Neuronale Netze

Einführung und Perceptron

INTERACTIVE VISUAL DATA MINING

NEURONALE NETZE

– Motivation & Definition

– Vorbild Biologie

– Einführung

– Perceptron

NEURONALE NETZE

Künstliche neuronale Netze sind:

- Massiv parallel verbundene Netzwerke aus
- Einfachen (adaptiven) Elementen in
- Hierarchischer Ordnung oder Organisation

Diese Netze sollen in der selben Art wie biologische Nervensysteme mit der Welt interagieren.

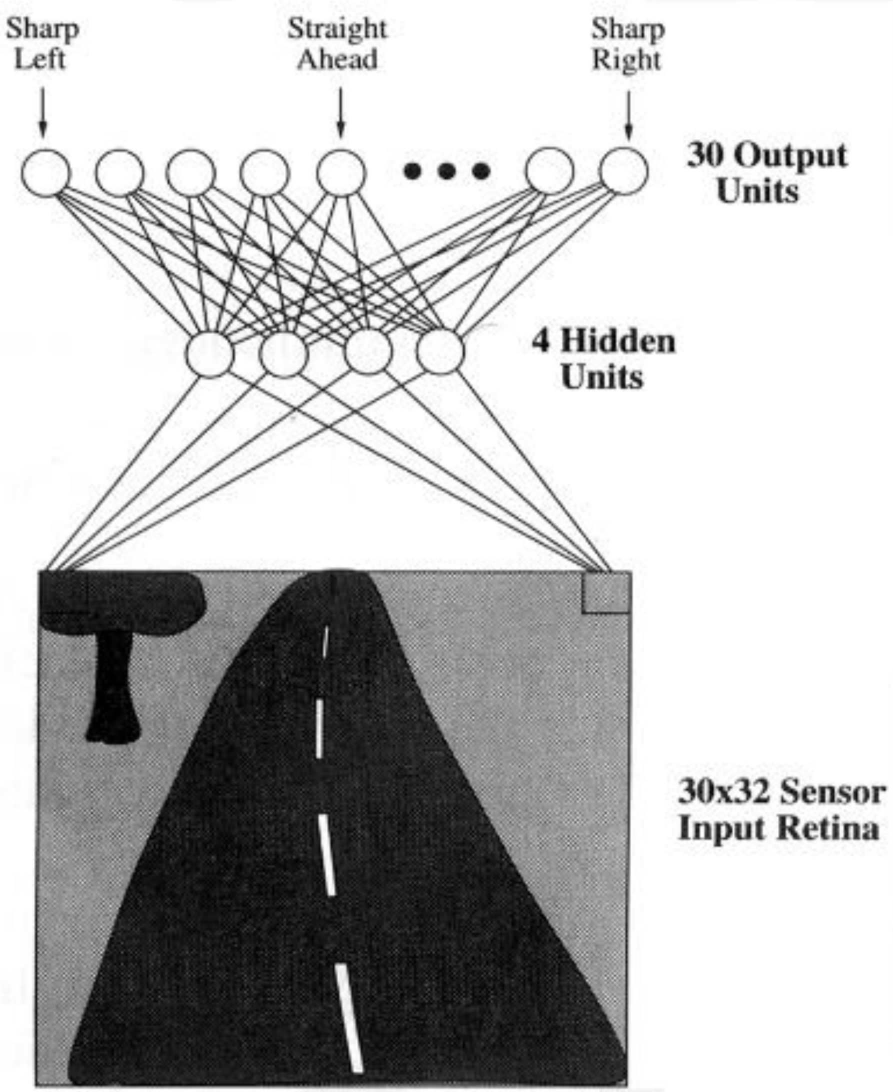
(Kohonen 84)

NEURONALE NETZE

- Forschung:
  - Modellierung und Simulation biologischer neuronaler Netze
  - Speicherung von Information
  - Funktionsapproximation
  - …

- Anwendungen:
  - Analyse Sensordaten
  - Medizin
  - Schrifterkennung
  - Bilderkennung
  - Zeitreihenanalyse
  - Sprachverarbeitung
  - ….

NEURONALE NETZE



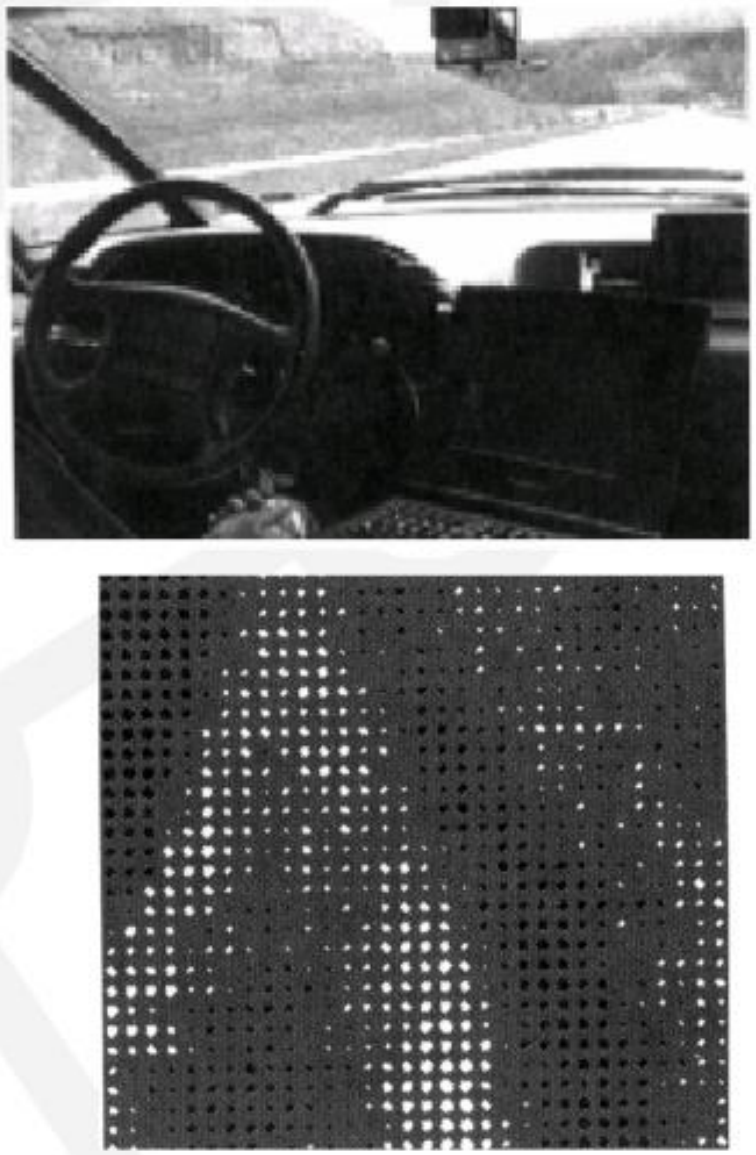


Figure 1: Neural Network learning to steer an autonomous vehicle (Mitchell 1997)

