

## 1 BAYES-THEORIE

## 2 AKTIVIERUNGSFUNKTIONEN

Marius

## 3 FEHLERFUNKTIONEN

- $E_{MSE}(w) = \frac{1}{2} \sum_{x \in X} \sum_k (t_k^x - o_k^x)^2$
- Mean-Squared-Error =  $\frac{1}{N} * SSE$
- $E_{CE}(w) = - \sum_{x \in X} \sum_k [t_k^x * \log(o_k^x) + (1 - t_k^x) * \log(1 - o_k^x)]$

## 4 PERZEPTRON LERNALGORITHMUS

$$w_i^{t+1} = w_i^t + \eta(t_x - o_x)x_i$$

## 5 BACKPROPAGATION

$$w = w - \eta \nabla_w E(x, w) \text{ mit } \nabla_w E = \frac{\partial E}{\partial o} \frac{\partial o}{\partial \sigma} \frac{\partial \sigma}{\partial w}$$

## 6 HOPFIELD

Aktivierung

## 7 BOLTZMANN-MASCHINEN

Aktivierung

## 8 RESTRICTED-BOLTZMANN-MASCHINEN

Aktivierung

## 9 REINFORCEMENT LEARNING

Q-Learning Bellmanngleichung Policyfunktion TD-Learning — SARSA

## 10 GENERALISIERUNG

$\langle \epsilon_{\text{Dest}} \rangle = \langle \epsilon_{\text{train}} \rangle + 2 \cdot \sigma^2 \frac{p}{n}$  mit Varianz  $\sigma$ , Parameteranzahl  $p$  und Anzahl an Trainingsbeispielen  $n$

## 11 NORMALISIERUNG

- Max-Min (Rescaling):  $x' = \frac{x - \min(x)}{\max(x) - \min(x)}$
- Standardisierung:  $x' = \frac{x - \bar{x}}{\sigma}$
- Skalierung auf Einheitslänge:  $x' = \frac{x}{\|x\|}$
- lückenhafte Daten: Null filling - Smoothing

## 12 REGULARISIERUNG

- L1 Norm:  $\|w\|_{L1} = \sum_j |w_j|$
- L2 Norm:  $\|w\|_{L2} = \sum_j w_j^2$
- KL-Divergenz:
- Cross-Entropy:  $E_{CE}(w) = - \sum_{x \in X} \sum_k [t_k^x * \log(o_k^x) + (1 - t_k^x) * \log(1 - o_k^x)]$
- Edit-Distance:
- Dropout:
- Meiosis:

## 13 ADAPTIVE LERNRATENANPASSUNG

- AdaGrad:  $w_t = w_{t-1} - \frac{\eta}{\sqrt{G_t + \epsilon}} L(x, w_{t-1})$  mit der Diagonalmatrix  $G_t$ , die die Beträge des Gradienten enthält und  $\epsilon$ : Smoothingterm, um Division durch 0 zu verhindern
- $w_t = w_{t-1} - \frac{RMS[\Delta w]_{t-1}}{RMS[g]_t} g_t$ , wobei  $RMS[\Delta w]_t$  der „root mean squared error“  $\sqrt{E[\Delta w^2]_t + \epsilon}$  ist.
- RMSProp:  $w_t = w_{t-1} - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t$

## 14 UPDATES FÜR BACKPROP

- Momentum-Term:  $\Delta w_{ij}(t) = -\eta \frac{\partial E}{\partial w_{ij}(t)} + \alpha * \Delta w_{ij}(t-1)$
- QuickProp:
- WeightElimination:

## 15 AUTOENCODERS

Deaktivierung von Sparse Autoencoder:

## 16 SHARED WEIGHTS BEI TDNNs

Formel:

## 17 KOMISCHES BILD MIT RANDOM *cos* EINFÜGEN

## 18 LVQ

Random Formeln hier: