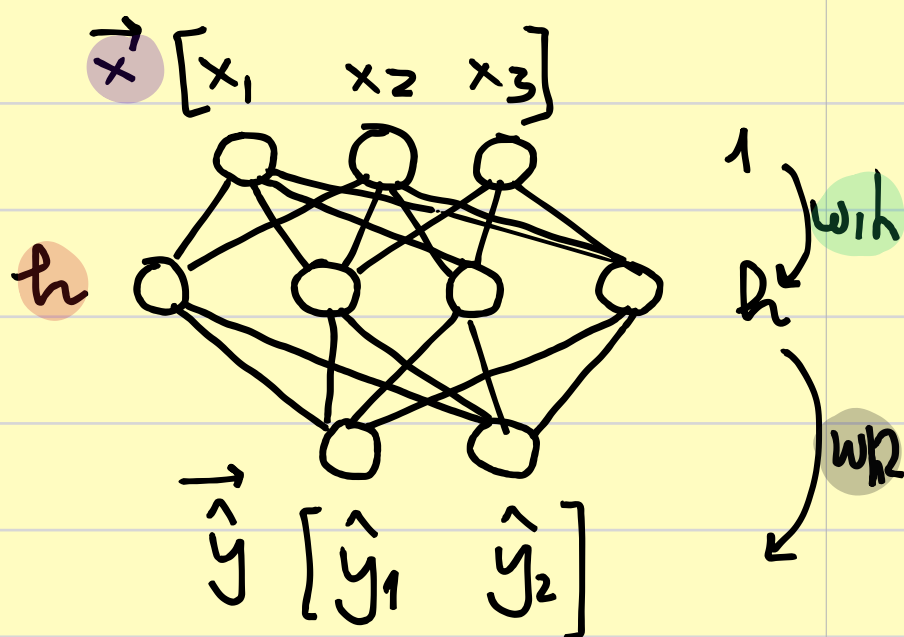
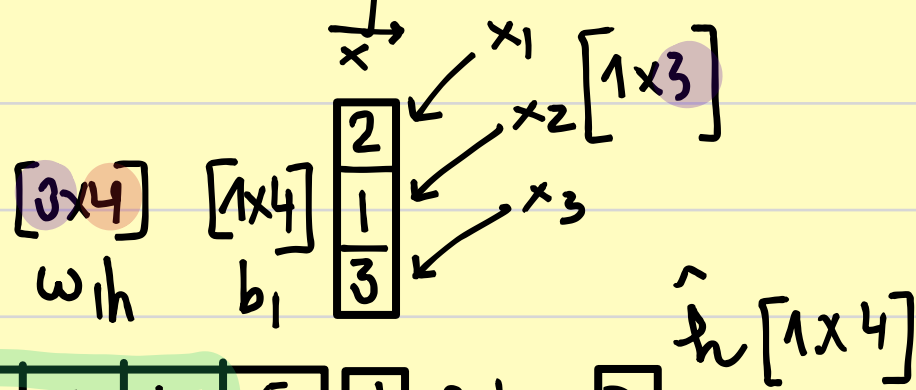


Beispiel 3. 3 layer perceptron forward pass [3,4,2]  
Hidden Layer mit 4 Neuronen.



1	-1	1	-5	-1
1	1	0	0	3
0	1	1	1	5
1	0	1	-2	3

ReLU  
 $\approx$

0
3
5
3

$\hat{h}$  [1x4]

$w_{h2}$  [4x2]

$b_2$  [1x2]

$\hat{y}$  [1x2]

