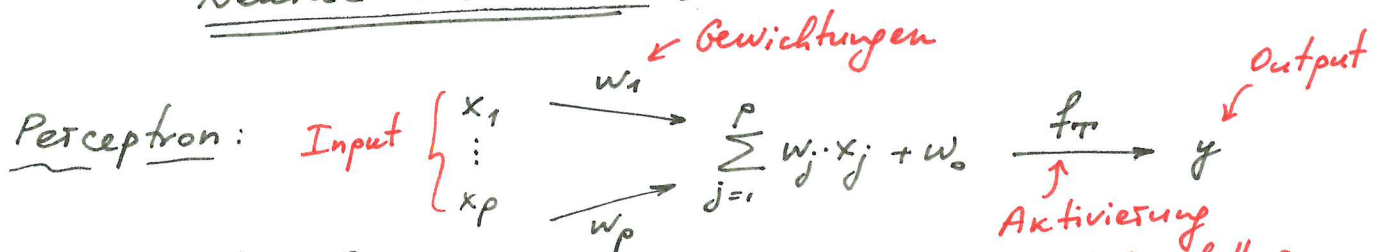


Neural Networks II



$$y = f(w^T x), \text{ wobei } x = \begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_p \end{pmatrix}, w = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{pmatrix}$$

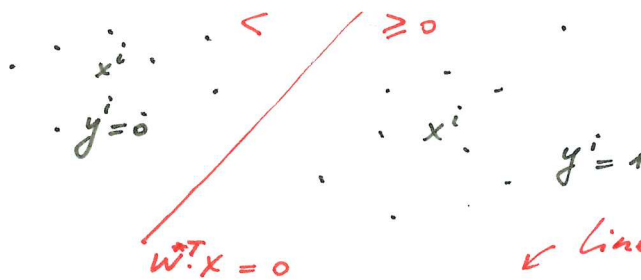
Daten: $x^i \in \mathbb{R}^{p+1}, y^i \in \{0, 1\}, i = 1, \dots, n$

(nicht stetig / diff'bar)

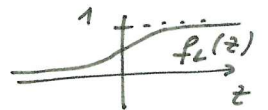


$\uparrow$ **Classification**
Annahme: linear trennbar, d. h.

$$\exists w^* \in \mathbb{R}^{p+1} \text{ mit } \begin{aligned} w^{*T} x^i &\geq 0 \Leftrightarrow y^i = 1 \\ w^{*T} x^i &< 0 \Leftrightarrow y^i = 0 \end{aligned} \quad \forall i = 1, \dots, n$$



$\nwarrow$ **Linearer Klassifizierer**



Frage: wie wird w^* erlernt?

Gradientenverfahren via logistische Regression $f_L(z) = \frac{1}{1 + e^{-z}}$

$$w(t+1) = w(t) - \underset{\substack{\uparrow \\ \text{Schritt} \\ t \rightarrow t+1}}{\alpha} \cdot \underset{\substack{\uparrow \\ \text{Schritt-} \\ \text{weite}}}{\left[f_L(w^T(t) \cdot x^i) - y^i \right]} \cdot \underset{\substack{\text{Aktivierungs-} \\ \text{wahrscheinlichkeit}}}{f_L(w^T(t) \cdot x^i)} \cdot \underset{\substack{\text{Input } i}}{x^i}$$

Rosenblatt Learning:

$$w(t+1) = w(t) - \underset{\substack{\text{Aktivierung } 1/0}}{1} \cdot \left[f_{\pi}(w^T(t) \cdot x^i) - y^i \right] \cdot x^i$$

(1) Richtige Klassifizierung: $f_{\pi}(w^T(t) \cdot x^i) = y^i = 1 \text{ oder } 0$
 $\Rightarrow w(t+1) = w(t)$ kein Update

(2) Falsche Klassifizierung: $f_{\pi}(w^T(t) \cdot x^i) \neq y^i$, d. h. $1/0$ oder $0/1$
 $\Rightarrow w(t+1) = w(t) - x^i$ (for $1/0$)
 $\Rightarrow w(t+1) = w(t) + x^i$ (for $0/1$) } Update

Konvergenzanalyse

Für alle $i=1, \dots, n$ sukzessive:

$w(1)=0$
Anfangs-
klassifizierer

falsch
klassifizieren
"1/0": $w^T(t) \cdot x^i \geq 0, y^i=0 \Rightarrow w(t+1)=w(t)-x^i$
"0/1": $w^T(t) \cdot x^i \leq 0, y^i=1 \Rightarrow w(t+1)=w(t)+x^i$

Algorithmus stoppt nach endlich vielen Schritten
mit wahrem Klassifizierer.