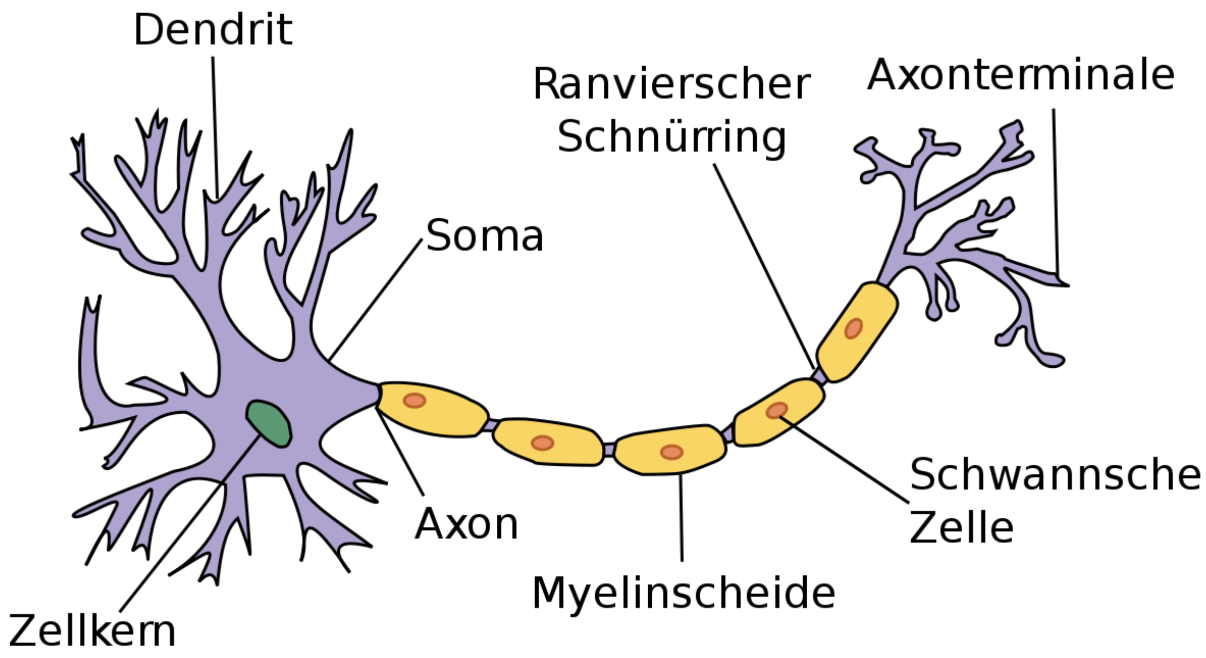
NEURONALE NETZE

- Künstliche neuronale Netze sind gekennzeichnet durch:
  - Massiv parallele Informationsverarbeitung
  - Propagierung der Informationen durch Kanten
  - Verteilte Informationsspeicherung
  - Black Box Modell

- Phasen:
  - Aufbauphase (Topologie des Netzes)
  - Trainingsphase (Lernen)
  - Arbeitsphase (Propagation)

NEURONALE NETZE

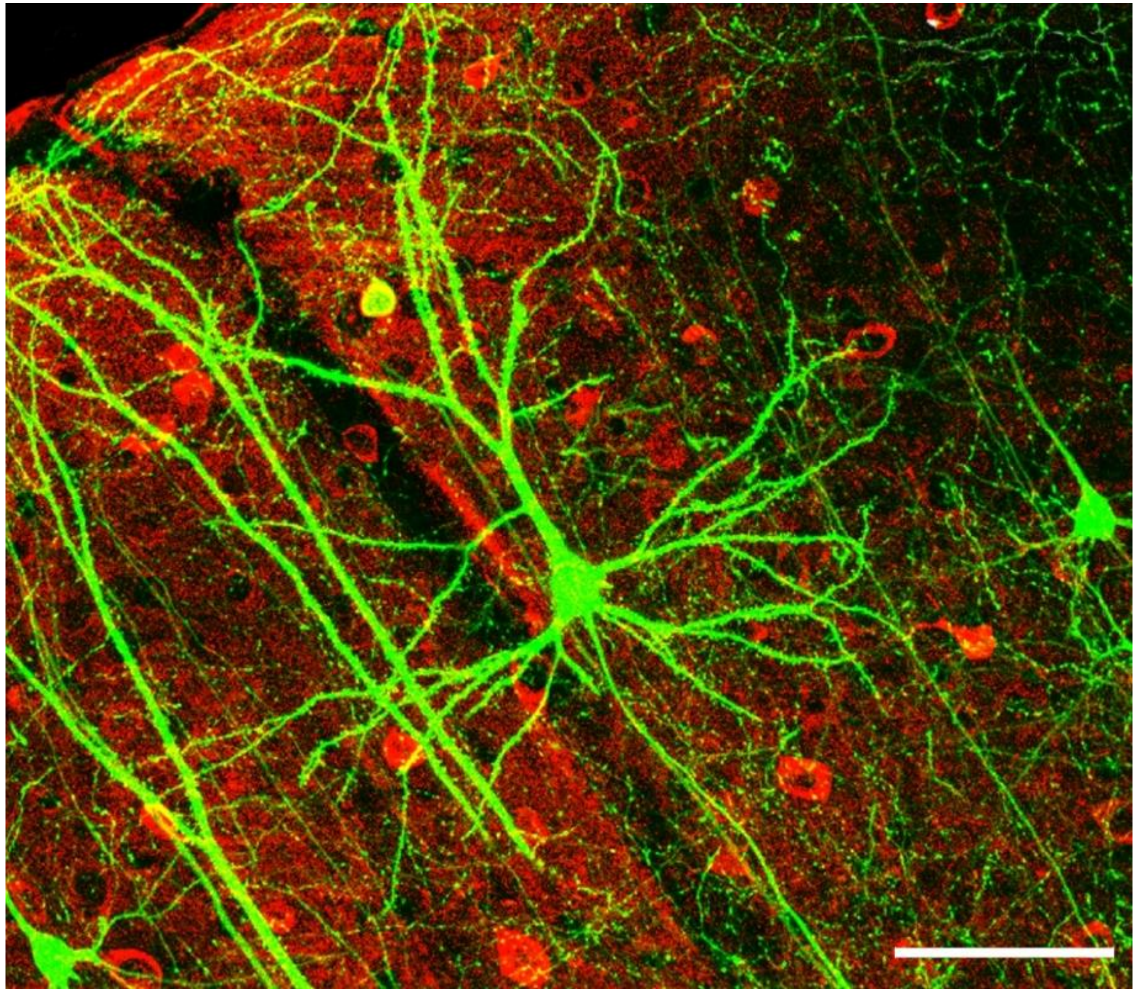
- Vorbild Biologie:
  - Neuronen können
    - Informationen aufnehmen (Dendriten)
    - Informationen verarbeiten (Zellkörper)
    - Informationen weiterleiten (Axon, Synapsen)



