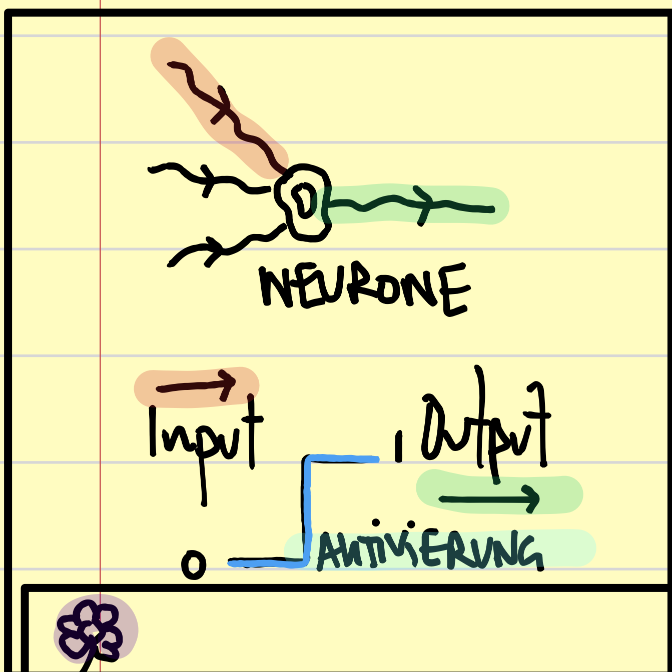
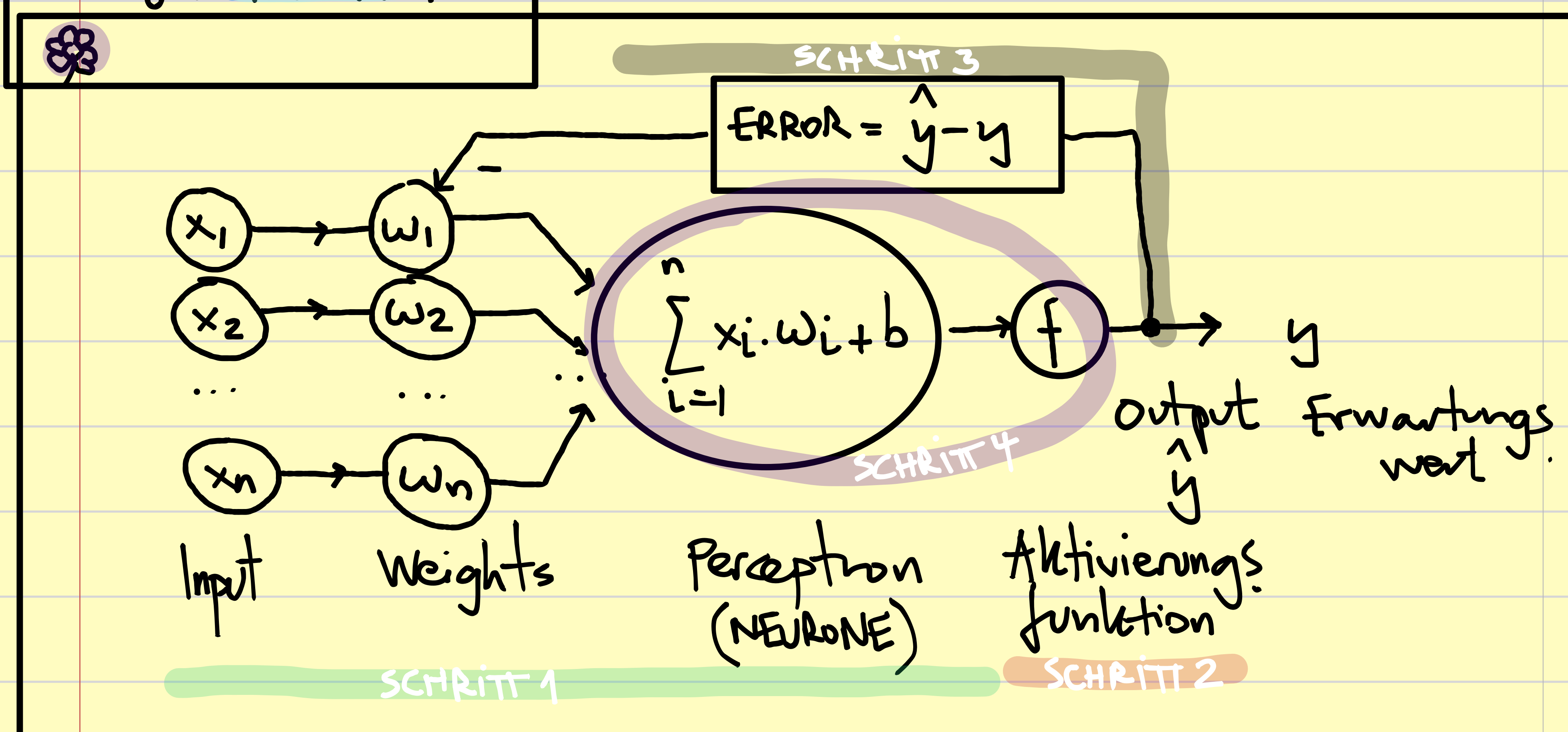
DEEP LEARNING ..by hand

