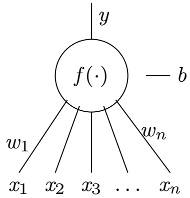
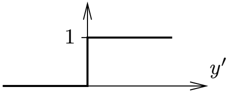
Mustererkennung
Thema 1: Neuron

D. Schlesinger – TUD/INF/KI/IS

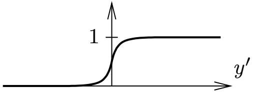
Neuron



Input $x \in \mathbb{R}^n$, Gewichte $w \in \mathbb{R}^n$, Schwellwert $b \in \mathbb{R}$,
Aktivierung $y' = \sum_i w_i x_i = \langle w, x\rangle$,
Output $y = f(y' - b) = f(\langle w, x\rangle - b)$



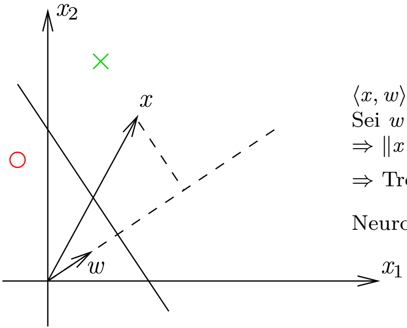
$$f(y') = \begin{cases} 1 & \text{wenn } y' > 0 \\ 0 & \text{sonst} \end{cases}$$



$$f(y') = \frac{1}{1+\exp(-y')}$$

Kurz (Swellwertneuron): $\langle x, w\rangle \lessgtr b$

Geometrische Interpretation



$\langle x, w\rangle = \|x\| \cdot \|w\| \cdot \cos\phi$
Sei w normiert, d.h. $\|w\| = 1$
$\Rightarrow \|x\| \cdot \cos\phi$ – Länge der Projektion von x auf w

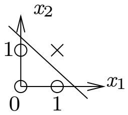
$\Rightarrow$ Trennebene $\langle x, w\rangle = const$

Neuron realisiert einen *linearen Klassifikator*

Spezialfall – Boolsche Funktionen

Input: $x = (x_1, x_2),\ x_i \in \{0, 1\}$.
Gesucht ist das Neuron (w und b), dass $y = x_1 \& x_2$ realisiert.

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1



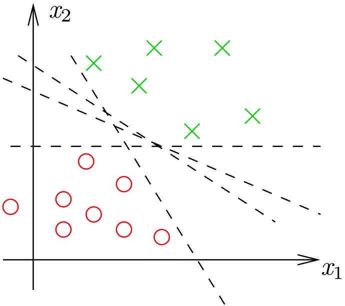
$$w_1 = w_2 = 1,\ b = 1.5$$

ODER, Andere boolsche Funktionen, XOR geht nicht!!!

Gegeben: Lernstichprobe $\Big((x^1,y^1),(x^2,y^2),\ldots,(x^L,y^L)\Big)$, $x^l\in\mathbb{R}^n$, $y^l\in\{0,1\}$
Gesucht: $w\in\mathbb{R}^n$, $b\in\mathbb{R}$ so dass $f(\langle x^l,w\rangle-b)=y^l$ für alle $l=1,\ldots,L$

Für einen Schwellwertneuron:

$$\begin{cases}\langle x^l,w\rangle>b & \text{wenn } y^l=1,\\ \langle x^l,w\rangle<b & \text{wenn } y^l=0.\end{cases}$$

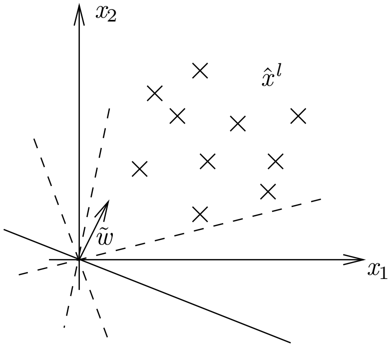


Vorbereitung 1:
w und b zu einem Parametervektor $\tilde{w}$:
$x = (x_1, x_2, \dots, x_n) \Rightarrow$
$\tilde{x} = (x_1, x_2, \dots, x_n, 1)$
$w = (w_1, w_2, \dots, w_n) \Rightarrow$
$\tilde{w} = (w_1, w_2, \dots, w_n, -b)$
$\Downarrow$
$\langle x^l, w \rangle \gtrless b \Rightarrow \langle \tilde{x}^l, \tilde{w} \rangle \gtrless 0$