NEURONALE NETZE

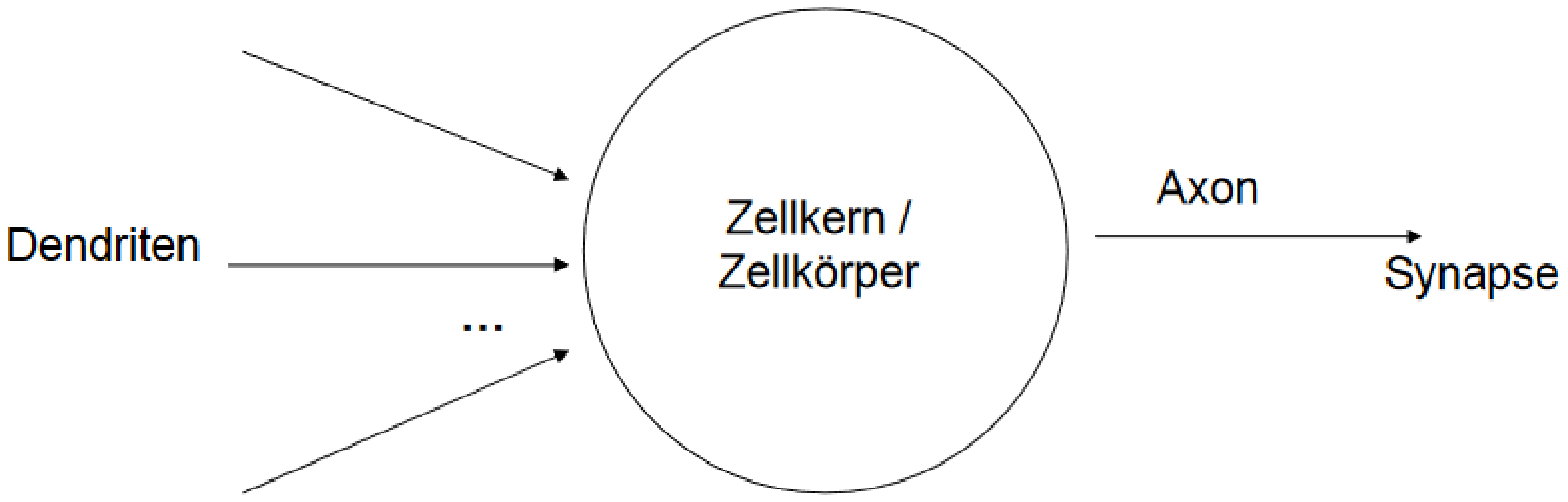
Vorbild Biologie

- Das menschliche Gehirn besteht aus
  - ca. 10^11 Neuronen, die mit
  - ca. 10^4 anderen Neuronen durch
  - ca. 10^13 Synapsen verbunden sind



NEURONALE NETZE

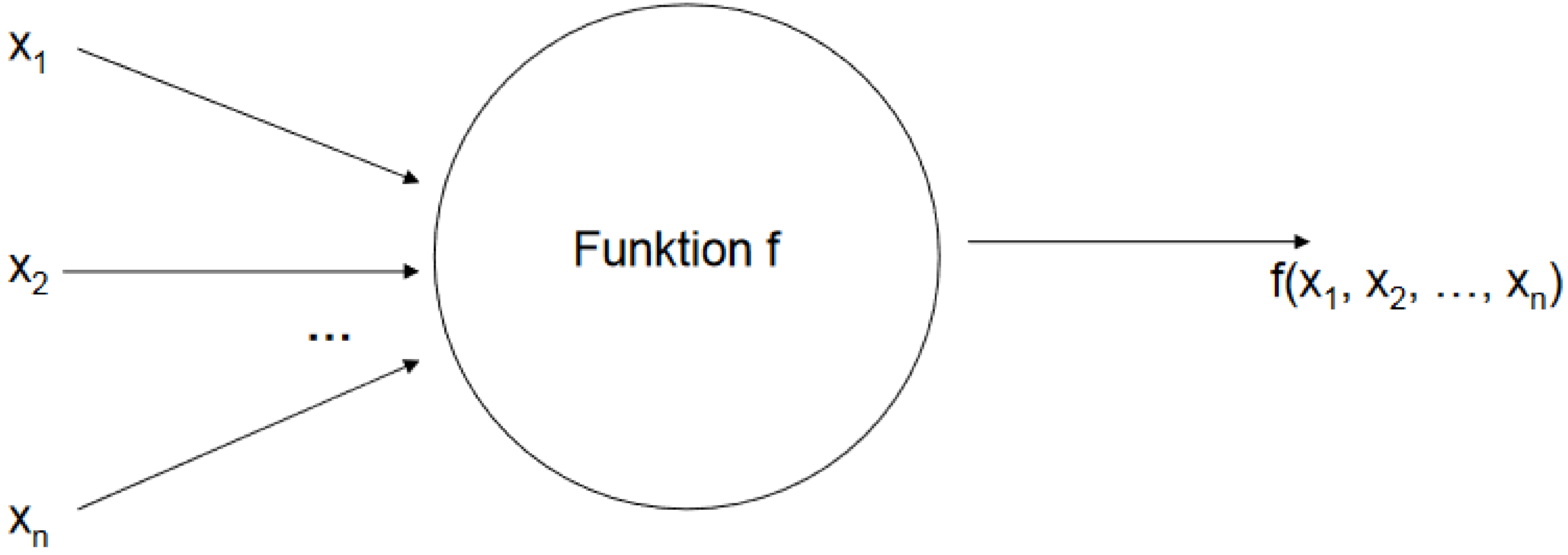
– Abstraktion



Signal-Eingabe **Signal-Verarbeitung** **Signal-Ausgabe**

NEURONALE NETZE

– Modell



McCulloch-Pitts-Neuron 1943:

$x_i \in \{ 0, 1 \} =: \mathbb{B}$

$f: \mathbb{B}^n \to \mathbb{B}$

NEURONALE NETZE

- 1943: Warren McCulloch & Walter Pitts
- Beschreibung neurologischer Netzwerke
  - Modell McCulloch Pitts Neuron
  - Grundidee:
    - Neuron ist entweder aktiv oder inaktiv
    - Fähigkeiten entstehen durch Vernetzung der Neuronen

- Es werden nur statische Netze betrachtet
  - Topologie wird vorher festgelegt
  - Es werden keine neuen Verbindungen beim Lernen erstellt
    - Widerspruch zur Biologie!

NEURONALE NETZE

– McCulloch Pitts Neuron

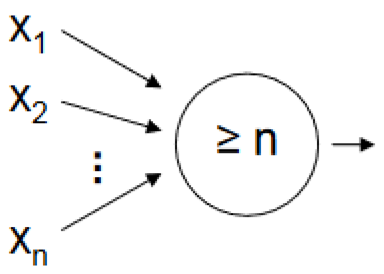
  – n binäre Eingangssignale x_1 bis x_n

  – Schwellwert θ > 0

$$f(x_1, \ldots, x_n) = \begin{cases} 1 & \text{falls } \sum_{i=1}^{n} x_i \geq \theta \\ 0 & \text{sonst} \end{cases}$$

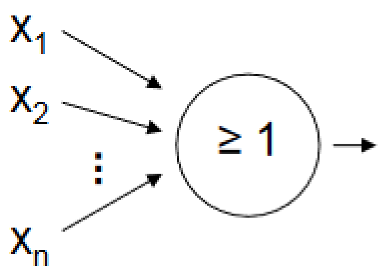
– Damit sind folgende logische Funktionen realisierbar:

