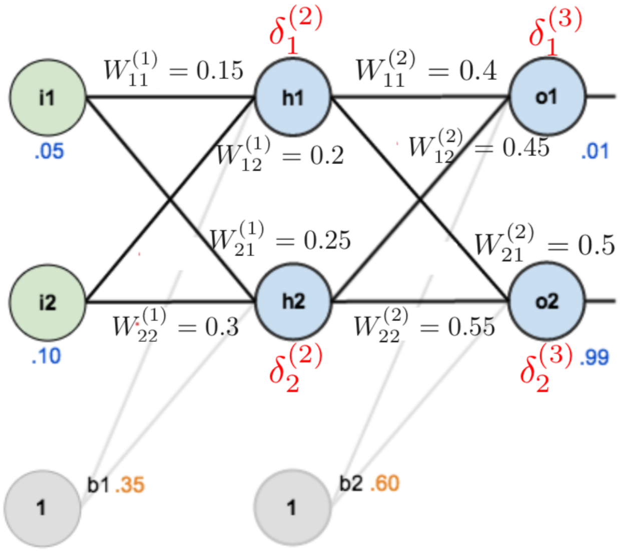
Practice Problem

Perform one iteration of forward propagation and backpropagation for the following neural network, assuming learning rate $\alpha = 0.5$ and a single training sample $\mathcal{X} = \{(i_1, i_2), (y_1, y_2)\}$, where $i_1 = 0.05$, $i_2 = 0.1$, $y_1 = 0.01$, and $y_2 = 0.99$. We also assume a sigmoid activation function $g(x) = \frac{1}{1+e^{-x}}$ for the hidden nodes, where $g'(x) = g(x)[1 - g(x)]$.



Backpropagation Algorithm

- For each node i in output layer L: $\delta_i^{(L)} = (\alpha_i^{(L)} - y_n) f'(z_i^{(L)})$
- For each (hidden) node i in layer $l = L-1, L-2, \ldots, 2$: $\delta_i^{(l)} = \left(\sum_{j=1}^{s_{l+1}} W_{ji}^{(l)} \delta_j^{(l+1)} \right) f'(z_i^{(l)})$
- Compute the desired partial derivatives as: $\frac{\vartheta J(\mathbf{W},\mathbf{b})}{\vartheta W_{ij}^{(l)}} = \alpha_j^{(l)} \delta_i^{(l+1)}$, $\frac{\vartheta J(\mathbf{W},\mathbf{b})}{\vartheta b_i^{(l)}} = \delta_i^{(l+1)}$
- Update the weights as: $W_{ij}^{(l)} := W_{ij}^{(l)} - \alpha \frac{\vartheta J(\mathbf{W},\mathbf{b})}{\vartheta W_{ij}^{(l)}}$, $b_i^{(l)} := b_i^{(l)} - \alpha \frac{\vartheta J(\mathbf{W},\mathbf{b})}{\vartheta b_i^{(l)}}$