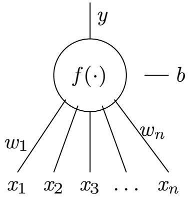
Mustererkennung: Neuron

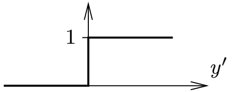
Neuron

Input $x \in \mathbb{R}^n$, Gewichte $w \in \mathbb{R}^n$, Schwellwert $b \in \mathbb{R}$,

Aktivierung $y' = \sum_i w_i x_i = \langle w, x \rangle$,

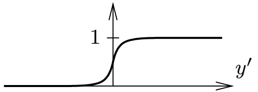
Output $y = f(y' - b) = f(\langle w, x \rangle - b)$





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

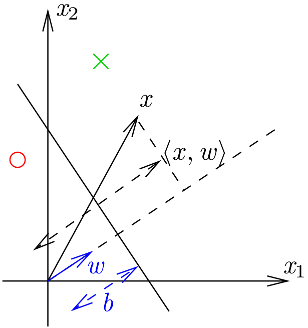
Step-Funktion



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

Sigmoid-Funktion (differenzierbar)

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



$\langle x, w\rangle = \|x\| \cdot \|w\| \cdot \cos\phi$

Sei w normiert, d.h. $\|w\| = 1$

$\Rightarrow$ $\|x\| \cdot \cos\phi$ – Länge der Projektion von x auf w

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

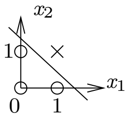
Neuron realisiert einen *linearen Klassifikator*

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



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

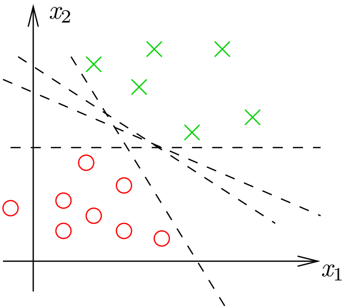
ODER, Andere boolsche Funktionen, XOR geht nicht!!!

Lernaufgabe

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 – System linearer Ungleichungen:

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



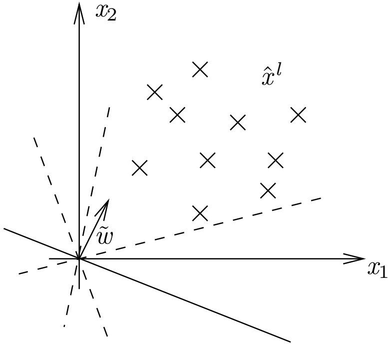
Es gibt (im Allgemeinen) mehrere Lösungen!!!

Vorbereitung 1:
w und b zu einem Parametervektor $\tilde{w}$:

$x = (x_1, x_2, \ldots, x_n) \Rightarrow$
$\tilde{x} = (x_1, x_2, \ldots, x_n, 1)$
$w = (w_1, w_2, \ldots, w_n) \Rightarrow$
$\tilde{w} = (w_1, w_2, \ldots, w_n, -b)$
$\Downarrow$
$\langle x^l, w \rangle \gtrless b \Rightarrow \langle \tilde{x}^l, \tilde{w} \rangle \gtrless 0$

Vorbereitung 2:
alles zum einheitlichen System:

$\hat{x}^l = \tilde{x}^l$ für l mit $y^l = 1$
$\hat{x}^l = -\tilde{x}^l$ für l mit $y^l = 0$



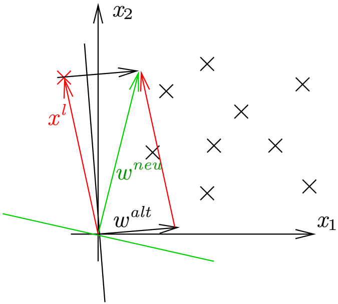
$$\begin{cases} \langle x^l, w \rangle > b & \text{wenn } \; y^l = 1 \\ \langle x^l, w \rangle < b & \text{wenn } \; y^l = 0 \end{cases} \quad \Rightarrow \quad \langle \hat{x}^l, \tilde{w} \rangle > 0 \quad \forall l$$

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 $\langle x^l, w \rangle \leq 0$ gilt;
2) Wenn nicht gefunden – Ende,

   sonst, aktualisiere $w^{neu} = w^{alt} + x^l$, gehe zu 1).



– Der Algorithmus terminiert, wenn eine Lösung existiert.
  Wenn keine Lösung existiert, hält er nie an.