1	1	-1	0	0	-2
0	0	1	-1	1	3

ReLU  
 $\approx$

0
3

$$2 \cdot 1 + 1 \cdot (-1) + 3 \cdot 1 + (-5) = -1$$

$$2 \cdot 1 + 1 \cdot 1 + 3 \cdot 0 + 0 = 3$$

$$2 \cdot 0 + 1 \cdot 1 + 3 \cdot 1 + 1 = 5$$

$$2 \cdot 1 + 1 \cdot 0 + 3 \cdot 1 + (-2) = 3$$

$$0 \cdot 1 + 3 \cdot 1 + 5 \cdot (-1) + 3 \cdot 0 + 0 = -2$$

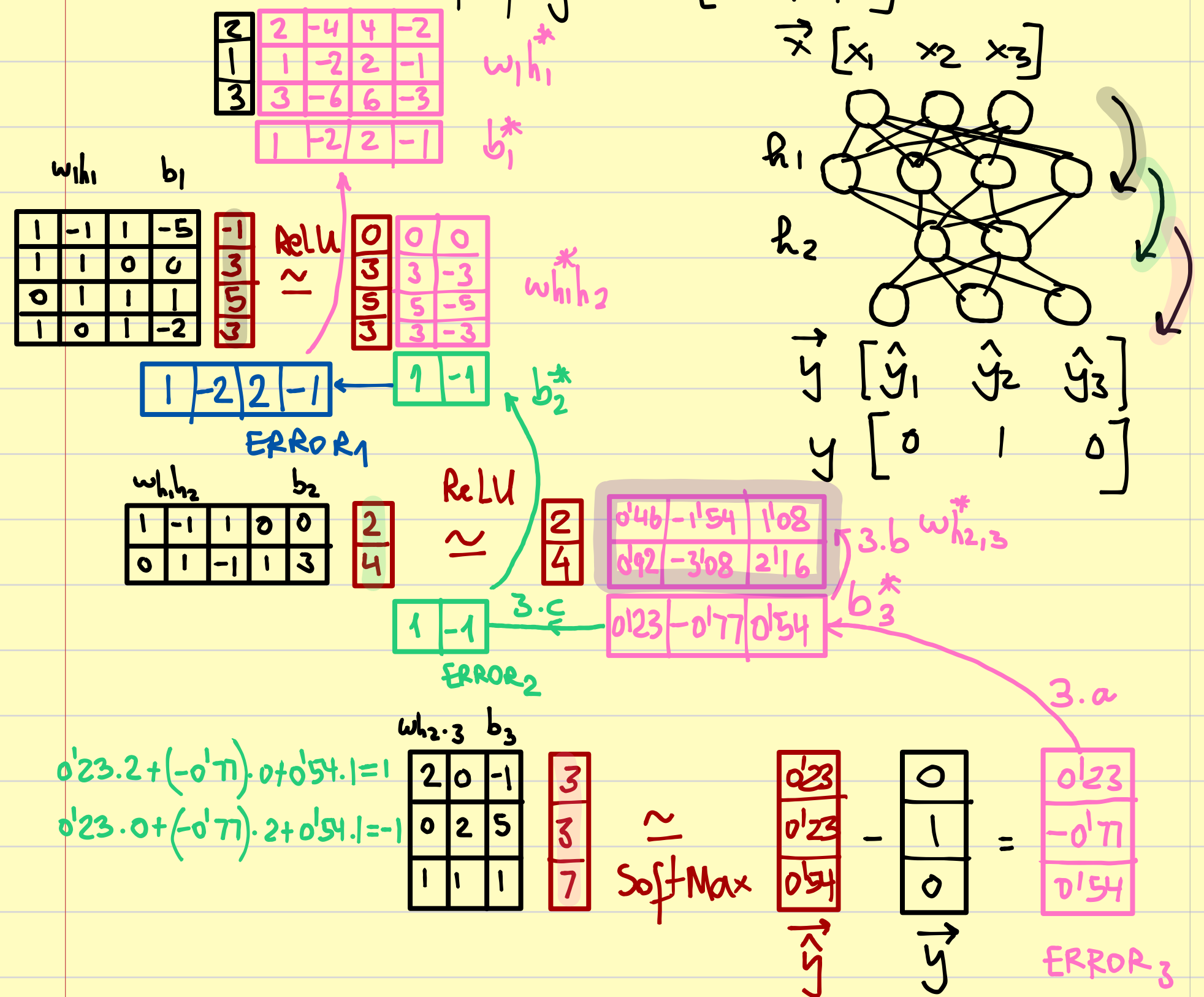
$$0 \cdot 0 + 3 \cdot 0 + 5 \cdot 1 + 3 \cdot (-1) + 1 = 3$$

Entscheidungsfunktion: [letzten Layer]

$$\text{SOFTMAX}(\vec{x}) : \vec{x} = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \rightarrow \text{SOFTMAX}(\vec{x}) = \begin{bmatrix} \frac{2}{2+1+3} \\ \frac{1}{2+1+3} \\ \frac{3}{2+1+3} \end{bmatrix} = \begin{bmatrix} 0.33 \\ 0.16 \\ 0.5 \end{bmatrix}$$

Nach der Umsetzung von Softmax, die Summe der Vektorelementen ergibt 1: wir kriegen eine Wahrscheinlichkeitsdistribution: erstes Element  $x_1$  hat eine Wahrscheinlichkeit von 0'33 einzutreffen,  $x_2$  von 0'16, und  $x_3$  von 0'5.