Boolsches AND



θ = n

Boolsches OR



θ = 1

NEURONALE NETZE

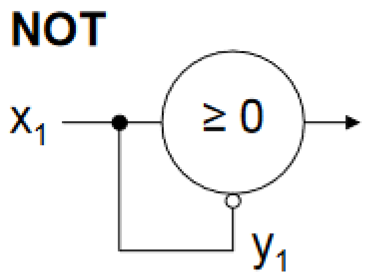
- McCulloch Pitts Neuron
  - n binäre Eingangssignale x_1 bis x_n
  - Schwellwert θ > 0
  - Zusätzlich m binäre hemmende Signale y_1 bis y_m

$$\tilde{f}(x_1, \ldots, x_n; y_1, \ldots, y_m) = f(x_1, \ldots, x_n) \cdot \prod_{j=1}^{m} (1 - y_j)$$

- Ist ein y_j = 1, dann ist die Ausgabe 0
- Sonst gilt:

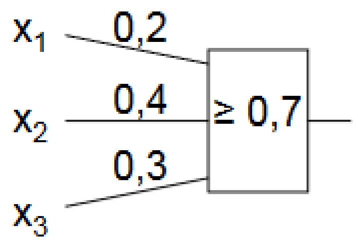
Summe der Eingänge >= Schwellwert -> Ausgabe = 1

Ansonsten Ausgabe = 0

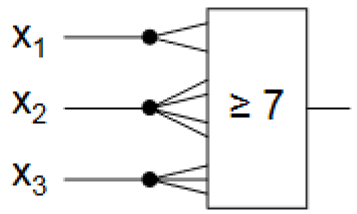


NEURONALE NETZE

- Verallgemeinerung mit Gewichten



  - Ist äquivalent zu



- Dieser Knoten liefert 1 bei

  $0{,}2\ x1 + 0{,}4\ x2 + 0{,}3\ x3 \geq 0{,}7$

- Wenn man die Gewichte mit 10 multipliziert

  $2\ x1 + 4\ x2 + 3\ x3 \geq 7$

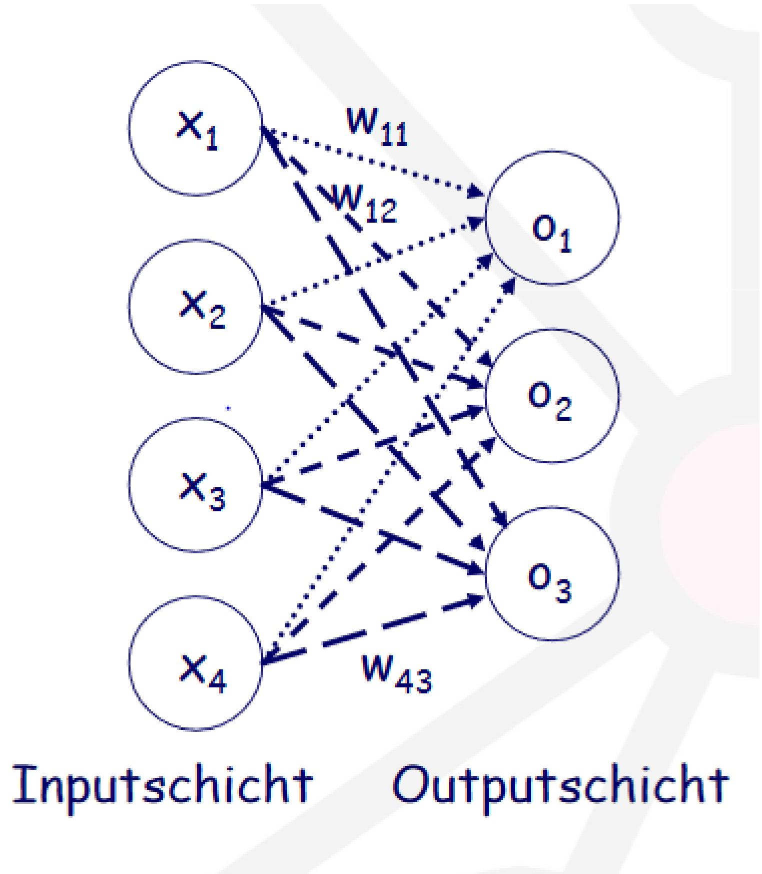
Die Eingänge entsprechend dupliziert, erhält man einen äquivalenten Knoten.

NEURONALE NETZE

- Einfaches Perceptron
  - Motivation
  - Definition
  - Geometrische Interpretation
  - Lernen mittels Delta Regel
  - XOR Problem

