

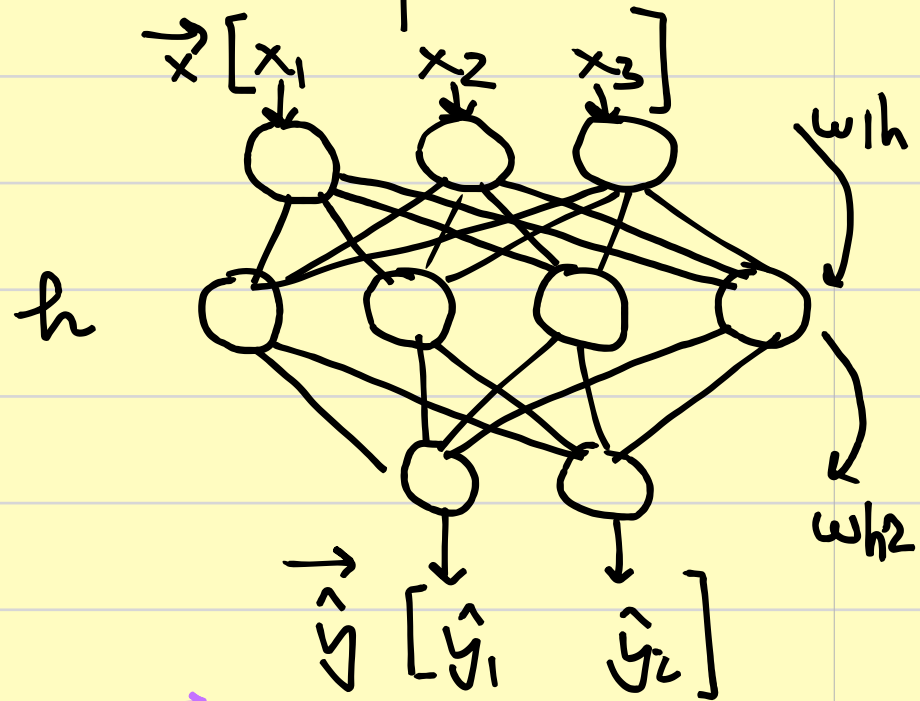
Beispiel 3. 3 Layers Perceptron mit Forward Pass

$[3, 4, 2]$: wir haben einen .. Hidden Layer mit 4 Neuronen

$[3 \times 4]$ $[1 \times 4]$
 w_{lh} b

$$\begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}$$

\vec{x}
 $\leftarrow x_1$
 $\leftarrow x_2$
 $\leftarrow x_3$



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

$$\begin{bmatrix} -1 \\ 3 \\ 5 \\ 3 \end{bmatrix}$$

ReLU
 \approx

$$\begin{bmatrix} 0 \\ 3 \\ 5 \\ 3 \end{bmatrix}$$

4×2 1×2
 w_{zh} b

1	1	-1	0	0
0	0	1	-1	1

ReLU
 \approx

$$\begin{bmatrix} -2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

$\leftarrow \hat{y}_1$
 $\leftarrow \hat{y}_2$

$$\sum x_i w_i + b \rightarrow \begin{cases} 2 \cdot 1 + 1 \cdot (-1) + 3 \cdot 1 + (-5) = -1 \\ 2 \cdot 1 + 1 \cdot 1 + 3 \cdot 0 + 0 = 3 \\ 2 \cdot 0 + 1 \cdot 1 + 3 \cdot 1 + 1 = 5 \\ 2 \cdot 1 + 1 \cdot 0 + 3 \cdot 1 + (-2) = 3 \end{cases}$$

$$\begin{cases} 0 \cdot 1 + 3 \cdot 1 + 5 \cdot (-1) + 3 \cdot 0 + 0 = -2 \\ 0 \cdot 0 + 0 \cdot 0 + 5 \cdot 1 + 3 \cdot (-1) + 1 = 3 \end{cases}$$

Beispiel 4. Feed-Forward + Back Propagation

ENTSCHEIDUNGSFUNKTION. Wird ausschließlich in dem letzten Layer angewendet, bei Backpropagation

