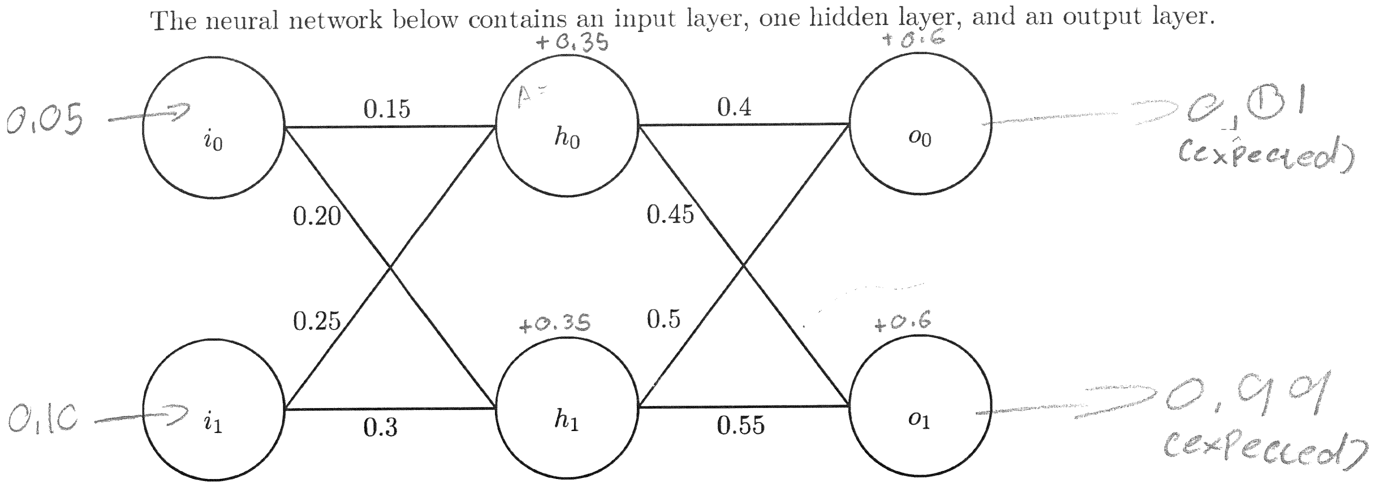
Neural Network

The neural network below contains an input layer, one hidden layer, and an output layer.



The bias for h_0 and h_1 is 0.35 and the bias for o_0 and o_1 is 0.6.

We will give the inputs 0.05 and 0.10 to i_0 and i_1 respectively, and expect outputs 0.01 and 0.99 from outputs o_0 and o_1 respectively.

1. Show that the activation value at h_0 is 0.3825.

$(i_0 \times i_0^0) + (i_1 \times i_1^0) + b_0^h = 0.3825$

$(0.05 \times 0.15) + (0.10 \times 0.25) + 0.35 = 0.3825$

2. Find the activation values for the output nodes.

$(0.05 \times 0.20) + (0.10 \times 0.3) + 0.35 = 0.39$ h_1^A

$(0.3825 \times 0.4) + (0.39 \times 0.5) + 0.6 = 0.95 - o_0$

$(0.3825 \times 0.45) + (0.39 \times 0.55) + 0.6 = 0.99 - o_1$

3. Calculate the sum of the squares of the errors.

$s^2 = \sum_{x=i} (y_i - \bar{y})^2$ $\qquad s^2 = 0.719$

$s^2 = (0.1 - 0.95)^2 + (0.99 - 0.99)^2$

4. Suppose we use scary multivariable calculus to find the the rate of change of the total error with respect to the weight of i_0^0 (which has a value of 0.15) is -0.2, and the rate of change of the total error with respect to the weight of i_1^0 is 0.9. Which should we change, and in which direction? (i.e. add or subtract). Briefly explain your answer.

Increase weight i_0^0 by learn factor → steeper gradient + gradient

Decrease weight i_1^0 by LF → less steep gradient - gradient

5. In the absence of multivariable calculus, which heuristic(s) that you have studied previously could be used to optimise your neural network.

Hillclimbing