

PROSEMINAR: DATA MINING

Neuronale Netze: Grundlagen

Lukas Krenz

12. Juni 2015

Technische Universität München

1. Was sind ANNs? 2. Topologie (Bild eines Beispielnetzes, was ist ein Input Layer, was ein Output, was ein Hidden) - woraus bestehen sie? 3. Aktivierungsfunktionen: Beschränkung auf eine, aber Ansprechen von alternativen. Plot von drei (+/-) Funktionen. 4. Feed-Forward -> Wie berechnet man Werte? Wie werte ich die Output-Knoten aus? 5. Lernen. Eine Kostenfunktion. Grundidee von Backprop: Effiziente Berechnung von partiellen Ableitungen durch Kettenregel. 6. Optimierung. Gradientenverfahren, dazu kurz Besprechung der Nachteile davon. 7. Kurzer Schluss: Erwähnen von Deep Learning.

blablablabla

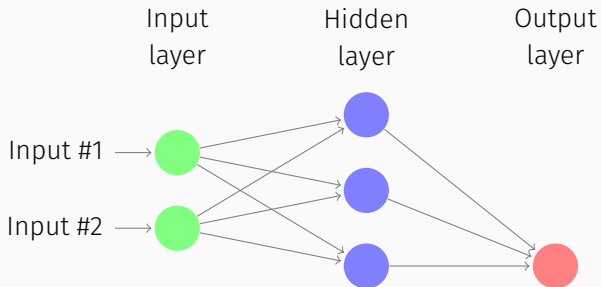
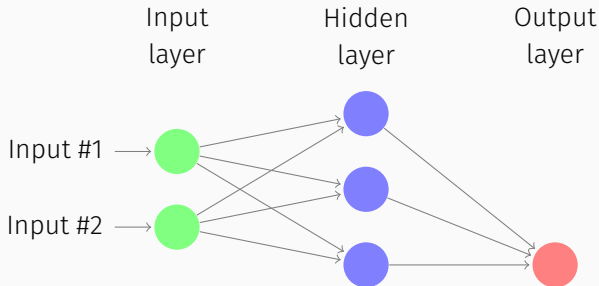


Abbildung 1: Ein 2-schichtiges MLP.

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

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

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



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

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

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

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

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

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

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

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

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

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

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

blablablabla