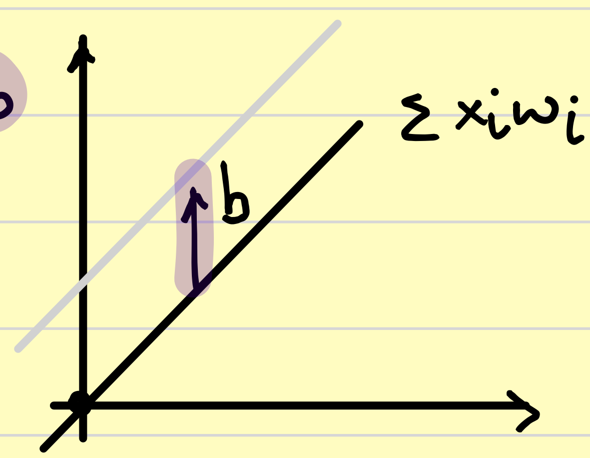


→ EINZELNEURONEN PERCEPTRON



- Schritt 1. Forward Pass. Wir multiplizieren die Inputsignale mit den Weights und addieren sie.

$$\sum_{i=1}^{n} x_i w_i + b$$

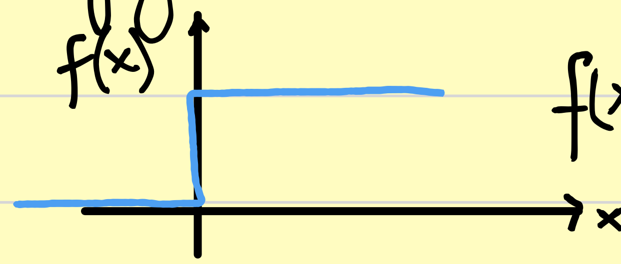


• Schritt 2. Wir setzen eine Aktivierungsfunktion ein.

$$\hat{y} = f\left(\sum_{i=1}^{n} x_i w_i + b\right)$$

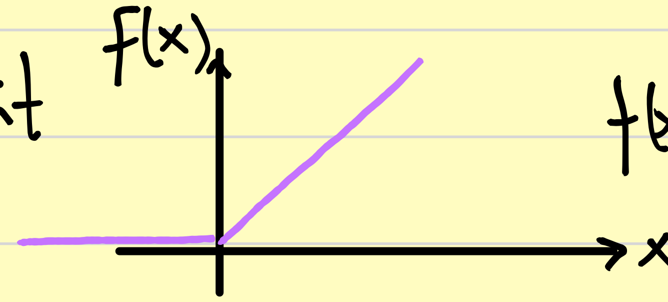
2 Beispiele für Aktivierungsfunktionen:

1) Schrittfunktion



$$f(x) = \begin{cases} 0 & x \le 0 \\ 1 & x > 0 \end{cases}$$

2) Rectifier linear Unit

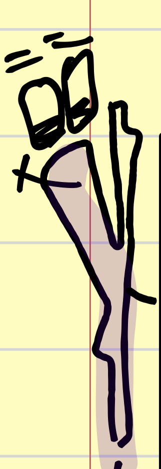


$$f(x) = \begin{cases} 0 & x \le 0 \\ x & x > 0 \end{cases}$$

• Schritt 3. Wir wollen eine Kostenfunktion (ERROR) minimieren. Damit sich die Werte nicht gegenseitig canceln (positive & negative), nehmen wir den quadratischen error:

$$C = \frac{1}{2}\left(y - \hat{y}\right)^2 = \frac{1}{2}\left(y - f\left[\sum_{i=1}^{n} x_i w_i + b\right]\right)$$

• Schritt 4. Unser Perceptron sucht die Werte von w_i um einen MINIMUM der Funktion C zu finden.