Dazu: o. B. d. A. seien alle Schritte mit Update bis $t+1$

Euklidische (bzw. nur Update-Schritte nachzählen)
Norm

$$\begin{aligned} \bullet \quad \|w(t+1)\|^2 &= \|w(t) \pm x^i\|^2 = \|w(t)\|^2 \pm 2 \underbrace{w^T(t) \cdot x^i}_{\leq 0 \text{ "1/0", "0/1"}} + \|x^i\|^2 \leq M \quad \forall i \\ &\leq \|w(t)\|^2 + M \leq \underbrace{\|w(1)\|^2}_{=0} + t \cdot M = t \cdot M \end{aligned}$$

$$\begin{aligned} \bullet \quad \underbrace{w^{*T}}_{\text{linearer Klassifizierer}} \cdot w(t+1) &= w^{*T} (w(t) \pm x^i) = \underbrace{w^{*T} \cdot w(t)}_{\text{positiv} \geq m > 0 \quad \forall i} \pm w^{*T} \cdot x^i \\ &\geq w^{*T} \cdot w(t) + m \\ &\geq w^{*T} \cdot \underbrace{w(1)}_{=0} + t \cdot m = t \cdot m \end{aligned}$$

$$\begin{aligned} - \text{"1/0": } y^i=0 &\Rightarrow w^{*T} \cdot x^i < 0 \\ + \text{"0/1": } y^i=1 &\Rightarrow w^{*T} \cdot x^i > 0 \end{aligned}$$

$$t \cdot m \leq w^{*T} \cdot w(t+1) \leq \|w^*\| \cdot \|w(t+1)\| \leq \|w^*\| \cdot \sqrt{t \cdot M} \Rightarrow \boxed{t \leq \frac{\|w^*\|^2 \cdot M}{m^2}}$$

Cauchy-
Schwarz

Anzahl der echten
Update-Schritte ist
begrenzt!

Beispiel: XOR - Problem

$$\begin{matrix} x_1 & \xrightarrow{w_1} \\ x_2 & \xrightarrow{w_2} \end{matrix} \quad w_1 \cdot x_1 + w_2 \cdot x_2 + w_0 \xrightarrow{f_T} y$$

Daten:

x^i	0 0	1 0	0 1	1 1
y^i	0	1	1	0

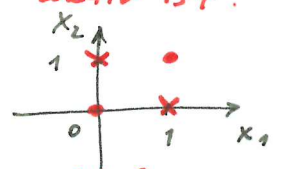
"exklusives Oder"

einschalten,
nur wenn
genau eines
von beiden
aktiv ist.

Angenommen $f_T(x^i \cdot w_1 + x^i \cdot w_2 + w_0) = y^i \quad \forall i=1,2,3,4$

$$\begin{aligned} \bullet \quad f_T(0 \cdot w_1 + 0 \cdot w_2 + w_0) &= 0 \Rightarrow w_0 < 0 \\ \bullet \quad f_T(1 \cdot w_1 + 0 \cdot w_2 + w_0) &= 1 \Rightarrow w_1 + w_0 \geq 0 \\ \bullet \quad f_T(0 \cdot w_1 + 1 \cdot w_2 + w_0) &= 1 \Rightarrow w_2 + w_0 \geq 0 \\ \bullet \quad f_T(1 \cdot w_1 + 1 \cdot w_2 + w_0) &= 0 \Rightarrow w_1 + w_2 + w_0 < 0 \end{aligned}$$

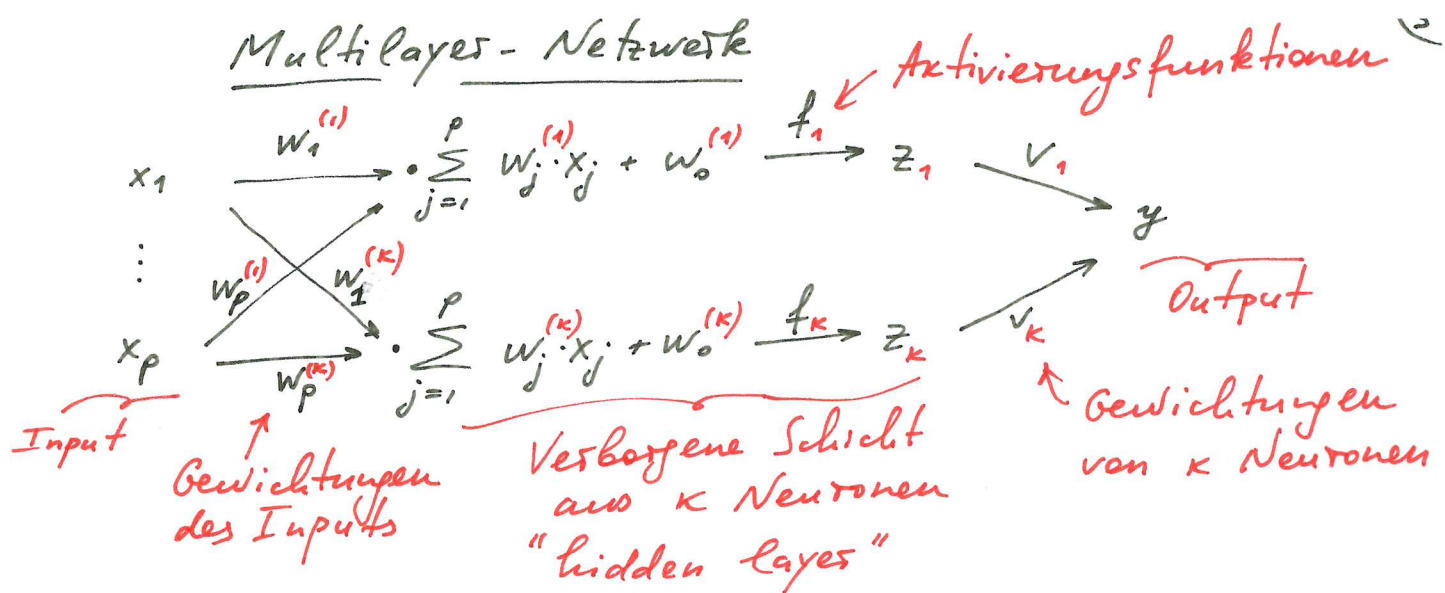
$$\begin{aligned} \Rightarrow w_1 \geq -w_0 > 0 \\ \Rightarrow 0 > w_1 > 0 \\ \Rightarrow w_1 < -(w_2 + w_0) \leq 0 \end{aligned}$$



