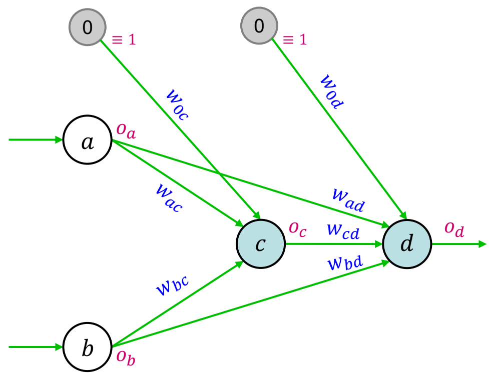
1. Consider the following network.

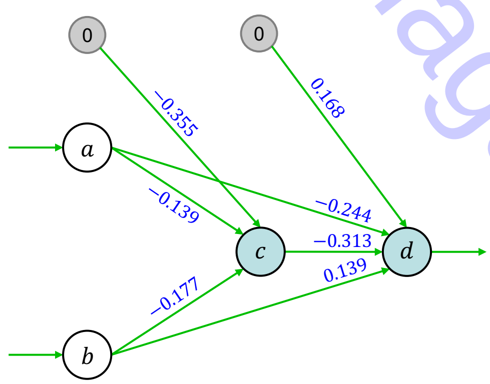


The letters in the circles represent the names of the various inputs and neurons. The net has 2 "real" neurons, c, and d, both of which use sigmoid nonlinearities. It has two inputs, a and b.

The net has a single output node, d, and a hidden node c. But note that the inputs a and b are connected to *both* the hidden and the output layer (this is sometimes called a "1.5-layer" network). Even so, backpropagation can be used to train this network.

Derive the relevant equations for the partial derivatives of the Epoch Error for this network, with respect to all the weights. Express the partials in terms of generalised errors δ_{pd} and δ_{pc}. Use squared error, i.e., $E = \sum_{p=1}^{P} 0.5(t_{pd} - o_{pd})^2$.

Use your equations to write code in Python to implement the training of this network using both pure gradient-descent and stochastic gradient-descent. Use sigmoid nonlinearities and ignore momentum. Initialise the weights as follows.



Train on binary-XOR. Does the training converge? What happens if you change the initial weights?

2. A certain single-layer network has output nodes called "quadratic neurons." The equations of such a neuron are:

$$\begin{aligned} y_k &= \sum_{j=0}^{J} w_{jk}(i_j - v_{jk})^2 \\ o_k &= f(y_k) \end{aligned}$$

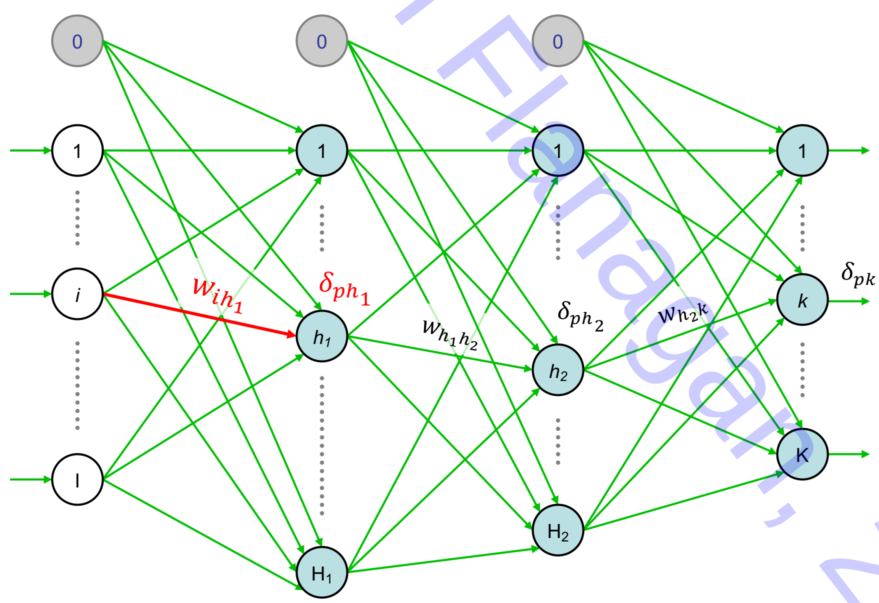
The neural function f is sigmoidal. y_k is the activation level of the k^{th} neuron, and o_k is its output. Both w_{jk} and v_{jk} are weights, and i_j is the j^{th} input value. You may assume that the w weights and the v weights are independent.

(a) Determine the weight-update equations for the w_{jk} weights.

(b) Determine the weight-update equations for the v_{jk} weights.

(c) What is the significance of this type of node? Hint: consider a single unit of the type described here, having two inputs, i_1 and i_2. How does this unit partition the input (i_1, i_2) space? You may assume that $i_0 \equiv 1$ and $v_{0k} \equiv 0$.

3. Consider a multi-layer perceptron with 2 hidden layers (3 layers of adjustable weights).



Derive an expression for $\partial E_p / \partial w_{ih_1}$ from first principles, and hence show that

$$\delta_{ph_1} = f'(y_{ph_1}) \sum_{h_2=1}^{H_2} w_{h_1h_2}\delta_{ph_2}.$$

You may assume $E_p = 0.5 \sum_{k=1}^{K} (t_{pk} - o_{pk})^2$, i.e., a squared error term.