$f(x) = y$

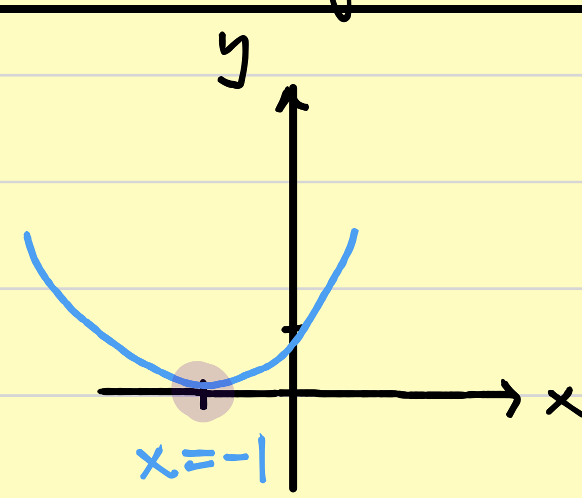
$\frac{\partial f(x)}{\partial x} = 0$

$\frac{\partial^2 f(x)}{\partial x^2} > 0$

Beispiel

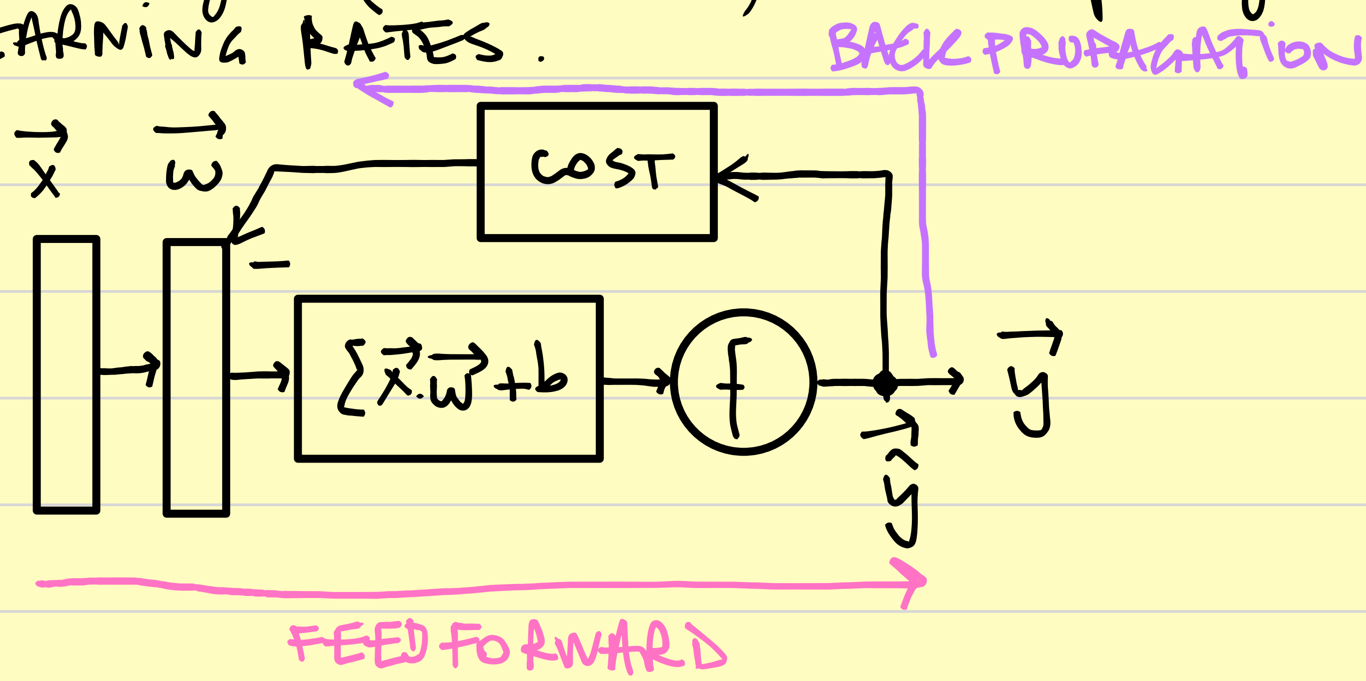
$y = x^2 + 2x + 1$