Beispiel 4. Perceptron mit 2 hidden layers with Feedforward & Back-propagation. [3,4,2,3]



1. Feed forward pass.

$$xh_1: 2 \cdot 1 + 1(-1) + 3 \cdot 1 + (-5) = -1$$

$$2.1 + 1.1 + 3.0 + 0 = 3$$

$$2.0 + 1.1 + 3.1 + 1 = 5$$

$$2.1 + 1.0 + 3.1 + (-2) = 3$$

$$h_1, h_2: 0.1 + 3.(-1) + 5.1 + 3.0 + 0 = \boxed{2}$$

$$0.0 + 3.1 + 5.(-1) + 3.1 + 3 = 4$$

$$h_2 y: 2.2 + 4.0 + (-1) = \boxed{3}$$

$$2.0 + 4.2 + (-5) = 3$$

$$2.1 + 4.1 + 1 = 7$$

2. SoftMax Decision Layer.

$$\text{SoftMax} \left[ \begin{bmatrix} 3 \\ 3 \\ 7 \end{bmatrix} \right] = \left[ \begin{bmatrix} \frac{3}{3+3+7} \\ \frac{3}{3+3+7} \\ \frac{7}{3+3+7} \end{bmatrix} \right] = \left[ \begin{bmatrix} \frac{3}{13} \\ \frac{3}{13} \\ \frac{7}{13} \end{bmatrix} \right] = \begin{bmatrix} 0.23 \\ 0.23 \\ 0.54 \end{bmatrix}$$

3. Back Propagation.

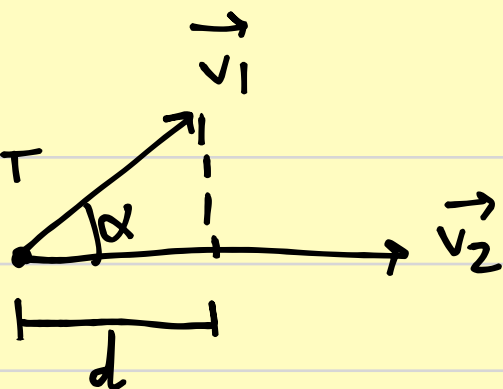
a) Transpose the error vector as new bias.

b) We DOT PRODUCT of the transposed error and the output of the previous layer.

$$\begin{aligned} w_{h_2,3}^* &: 2.0.23 = 0.46 \\ \text{DOT PRODUCT} & \quad 2.(-0.77) = -1.54 \\ & \quad 2.(0.54) = 1.08 \end{aligned}$$

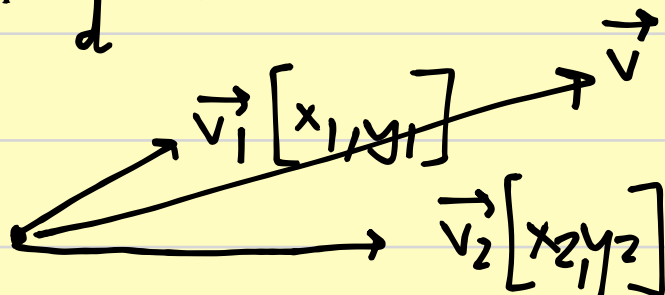
c) We multiply the transposed error with the weight matrix of the current layer.

SCALAR  
PRODUCT



$$\vec{v}_1 \cdot \vec{v}_2 = |\vec{v}_1| |\vec{v}_2| \cdot \cos \alpha = d$$

DOT  
PRODUCT



$$\vec{v}_1 \odot \vec{v}_2 = [x_1 \cdot x_2, y_1 \cdot y_2] = \vec{v}$$

\*  
 $w_{ih1}$

\*  
 $b_1$

$$\vec{x} = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}$$

2	1	3	1
-4	-2	-6	-2
4	2	6	2
2	-1	3	-1

15
-30
30
11

ReLU  
 $\sim$

15
0
30
11

\*  
 $w_{h1h2}$

\*  
 $b_2$

0	3	5	3	1
0	-3	-5	-3	-1

184
-184

ReLU  
 $\sim$

184
0

\*  
 $w_{h23}$

0'46	0'92	0'23
-1'54	-3'08	-0'77
1'08	2'16	0'54

84'87
-284'13
199'26

$$\underset{\text{Max}}{\text{Soft}} \begin{bmatrix} 0'15 \\ 0'5 \\ 0'35 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0'15 \\ -0'5 \\ 0'35 \end{bmatrix}$$

$\hat{y}^*$

~~ERROR~~  $^*_3$

$$\text{SoftMax} \begin{bmatrix} 84'87 \\ -284'13 \\ 199'26 \end{bmatrix} = \frac{1}{586'26} \begin{bmatrix} 84'87 \\ 284'13 \\ 199'26 \end{bmatrix} = \begin{bmatrix} 0'15 \\ 0'5 \\ 0'35 \end{bmatrix}$$

$$84^{\circ}87' + 284^{\circ}13' + 199^{\circ}26' = 586^{\circ}26'$$