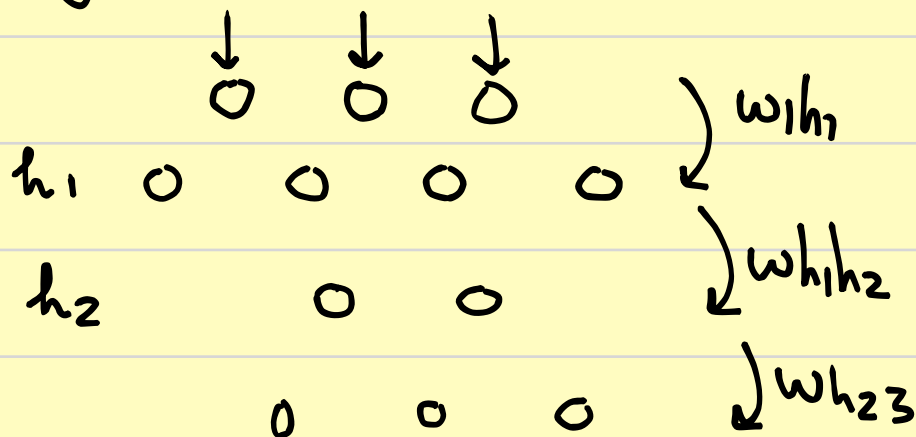
Die Entscheidungsfunktion heißt: SOFT-MAX.

SOFTMAX liefert aus dem Vektor eine Wahrscheinlichkeitsdistribution: die Summe aller Elemente ist nach der Softmax = 1.

$$\text{Beispiel: } \text{SoftMax} \left[\vec{x} = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \right] = \begin{bmatrix} \frac{2}{2+1+3} \\ \frac{1}{2+1+3} \\ \frac{3}{2+1+3} \end{bmatrix} = \begin{bmatrix} 0.33 \\ 0.16 \\ 0.5 \end{bmatrix}$$

$$\text{SoftMax} \left[\vec{x} = \begin{bmatrix} -2 \\ 1 \\ 3 \end{bmatrix} \right] = \begin{bmatrix} \frac{|-2|}{|-2|+1+3} \\ \frac{1}{|-2|+1+3} \\ \frac{3}{|-2|+1+3} \end{bmatrix} = \begin{bmatrix} \frac{2}{2+1+3} \\ \frac{1}{2+1+3} \\ \frac{3}{2+1+3} \end{bmatrix}$$

Beispiel. Perceptron mit 2 hidden layers mit feedforward und backpropagation [3, 4, 2, 3]



\vec{x}

DOTPRODUKT!

2	1'12	-2'28	2'28	-1'16
1	0'56	-1'14	1'14	-0'58
3	1'68	-3'42	3'42	-1'74
	0'56	-1'14	1'14	-0'58

$w_{1h_1}^*$

b_1^*

w_{1h_1}

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

-1
3
5
3

ReLU

\sim

0	0	0
3	1'68	-1'74
5	2'8	-2'9
3	1'68	-1'74

DOTPRODUKT!

$w_{h_1h_2}^*$

Nur die Weightmatrix

0'56	-1'14	1'14	-0'58
------	-------	------	-------

0'56	-0'58
------	-------

b_2^*

$w_{h_1h_2}$

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

2
4

ReLU

\sim

2	0'26	-0'88	0'6
4	0'52	-1'76	1'2
	0'13	-0'44	0'3

b. DOTPRODUKT!

$w_{h_2h_3}^*$

b_3^*

(Nur die Weightmatrix) ERROR

$w_{h_2h_3}$

2	0	-1
0	2	5
1	1	1

3
13
7

Soft Max

\sim

0'13
0'56
0'3

0
-1
0

a.

ERROR

$$0'56 = 0'13 \cdot 2 + (-0'44) \cdot 0 + 0'3 \cdot 1$$

$$-0'58 = 0'13 \cdot 0 + (-0'44) \cdot 2 + 0'3 \cdot 1$$

$$\text{Softmax} \begin{bmatrix} 3 \\ 13 \\ 7 \end{bmatrix} = \begin{bmatrix} \frac{3}{23} \\ \frac{13}{23} \\ \frac{7}{23} \end{bmatrix} = \begin{bmatrix} 0'13 \\ 0'56 \\ 0'3 \end{bmatrix}$$

1. Backpropagation

- a) Transponieren den ERROR Vektors als new bias
- b) Dot. Produkt vom Transponierten ERROR und OUTPUT Prev. Layer
- c) Wir multiplizieren den Transponierten ERROR mit der Gewichtmatrix des Layers