$y' = 2x + 2 = 0 \rightarrow x = -1$



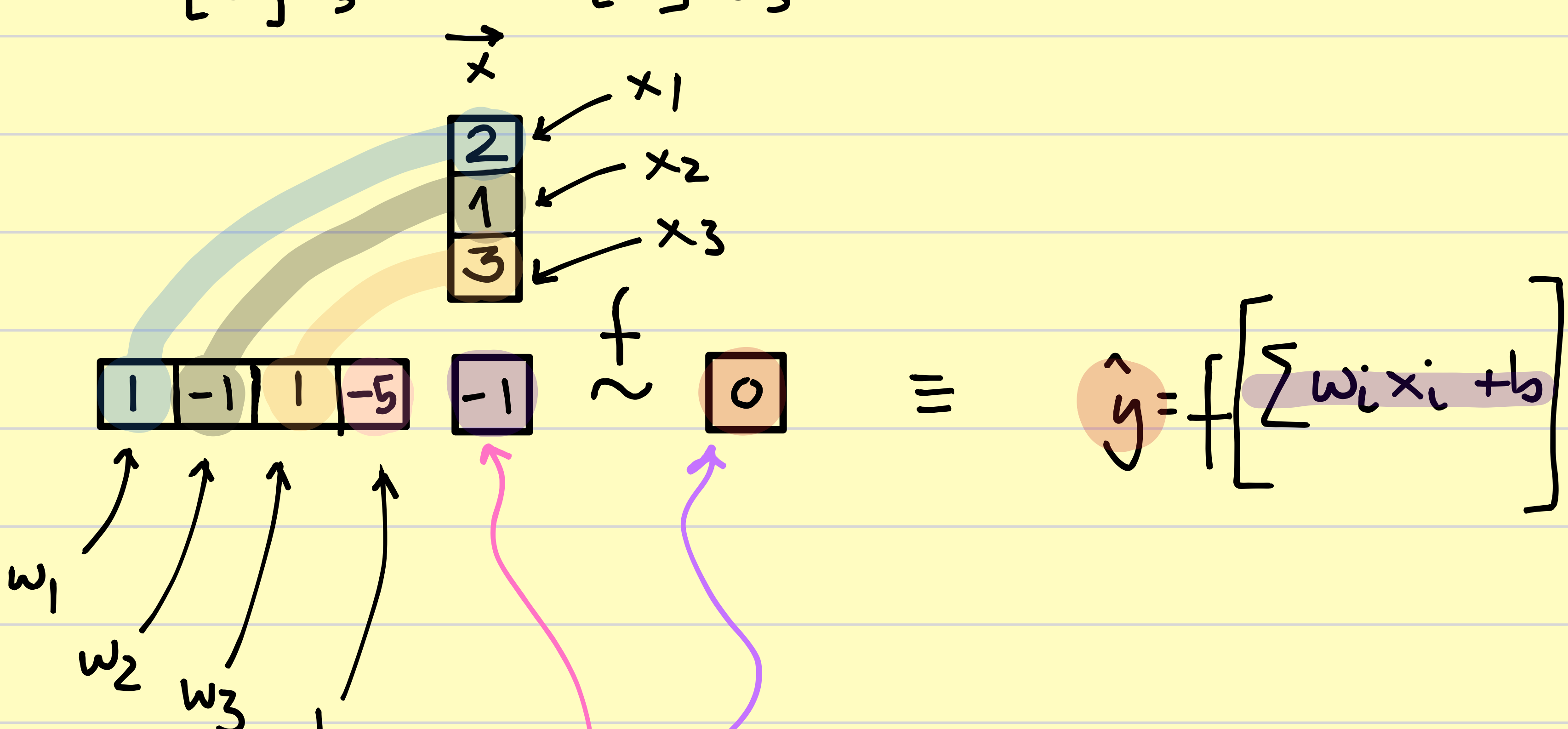
GRADIENT: $\dfrac{\partial C}{\partial w_i} = \dfrac{1}{2} \cdot 2 \cdot \left[y - w_1 x_1 - w_2 x_2 - \ldots - w_n x_n \right] (-x_i) \quad i = 1, \ldots, n$