nicht linear
separierbar/
trennbar

→ multilayer networks

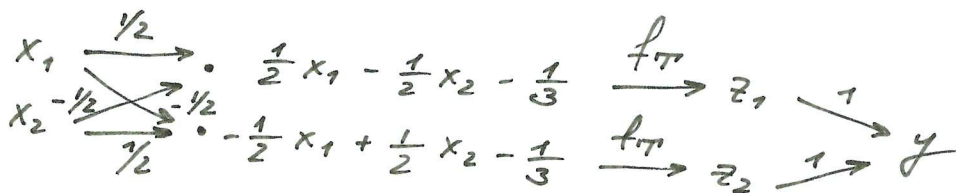
MultiLayer-Netzwerk



$$y = \sum_{e=1}^k v_e \cdot z_e = \sum_{e=1}^k v_e \cdot f_e \left(\sum_{j=1}^p w_j^{(e)} x_j + w_0^{(e)} \right) = \sum_{e=1}^k v_e f_e(w^{(e)T} x)$$

zu adjustierende Gewichtungen
 $v \in \mathbb{R}^k, w^{(e)} \in \mathbb{R}^{p+1}, e=1, \dots, k$

Lösung des XOR-Problems:



$$y = z_1 + z_2 = f_{\pi} \left(\frac{1}{2} x_1 - \frac{1}{2} x_2 - \frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{2} x_1 + \frac{1}{2} x_2 - \frac{1}{3} \right)$$

Datenabgleich:

x^i	$0 \ 0$	$\rightarrow f_{\pi} \left(\frac{1}{2} \cdot 0 - \frac{1}{2} \cdot 0 - \frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 0 - \frac{1}{3} \right) = f_{\pi} \left(-\frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{3} \right) = 0$
	$1 \ 0$	$\rightarrow f_{\pi} \left(\frac{1}{2} \cdot 1 - \frac{1}{2} \cdot 0 - \frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 0 - \frac{1}{3} \right) = f_{\pi} \left(\frac{1}{6} \right) + f_{\pi} \left(-\frac{5}{6} \right) = 1$
	$0 \ 1$	$\rightarrow f_{\pi} \left(\frac{1}{2} \cdot 0 - \frac{1}{2} \cdot 1 - \frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 - \frac{1}{3} \right) = f_{\pi} \left(-\frac{5}{6} \right) + f_{\pi} \left(\frac{1}{6} \right) = 1$
	$1 \ 1$	$\rightarrow f_{\pi} \left(\frac{1}{2} \cdot 1 - \frac{1}{2} \cdot 1 - \frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 - \frac{1}{3} \right) = f_{\pi} \left(-\frac{1}{3} \right) + f_{\pi} \left(-\frac{1}{3} \right) = 0$

Approximationsresultat im Allgemeinen:

$\forall F: K \rightarrow \mathbb{R}$ und $\varepsilon > 0 \exists v \in \mathbb{R}^k, w^{(e)} \in \mathbb{R}^{p+1}, e=1, \dots, k$
 ↑ kompakte Teilmenge $\subseteq \mathbb{R}^p$
 stetig auf K

mit

$$\left| \underbrace{F(x)}_{=y} - \sum_{e=1}^k v_e \cdot \underbrace{f_{\pi}}_{\text{Threshold-Funktion}} \left(w^{(e)T} x \right) \right| \leq \varepsilon \quad \forall x \in K$$

"Modell"

Fazit: Neuronales Netz mit einer Zwischenschicht kann beliebig gut jedes stetige Modell simulieren!