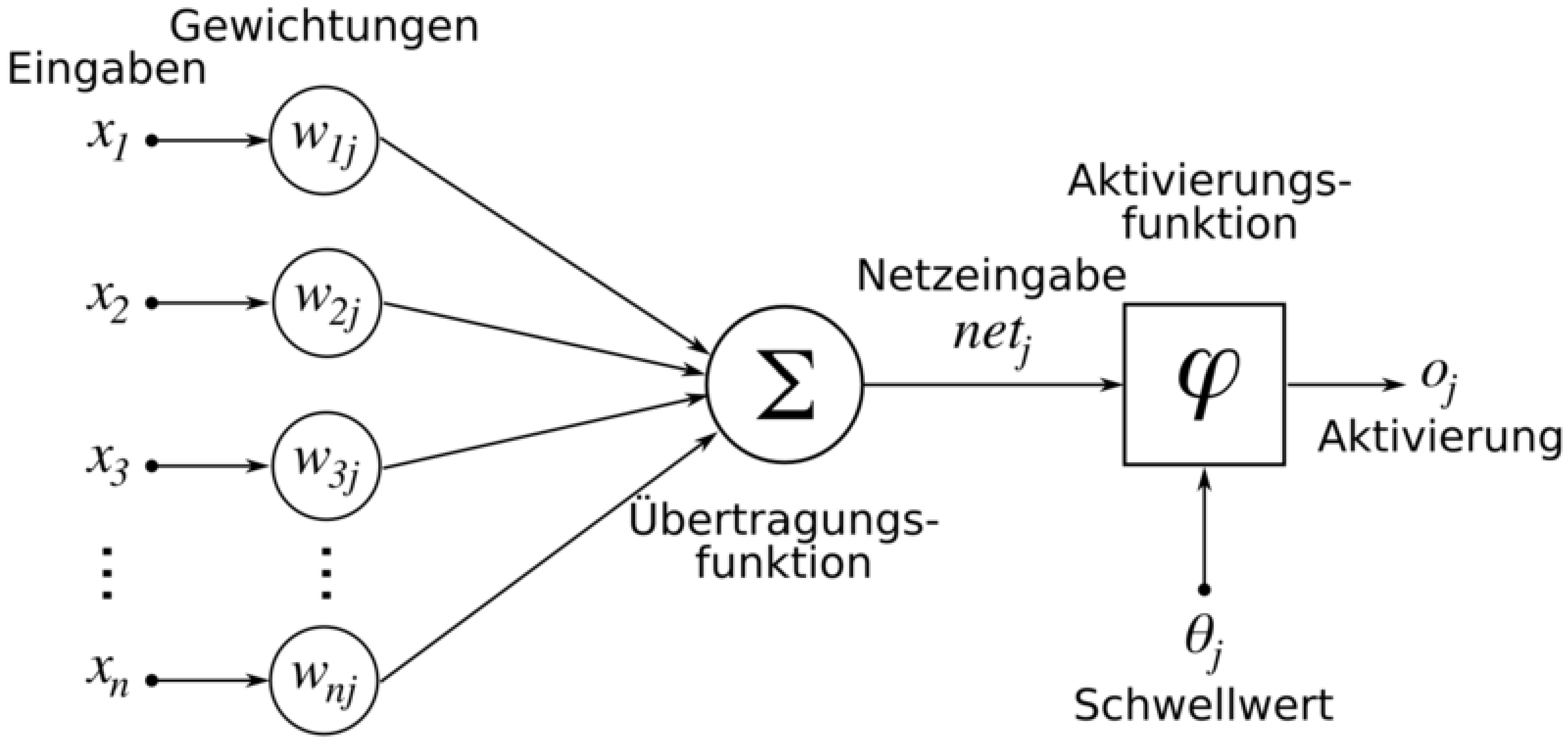
NEURONALE NETZE

- Perceptron (Rosenblatt 1958)
  - Jedes Outputneuron hat einen eigenen unabhängigen Netzbereich
  - Zur Vereinfachung hat jedes Netz im folgenden nur einen Output
  - War historisch als Hardwareimplementierung gedacht
  - Später auf Multilayer Perceptrons erweitert



NEURONALE NETZE

– Genereller Aufbau



NEURONALE NETZE

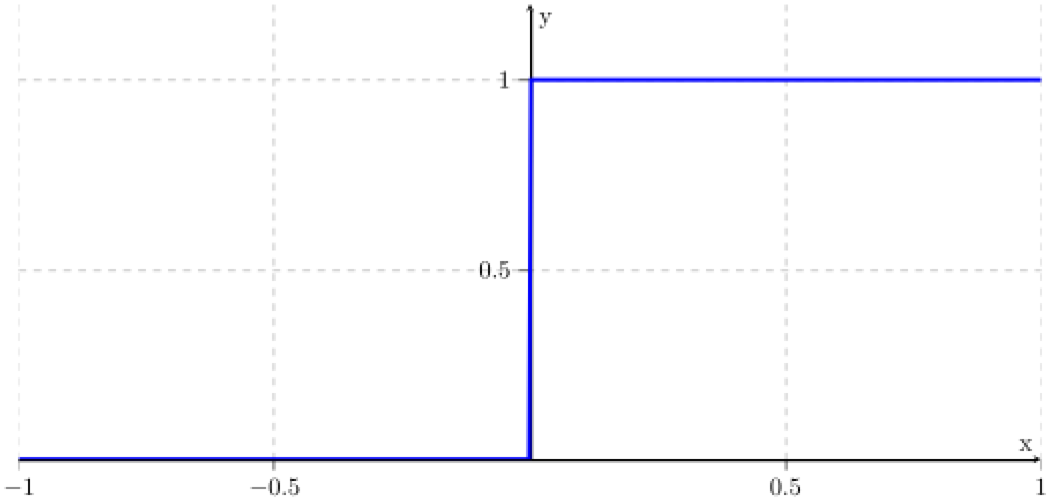
– Aktivierungsfunktionen

  – Schwellwertfunktion:

    – Nimmt nur die Werte 0 und 1 an

    – Schwellwert bestimmt die Aktivierung

    – Alles oder Nichts Funktionsweise



$$\varphi^{\mathbf{hlim}}(v) = \begin{cases} \mathbf{1} & \mathbf{wenn}\ v \geq \mathbf{0} \\ \mathbf{0} & \mathbf{wenn}\ v < \mathbf{0} \end{cases}$$

NEURONALE NETZE

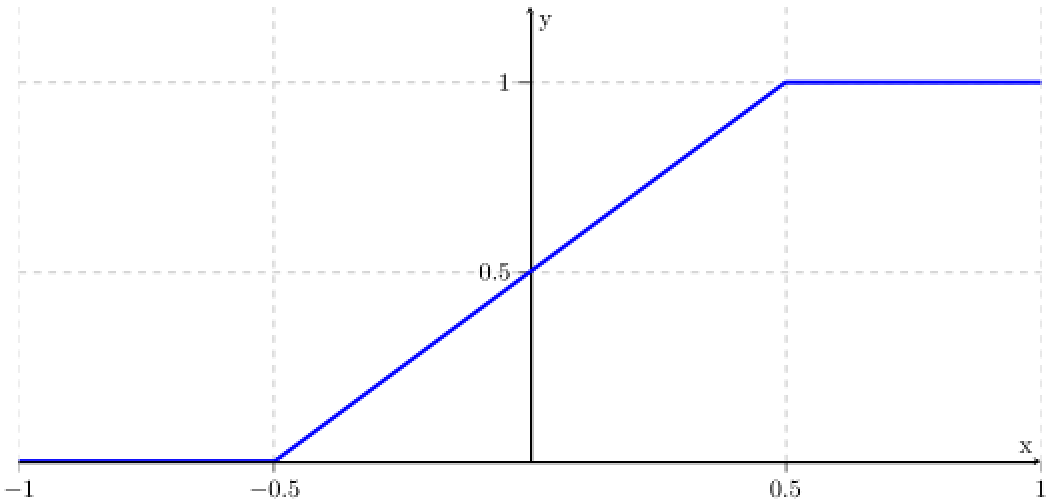
– Aktivierungsfunktionen

  – Stückweise lineare Funktion:

    – Abbildung eines begrenzten Intervalls mit einer linearen Funktion

    – Außerhalb konstante Werte

$$\varphi^{\mathrm{pwl}}(v) = \begin{cases} 1 & \text{wenn } v \geq \frac{1}{2} \\ v + \frac{1}{2} & \text{wenn } -\frac{1}{2} < v < \frac{1}{2} \\ 0 & \text{wenn } v \leq -\frac{1}{2} \end{cases}$$



