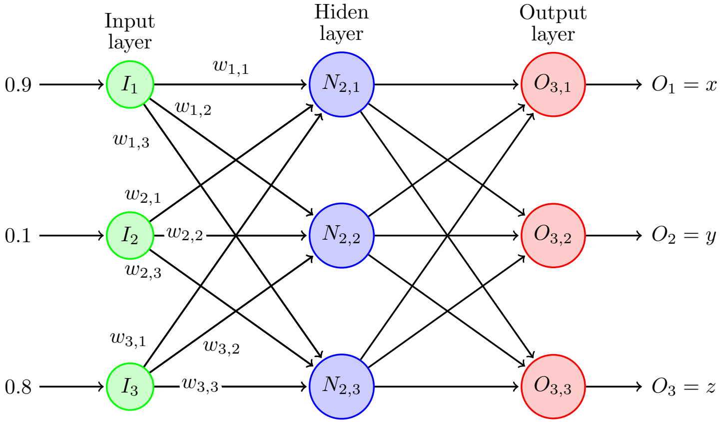
A Three Layer Matrix Multiplication

Terminology



Three layer example

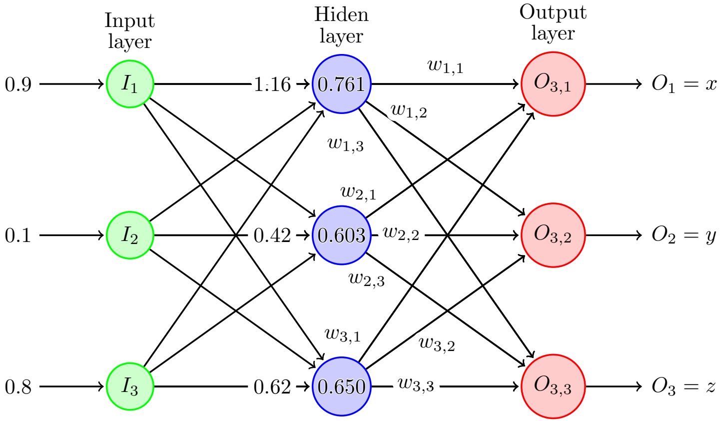
Input-Hidden Layer



$$
\begin{aligned}
w_{11} &= 0.9, & w_{12} &= 0.2, & w_{13} &= 0.1, \\
w_{21} &= 0.3, & w_{22} &= 0.8, & w_{23} &= 0.5, \\
w_{31} &= 0.4, & w_{32} &= 0.2, & w_{33} &= 0.6,
\end{aligned}
$$

Three layer example

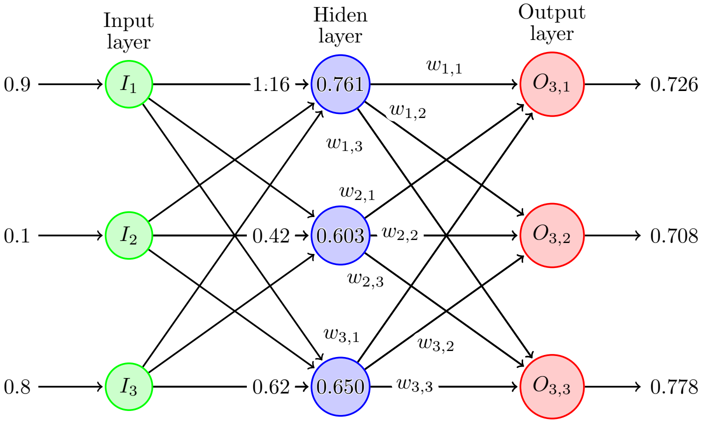
Hidden-Output Layer



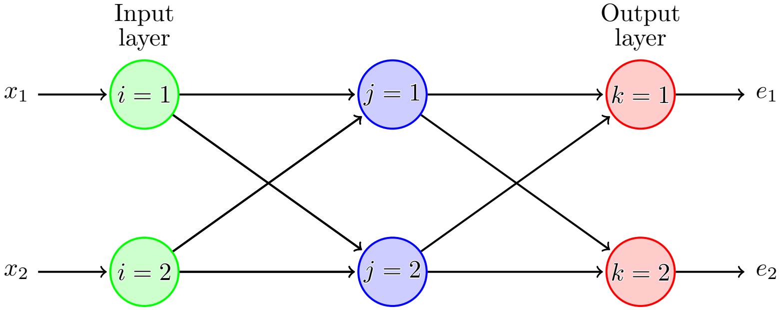
$$w_{11} = 0.3, \quad w_{12} = 0.6, \quad w_{13} = 0.8,$$
$$w_{21} = 0.7, \quad w_{22} = 0.5, \quad w_{23} = 0.2,$$
$$w_{31} = 0.5, \quad w_{32} = 0.2, \quad w_{33} = 0.9,$$