Vorbereitung 2:
alles zum System linearer Ungleichungen:
$\hat{x}^l = \tilde{x}^l$ für l mit $y^l = 1$
$\hat{x}^l = -\tilde{x}^l$ für l mit $y^l = 0$
$\Downarrow$

$$\begin{cases} \langle \tilde{x}^l, \tilde{w} \rangle > 0 & \text{wenn} \quad y^l = 1 \\ \langle \tilde{x}^l, \tilde{w} \rangle < 0 & \text{wenn} \quad y^l = 0 \end{cases}$$

$\Downarrow$
$\langle \hat{x}^l, \tilde{w} \rangle > 0 \quad \forall l$

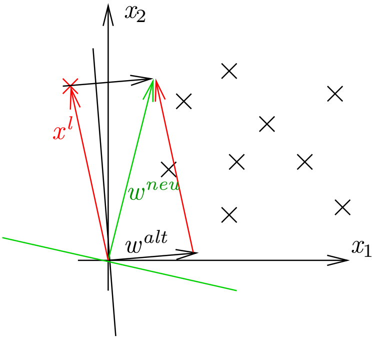


Perzeptron Algorithmus

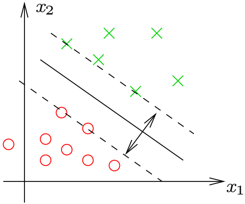
Algorithmus zur Lösung des Systems linearer Ungleichungen
$\langle x^l, w\rangle > 0$ für alle $l = 1, \ldots, L$.

1 Suche eine noch nicht erfüllte Gleichung, d.h. ein l so dass $\langle x^l, w\rangle < 0$ gilt;

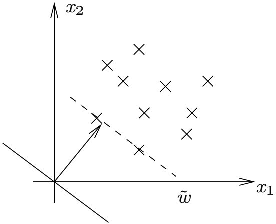
2 Wenn nich gefunden – Ende,
sonst, aktualisiere $w^{neu} = w^{alt} + x^l$, gehe zu 1.



Eigenschaften:
Der Algorithmus terminiert, wenn eine Lösung existiert. Wenn keine Lösung existiert, hält er nie an.
Die Lösung ist (bis auf eine Skalierung und unter Umsänden) ein Punk in der konvexen Hülle der Lernstichprobe



Man suche nach einem *„Streifen" maximaler Breite*, der die Lernstichprobe separiert.

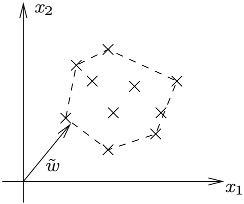


Nach „Vorbereitung 1" und „Vorbereitung 2":

$$\min_l \langle x^l, w \rangle \rightarrow \max_w$$

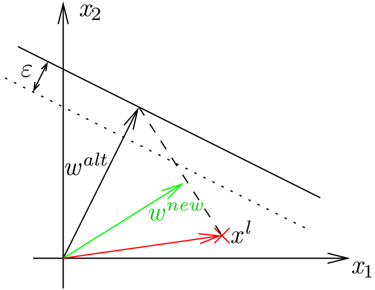
Vergleiche mit Perceptron

$$\min_l \langle x^l, w \rangle > 0$$



Äquivalent zu:

$$\|w\| \rightarrow \min$$
$$w \in conv(x^l)$$



Terminiert nicht unbedingt bei $\varepsilon = 0$.

ε-genauer Algorithmus:

1 Suche ein x^l so dass $\langle x^l, w\rangle < \|w\| - \varepsilon$ gilt;

2 Wenn nich gefunden – Ende.

3 Suche $\gamma^* = \arg\min_\gamma \|w^{alt} + \gamma(x^l - w^{alt})\|^2$, $\gamma \geq 0$,

aktualisiere $w^{neu} = w^{alt} + \gamma(x^l - w^{alt})$,

gehe zu 1.