– Die Lösung ist (bis auf eine Skalierung und unter Umständen) ein Punk in der konvexen Hülle der Lernstichprobe

Beweis der Konvergenz

$$\|w^{(n+1)}\|^2 = \|w^{(n)} + x^i\|^2 = \|w^{(n)}\|^2 + 2\langle w^{(n)}, x^i\rangle + \|x^i\|^2 \leq \|w^{(n)}\|^2 + D^2$$
$$\Rightarrow \|w^{(n)}\| \leq \sqrt{n}D \quad \text{weil} \quad \langle w^{(n)}, x^i\rangle \leq 0 \quad \text{(nicht erfüllt)}$$

$$\frac{\langle w^{(n+1)}, w^*\rangle}{\|w^*\|} = \frac{\langle w^{(n)}, w^*\rangle}{\|w^*\|} + \frac{\langle x^i, w^*\rangle}{\|w^*\|} \geq \frac{\langle w^{(n)}, w^*\rangle}{\|w^*\|} + \epsilon$$
$$\Rightarrow \frac{\langle w^{(n)}, w^*\rangle}{\|w^*\|} \geq n\epsilon \quad \text{weil} \quad \langle x^i, w^*\rangle > 0 \quad \text{(erfüllt)}$$

$$1 \geq \frac{\langle w^{(n)}, w^*\rangle}{\|w^*\| \cdot \|w^{(n)}\|} \geq \sqrt{n}\frac{\epsilon}{D} \quad \text{wegen Cauchy-Schwarz Ungleichung}$$
$$\Rightarrow n \leq \frac{D^2}{\epsilon^2}$$

Wenn eine Lösung w^* existiert,
konvergiert der Algorithmus nach höchstens D^2/ϵ^2 Schritten.

$D = \max_l \|x^l\|, \quad \epsilon = \min_l \langle x^l, w^*\rangle / \|w^*\|$ – der Margin.

Entscheidungsregel für eine reelwertige Größe $x \in \mathbb{R}$ sei ein Polynom k-tes Grades, d.h.

$$a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 = \sum_i a_i x^i \gtrless 0$$

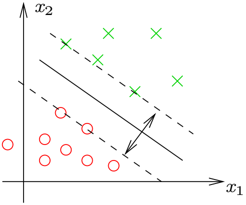
Man lerne die unbekannten Koeffizienten a_i des Polynoms
anhand einer klassifizierten Lernstichprobe $\big((x^l, y^l) \ldots\big)$, $x^l \in \mathbb{R}$, $y^l \in \{0, 1\}$.

Man überführe die Aufgabe in eine Perzeptron-Aufgabe.
Obwohl die Entscheidungsregel bezüglich x nicht mehr linear ist,
ist sie immer noch linear bezüglich der Parameter a_i
$\Rightarrow$ System linearer Ungleichungen

$w = (a_n, a_{n-1}, \ldots, a_1, a_0)$
$\tilde{x} = (x^n, x^{n-1}, \ldots, x, 1)$ – und das für jedes l (ein $n+1$-dimensionaler Vektor)

$\sum_i a_i x^i = \langle \tilde{x}, w \rangle \quad \Rightarrow$ Perzeptron Aufgabe.

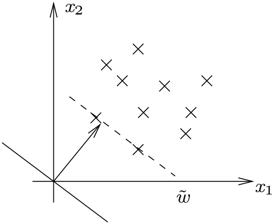
Allgemein: durch eine geeignete Transformation des Raums lassen sich viele nicht-lineare Entscheidungsregel mit dem Perzeptron Algorithmus lernen.



Es existieren mehrere Lösungen

$\Downarrow$

Man suche nach einem
„Streifen“ maximaler Breite,
der die Lernstichprobe separiert.
(Max-margin, large-margin training)

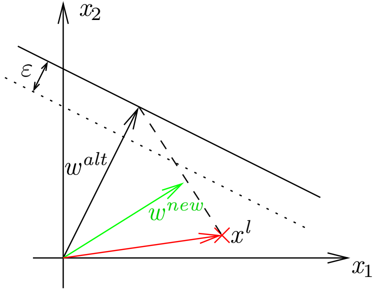


Nach „Vorbereitung 1“ und „Vorbereitung 2“:

$$\min_l \frac{\langle x^l, w \rangle}{\|w\|} \to \max_w$$

Vergleiche mit Perceptron

$$\min_l \frac{\langle x^l, w \rangle}{\|w\|} > 0$$



ε-genauer Algorithmus:

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

2 Wenn nich gefunden – Ende.

3 Suche $\gamma^* = \arg\min_\gamma \|w^{alt} + \gamma(x^l - w^{alt})\|^2$, aktualisiere $w^{neu} = w^{alt} + \gamma(x^l - w^{alt})$, gehe zu 1.

Terminiert nach einer endlichen Anzahl der Schritte bei $\varepsilon > 0$
(Beweis ähnlich dem Perzeptron Algorithmus)

Terminiert nicht unbedingt bei $\varepsilon = 0$.