Three layer example

Resulting Output



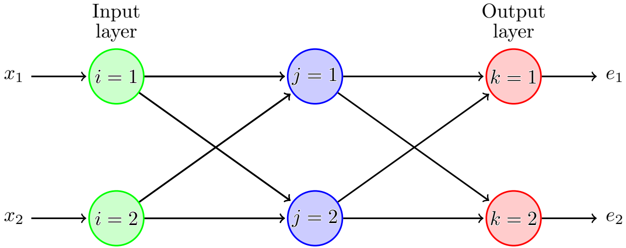
The Error



$$e_k = t_k - o_k \quad (4)$$

The Error

Backpropagation



$$e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \tag{5}$$

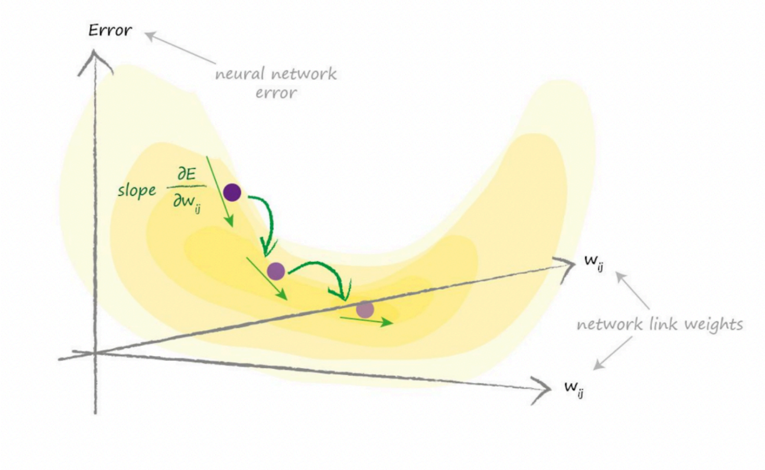
$$e_h = \begin{bmatrix} \frac{w_{11}}{w_{11}+w_{21}} & \frac{w_{12}}{w_{12}+w_{22}} \\ \frac{w_{21}}{w_{21}+w_{11}} & \frac{w_{22}}{w_{22}+w_{12}} \end{bmatrix} \tag{6}$$

$$e_h = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \tag{7}$$

$$e_h = W_{ho}^T \cdot e_{out} \tag{8}$$

The Error

Gradient concept



$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \sum_n (t_n - o_n)^2 \quad (9)$$

The Error

Gradient formula

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \sum_{n} (t_n - o_n)^2$$

- o_n only depends on the links connected to it
- o_k depends on w_{jk}

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} (t_k - o_k)^2 \quad (10)$$

The Error

Gradient formula

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}}(t_k - o_k)^2$$

- t_k is constant
- o_k depends on w_{jk}

The Error

Gradient formula: Chain Rule

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial o_k} \cdot \frac{\partial o_k}{\partial w_{jk}}$$

The Error

Gradient formula: Chain Rule

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial o_k} \cdot \frac{\partial o_k}{\partial w_{jk}}$$

$$\frac{\partial E}{\partial w_{jk}} = -2(t_k - o_k) \cdot \frac{\partial}{\partial w_{jk}} \sigma \left(\sum_j w_{jk} o_j \right)$$

o_j is the output of the previous hidden layer node; the input of the current layer!!

The Error

Gradient formula: Chain Rule

$$\frac{\partial}{\partial x}\sigma\left(x\right) = \sigma(x)\left(1 - \sigma(x)\right)$$

The Error

Gradient formula: Chain Rule

$$\frac{\partial}{\partial x}\sigma\left(x\right) = \sigma(x)\left(1 - \sigma(x)\right)$$

$$\frac{\partial E}{\partial w_{jk}} = -2(t_k - o_k) \cdot \sigma\left(\sum_j w_{jk} o_j\right)\left(1 - \sigma\left(\sum_j w_{jk} o_j\right)\right) o_j$$

The Error

Updating weights

$$\frac{\partial E}{\partial w_{jk}} = -e_j \cdot \sigma\left(\sum_i w_{ij} o_i\right)\left(1 - \sigma\left(\sum_i w_{ij} o_i\right)\right) o_i \qquad (11)$$

- e_j is the error at the output
- The σ refers to the previous layers; the hidden node j
- o_i is the output of the first layers of nodes

$$w_{jk} = w_{jk} - \alpha \frac{\partial E}{\partial w_{jk}} \qquad (12)$$

The Error

Updating weights

$$\begin{bmatrix} \Delta w_{11} & \Delta w_{12} & \Delta w_{13} & \cdots \\ \Delta w_{21} & \Delta w_{22} & \Delta w_{23} & \cdots \\ \Delta w_{31} & \Delta w_{32} & \Delta w_{jk} & \cdots \\ \dots & \dots & \dots & \dots \end{bmatrix} = \begin{bmatrix} E_1\sigma_1(1-\sigma_1) \\ E_2\sigma_2(1-\sigma_2) \\ E_k\sigma_k(1-\sigma_k) \\ \dots \end{bmatrix} \begin{bmatrix} o_1 & o_2 & o_j \cdots \end{bmatrix} \tag{13}$$

- k values from next layer
- j values from previous layer

References

Rashid, Tariq. Make your own neural network. CreateSpace Independent Publishing Platform, 2016.