NEURONALE NETZE

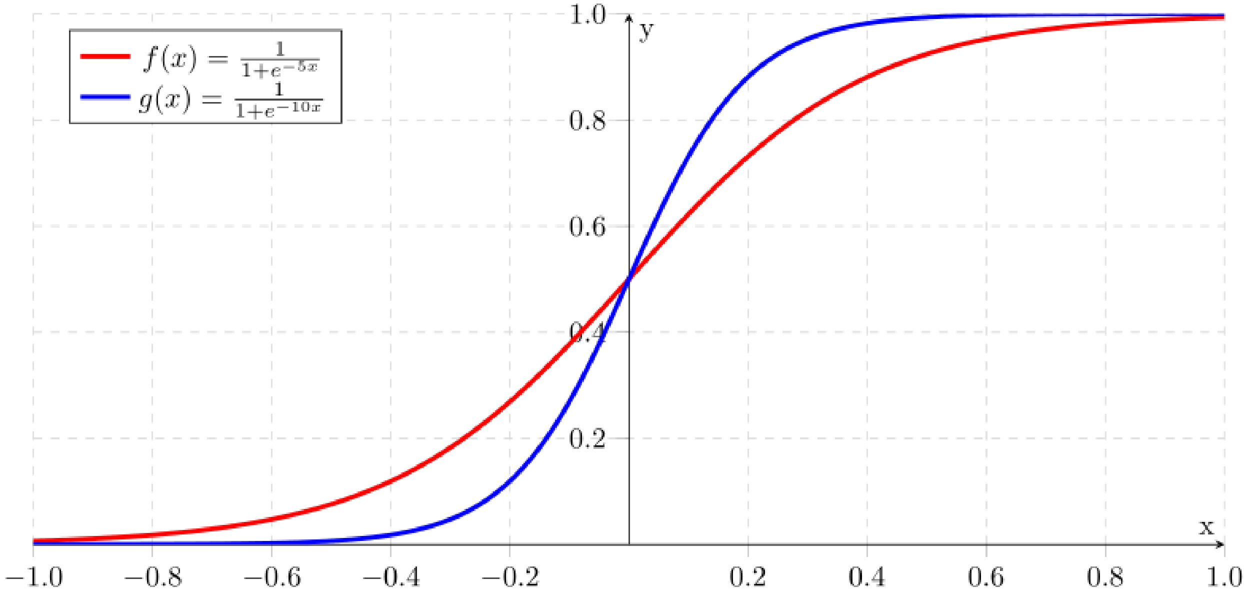
- Aktivierungsfunktionen

  - Sigmoid Funktion

    - Wurden lange als Standardfunktion genutzt

    - Steigungsmaß alpha kann modifiziert werden

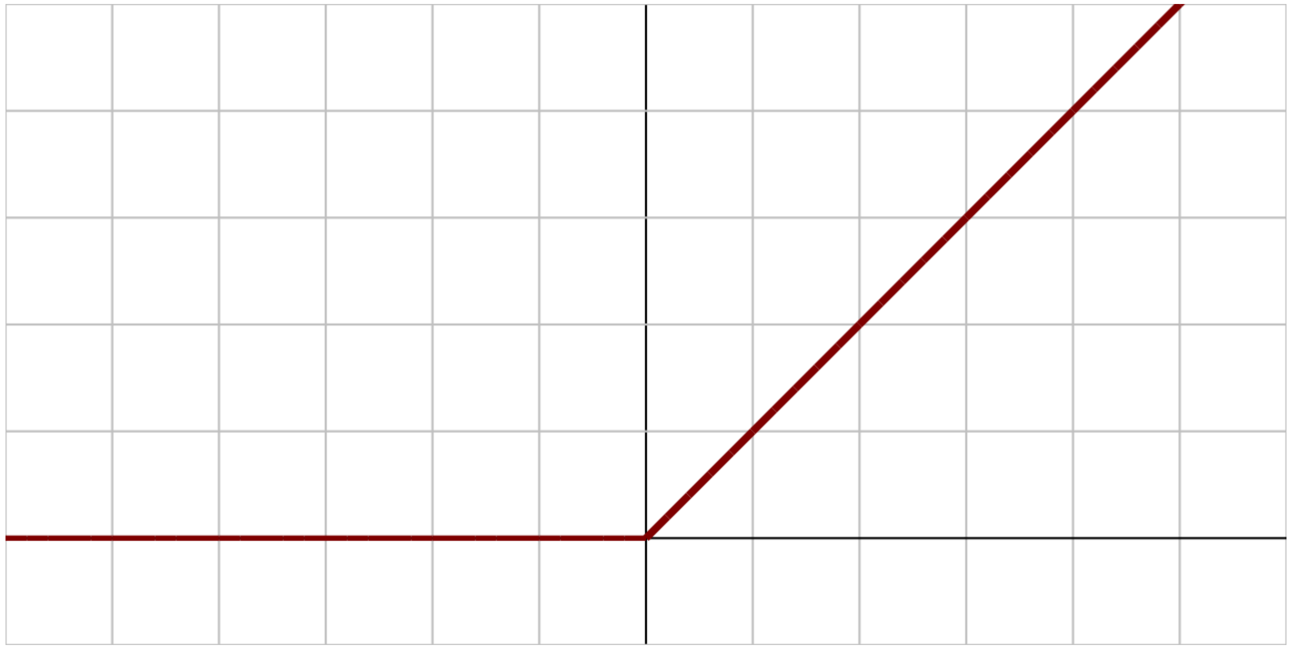
$$\varphi_a^{sig}(v) = \frac{1}{1 + \exp(-av)}$$



NEURONALE NETZE

- Aktivierungsfunktionen
  - Rectifier (ReLU) Funktion
    - rectified linear activation unit
    - Abwandlung der stückweisen linearen Funktion
    - Nur positiver Teil wird linear abgebildet

$$\varphi(v) = \max(0, v)$$



NEURONALE NETZE

- Outputneuron hat:
  - Schwellwert s
  - Aktivität a aus den Inputs und Gewichten
  - Nutzt Aktivierungsfunktion, um den Output zu berechnen
  - Historisch wurde die Sprungfunktion verwendet

- Bildet damit eine Trenngerade zwischen 2 Klassen

$$w_1 x_1 + w_2 x_2 \geq \theta \quad \begin{matrix} \xrightarrow{J} 1 \\ \xrightarrow{N} 0 \end{matrix}$$

- Umgestellt nach x2

$$x_2 \geq \frac{\theta}{w_2} - \frac{w_1}{w_2} x_1 \quad \begin{matrix} \xrightarrow{J} 1 \\ \xrightarrow{N} 0 \end{matrix}$$

NEURONALE NETZE

– Bildet damit eine lineare Trenngerade zwischen 2 Klassen

$$w_1 x_1 + w_2 x_2 \geq \theta \quad \begin{matrix} \xrightarrow{J} 1 \\ \xrightarrow{N} 0 \end{matrix}$$

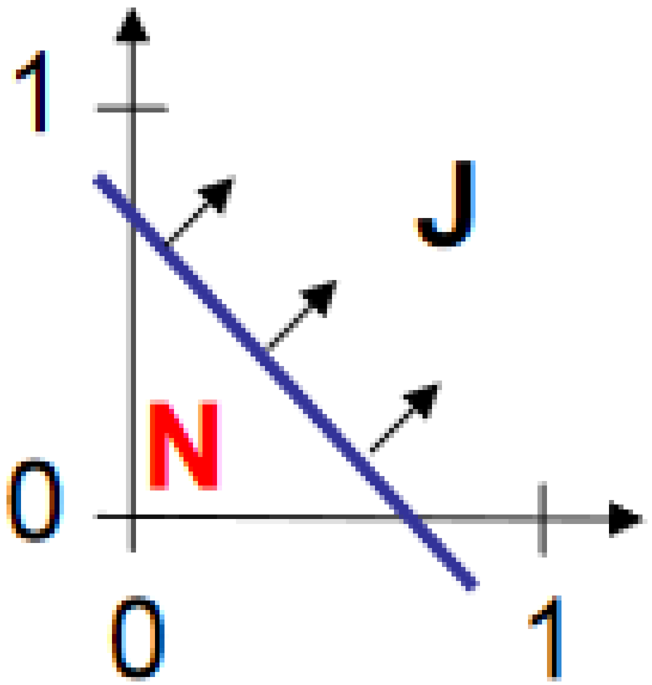
– Umgestellt nach x2

$$x_2 \geq \frac{\theta}{w_2} - \frac{w_1}{w_2} x_1 \quad \begin{matrix} \xrightarrow{J} 1 \\ \xrightarrow{N} 0 \end{matrix}$$

– Beispiel:

