

# PROSEMINAR: DATA MINING

## Neuronale Netze: Grundlagen

---

Lukas Krenz

12. Juni 2015

Technische Universität München



man in black shirt is  
playing guitar.



black and white dog  
jumps over bar.



a woman holding a  
teddy bear in front of  
a mirror.

Quelle: Deep Visual-Semantic Alignments for Generating Image  
Descriptions, CVPR 2015

Neuronale Netze sind ein:

- graphisches
- nicht-lineares
- flexibles

Modell für maschinelles Lernen.

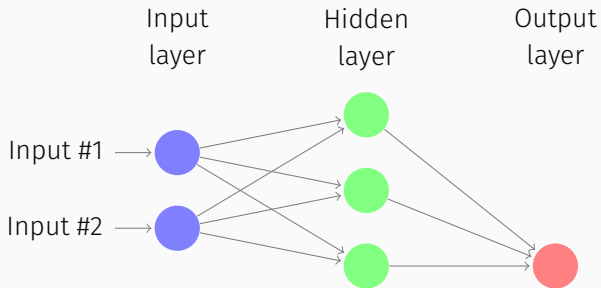
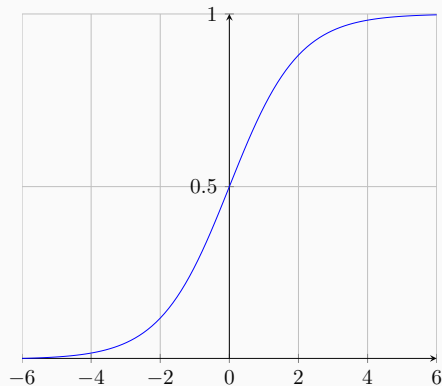


Abbildung 1: Ein 2-schichtiges MLP.

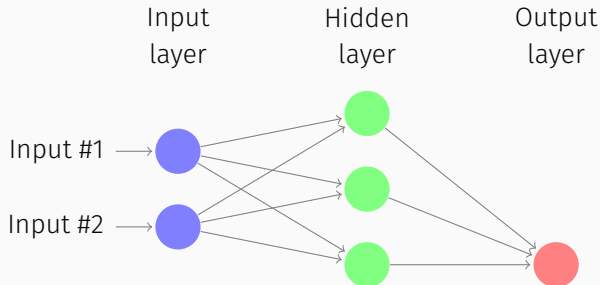
Abbildung 2: Die logistische Funktion  $\sigma_1$



$$\sigma_1(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

$$\sigma_2(x) = \tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \quad (2)$$

$$\sigma_3(x) = \max(0, x) \quad (3)$$



$$\text{net}_j = \sum_i w_{ji} y_i = w_j^t x \quad (4)$$

$$y_j = \sigma(\text{net}_j) \quad (5)$$

$$E = \sum_{k=1}^K \sum_{i=1}^N \left( y_i^k - t_i^k \right)^2 \quad (6)$$

$$= \sum_n E_n(y_1, \dots, y_c) \quad (7)$$

$$\frac{\partial E^n}{\partial w_{ij}} = \frac{\partial E^n}{\partial net_j} \frac{\partial net_j}{\partial w_{ji}} \quad (8)$$

$$\frac{\partial net_j}{\partial w_{ji}} = y_i \quad (9)$$

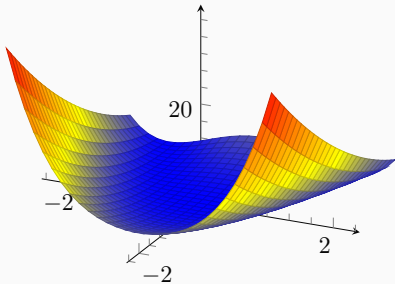
$$\frac{\partial E^n}{\partial net_j} = \delta_j \quad (10)$$

$$\frac{\partial E^n}{\partial w_{i,j}} = y_i \delta_j. \quad (11)$$

$$\delta_j = \begin{cases} \sigma'(net_j)(y_j - t_j) & \text{wenn } j \text{ Ausgabeneuron ist} \\ \sigma'(net_j) \sum_k w_{kj} \delta_k & \text{wenn } j \text{ verdecktes Neuron ist} \end{cases} \quad (12)$$



Abbildung 3: Die  
Rosenbrock Funktion  
 $(1 - x)^2 + 100(y - x^2)^2$



$$x_{n+1} = x_n - \eta \nabla F(x_n) \quad (13)$$

$$\Delta_w^t = -\eta \nabla_w E(w; t) \quad (14)$$

## Vorteile:

- mächtig
- flexibel

## Nachteile

- viele Parameter, kompliziert anzupassen
- langsam

In der **aktuellen** Forschung relevant: Netze anderer Topologien