Um das Minimum zu finden, machen wir kleine Anpassungen zu diesen Ableitungen (GRADIENTS); diese Anpassungen nennen wir LEARNING RATES.



Beispiel 1. EINZELNEURONE PERCEPTRON. FORWARDPASS.

$$\vec{x} = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \begin{matrix} x_1 \\ x_2 \\ x_3 \end{matrix} \qquad \vec{w} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \begin{matrix} w_1 \\ w_2 \\ w_3 \end{matrix} \qquad b = -5$$



$$\hat{y} = f\left[\left[\left[w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3\right] + b\right]\right] = f\left[\left[\left[1 \cdot 2 + (-1) \cdot 1 + 1 \cdot 3\right] + (-5)\right]\right]$$

$$= f\left[\left(-1\right)\right] = 0 \equiv \hat{y}$$

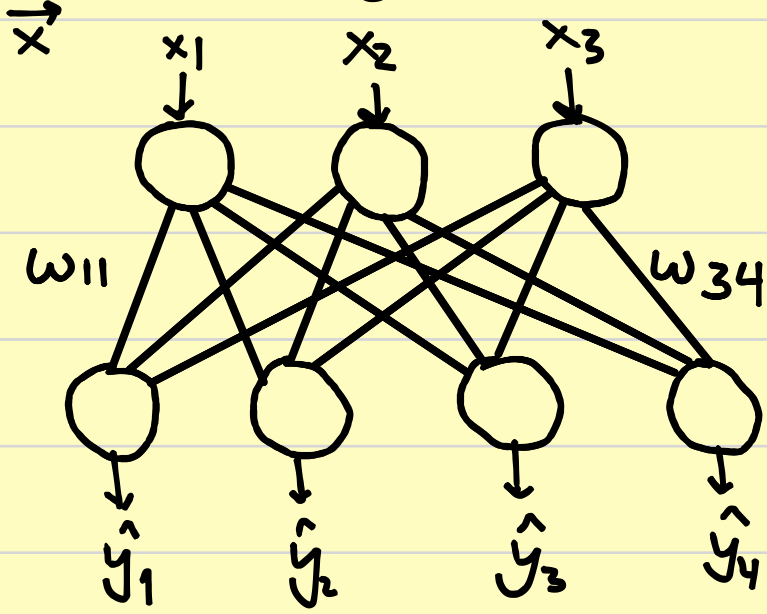
$$f \equiv \mathrm{ReLu} \equiv \begin{cases} 0 & x \leq 0 \\ x & x > 0 \end{cases}$$

Beispiel 2. Ermitteln Sie den Output des FORWARPASSes von 2 Perceptron Layers mit jeweils 3 und 4 Neuronen.

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

	w 1	2	3	$\vec{b}$
1	1	-1	1	-5
2	1	1	0	0
3	0	1	1	1
4	1	0	1	-2

$$\begin{bmatrix} -1 \\ 3 \\ 5 \\ 3 \end{bmatrix} \overset{f}{\sim} \vec{\hat{y}} = \begin{bmatrix} 0 \\ 3 \\ 5 \\ 3 \end{bmatrix} \begin{array}{l} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \hat{y}_4 \end{array}$$



$w_{11} = 1$; $w_{12} = 1$; $w_{13} = 0$; $w_{14} = 1$; ...

$$\hat{y} = f\left[\left(\sum x_i w_i\right) + b\right] = f\left[\begin{bmatrix} 2 & 1 & 3 \end{bmatrix}\begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} + \begin{bmatrix} -5 \\ 0 \\ 1 \\ -2 \end{bmatrix}\right] = f\begin{bmatrix} -1 \\ 3 \\ 5 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 5 \\ 3 \end{bmatrix}$$

$f = \mathrm{ReLU}$

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