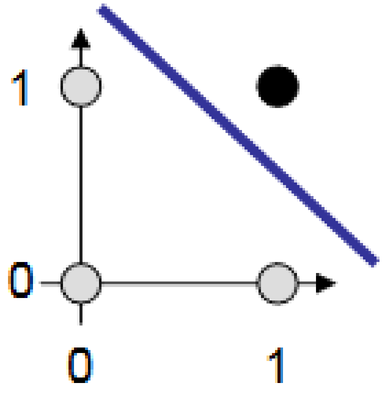
0.9 x_1 + 0.8 x_2 >= 0.6

x_2 >= 3/4 - 9/8 x_1

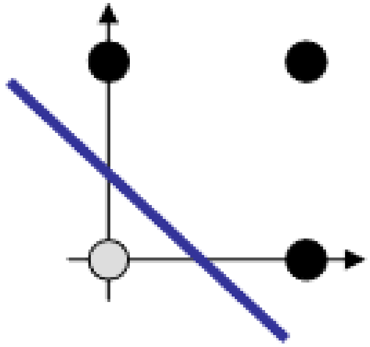


NEURONALE NETZE

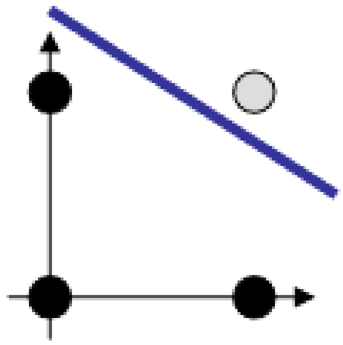
AND



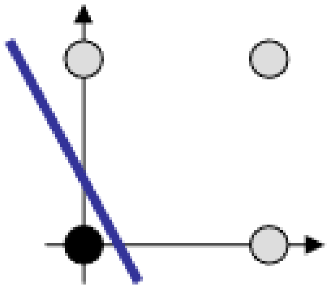
OR



NAND



NOR



NEURONALE NETZE

- Lernverfahren
  - Gegeben ist ein Trainingsdatensatz
    - Daten sind disjunkt in 2 Menge x und y aufgeteilt
    - Gesucht wird eine trennende Hyperebene
      - Basiert auf den Gewichten
      - Teilt x und y auf

- Problem: x und y müssen linear trennbar sein!

NEURONALE NETZE

- Lernverfahren Delta Regel
  - Beim Training werden die Daten dem Netz als Input gezeigt
  - Output für die Beispiele bekannt (supervised)
  - Vergleich von Soll und Ist Zustand im Output

- Bei Abweichung werden Schwellwert und Gewichte adaptiert
- Ist nur auf differenzierbare Aktivierungsfunktionen anwendbar!
- Anpassung kann nach jedem Input oder nach jeder Epoche durchgeführt werden
- Epoche = Ein Durchlauf aller Trainingsdaten

NEURONALE NETZE

- Lernverfahren Delta Regel

$$\Delta w_{ji} = \alpha(t_j - y_j)g'(h_j)x_i$$

  - Alpha: konstante Lernrate
  - g(x) ist die Aktivierungsfunktion
  - g' ist die Ableitung von g
  - t_j ist die Sollausgabe

- y_j ist die Istausgabe
- h_j ist die gewichtete Summe der Inputs des Neurons
- x_i ist der Input I
- Mit

$$h_j = \sum x_i w_{ji}$$

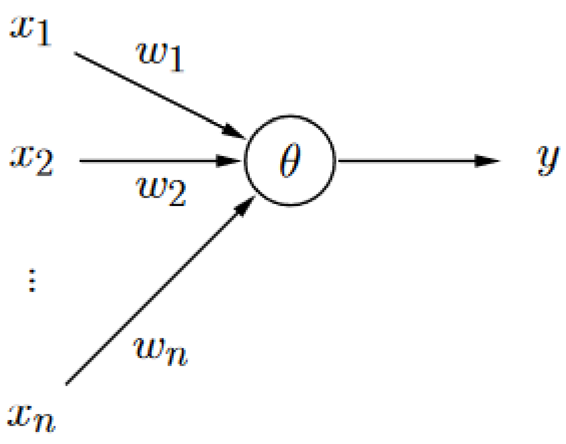
$$y_j = g(h_j)$$

- Die gleiche Formel wird auf auf den Schwellwert angewendet

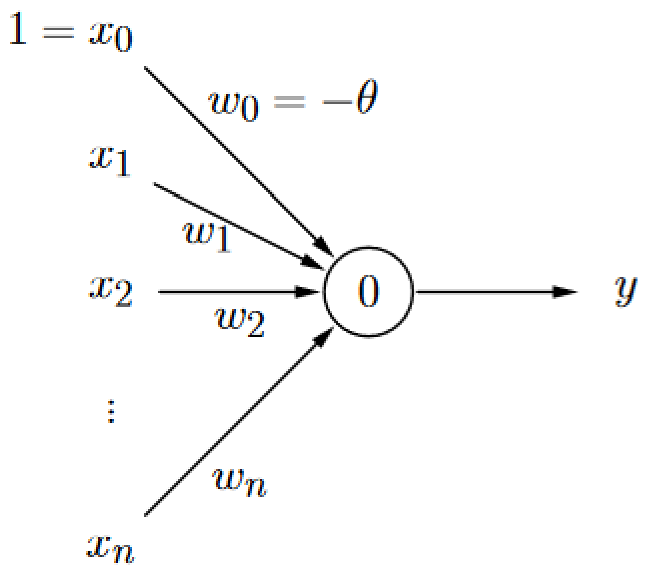
NEURONALE NETZE

– Lernverfahren Delta Regel

  – Der Schwellwert ist damit auch nur ein Gewicht!



$$\sum_{i=1}^{n} w_i x_i \geq \theta$$



$$\sum_{i=1}^{n} w_i x_i - \theta \geq 0$$

NEURONALE NETZE

- Lernverfahren Delta Regel
- Initialer Schwellwert -1
- Lernfaktor 0.2

Eingang1	Eingang2	Ausgang
0	0	0
0	1	1
1	0	0
1	1	1

	$i_1 = 0/i_2 = 0$	$i_1 = 0/i_2 = 1$	$i_1 = 1/i_2 = 0$	$i_1 = 1/i_2 = 1$
ω_0	1.0	1.0	1.2	1.2
ω_1	1.0	1.0	0.8	0.8
ω_2	1.0	1.0	1.0	1.0

	$i_1 = 0/i_2 = 0$	$i_1 = 0/i_2 = 1$	$i_1 = 1/i_2 = 0$	$i_1 = 1/i_2 = 1$
ω_0	1.2	1.0	1.0	1.0
ω_1	0.8	0.8	0.8	0.8
ω_2	1.0	1.2	1.2	1.2

