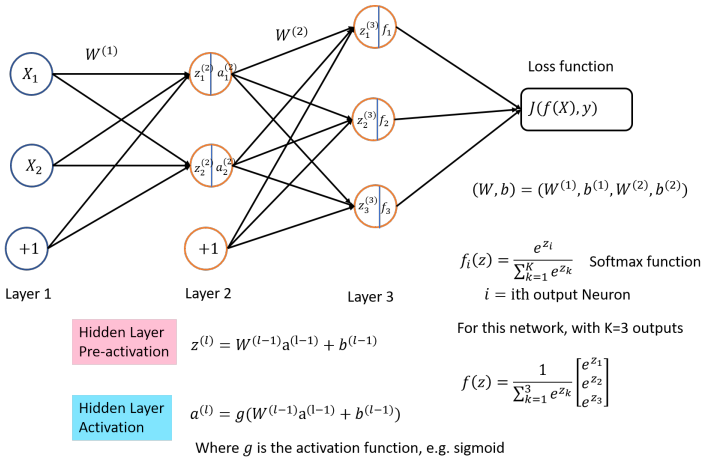
Practical to Backpropagation

Aufgabe P 3. *Backpropagation, due 30. November (in class)*

In this practical you will implement the backpropagation algorithm and apply it for following classification network for classifying three classes:



(a) Implement a function for the forward propagation on a hidden layer l and a function for evaluating the output layer with the softmax function (20 Pts)

(b) Implement a function for the delta rule on the output layer (10 Pts)

(c) Implement a function for the delta rule on a hidden layer l (10 Pts)

(d) Implement a function for computing the gradient for the weigths $W^{(l)}$ (10 Pts)

(e) Implement a function for computing the gradient for the biases $b^{(l)}$ (10 Pts)

(f) Put everything together to apply it for the given network with inputs $x = \begin{bmatrix} 0.1 \\ 0.8 \end{bmatrix}$, initial values for $W^{(1)} = \begin{bmatrix} 0.2 & 0.5 \\ 0.3 & 0.4 \end{bmatrix}$ and $b^{(1)} = \begin{bmatrix} 0.1 \\ 0.6 \end{bmatrix}$, $W^{(2)} = \begin{bmatrix} 0.1 & 0.3 \\ 0.7 & 0.4 \\ 0.01 & 0.02 \end{bmatrix}$ and $b^{(2)} = \begin{bmatrix} 0.2 \\ 0.1 \\ 0.4 \end{bmatrix}$, the correct class of the input being the third one.

i) assemble your network given in the figure and perform a forward propagation (20 Pts)

ii) use the functions defined beforehand to implement the backpropagation (20 Pts)

You should get following results for the gradients: $\nabla_{W^{(1)}} J = \begin{bmatrix} 0.00742947 & 0.05943574 \\ 0.00489816 & 0.03918524 \end{bmatrix}$, $\nabla_{b^{(1)}} J = \begin{bmatrix} 0.07429468 \\ 0.04898155 \end{bmatrix}$, $\nabla_{W^{(2)}} J = \begin{bmatrix} 0.18664481 & 0.21461037 \\ 0.26443818 & 0.30405974 \\ -0.451083 & -0.51867014 \end{bmatrix}$, $\nabla_{b^{(2)}} J = \begin{bmatrix} 0.297609 \\ 0.42165214 \\ -0.71926117 \end{bmatrix}$