NEURONALE NETZE

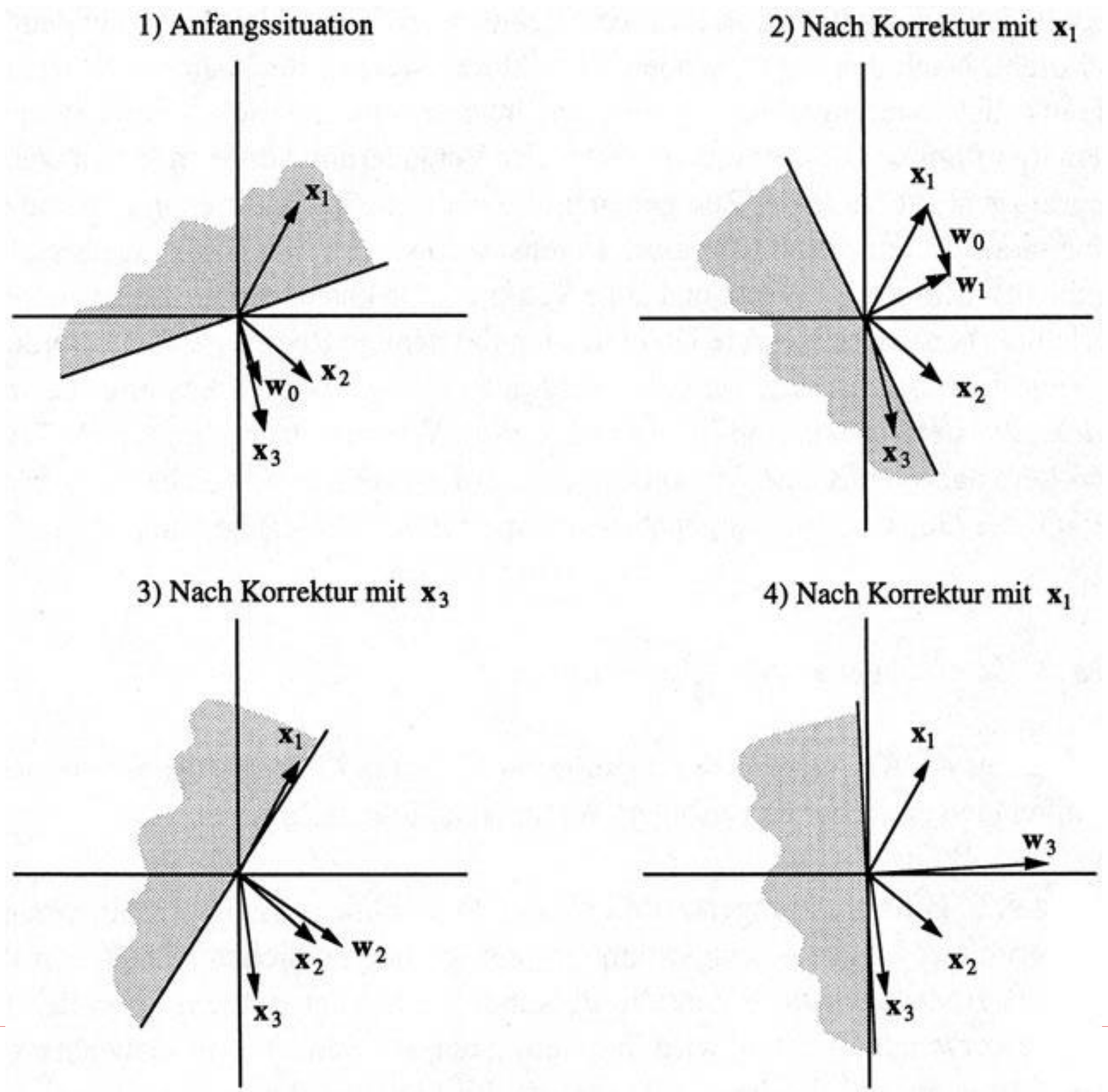
- Lernverfahren Delta Regel
  - Kovergenz und Korrektheit
  - Satz: Wenn ein Perzeptron eine Klasseneinteilung lernen kann, dann lernt es diese mit der Delta Regel in endlich vielen Schritten

- Problem: Falls das Perzeptron das Modell nicht erlernt, kann man nicht entscheiden:
- Ob genügend Epochen gerlernt wurde oder
- Das Problem nicht erlernbar ist
- → Es gibt keine obere Schranke für die Lerndauer
  - → Overfitting Problematik

NEURONALE NETZE

– Lernverfahren Delta Regel

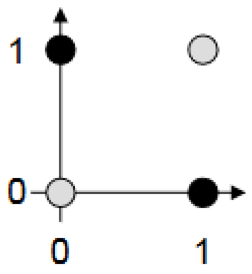
  – Geometrische Interpretation nach Rojas



NEURONALE NETZE

– XOR Problem

XOR



x_1	x_2	xor
0	0	0
0	1	1
1	0	1
1	1	0

$\Rightarrow 0 < \theta$

$\Rightarrow w_2 \geq \theta$

$\Rightarrow w_1 \geq \theta$

$\Rightarrow w_1 + w_2 < \theta$

$w_1, w_2 \geq \theta > 0$

$\Rightarrow w_1 + w_2 \geq 2\theta$

Widerspruch!

$$w_1 x_1 + w_2 x_2 \geq \theta$$

– Triviales Problem ist mit künstlichen Neuronen nicht lösbar!

NEURONALE NETZE

– XOR Problem

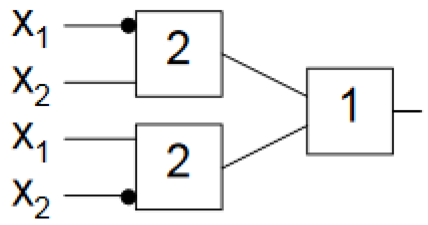
  – 1969 publiziert von Minsky und Papert

  – Folgerung: künstliche neuronen sind eine Sackgasse

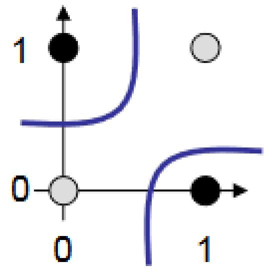
  – Forschung wurde daraufhin ca 15 Jahre eingestellt

– Lösungen sind:

  – Mehrschichtige Perzeptrons (Feedforward Netze)



  – Nichtlineare Trennfunktionen



$g(0,0) = -1$
$g(0,1) = +1$
$g(1,0) = +1$
$g(1,1) = -1$

$g(x_1, x_2) = 2x_1 + 2x_2 - 4x_1x_2 - 1$ mit $\theta = 0$

NEURONALE NETZE

- Vorteile künstlicher neuronaler Netze
  - Sehr gute Mustererkennung
  - Verarbeitung von Inputs mit
    - Verrauschten Eigenschaften
    - Unvollständigen Eigenschaften
    - Widersprüchlichen Eigenschaften
  - Möglicher multimodaler Input (Zahlen, Farben, Töne, Sprache, etc)
  - Erstellt ein Modell ohne Hypothesen des Nutzers
  - Fehlertolerant
  - Im Produktivbetrieb einfach zu nutzen

NEURONALE NETZE

- Nachteile künstlicher neuronaler Netze
  - Lange Trainingszeiten (bis zu Monaten)
  - Lernerfolg ist nicht garantiert
  - Generalisierbarkeit ist nicht garantiert
  - Viele Daten sind notwendig
  - Komplexes Blackbox Verfahren
  - Evaluierung des Netzes ist schwierig
  - Anzahl der Knoten und Kanten kann schnell sehr groß werden
  - Löst „nur“ das Problem mit einem Modell, liefert aber keine Hinweise auf die wichtiges Features