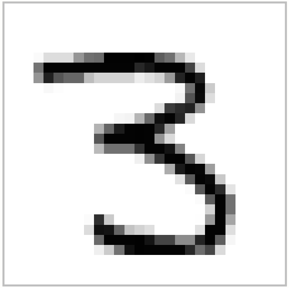
NEURONALE NETZE

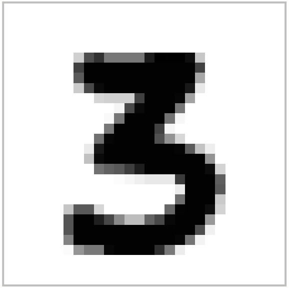
Handschriftliche Zahlen erkennen

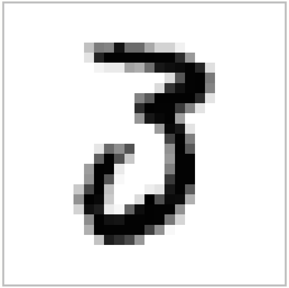
Jasper Gude

28. November 2023
Carl-Friedrich-Gauß-Gymnasium

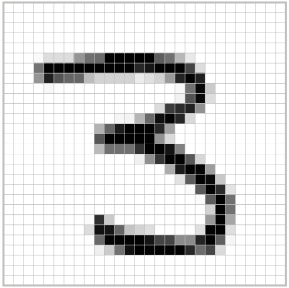
Modellierung des Problems

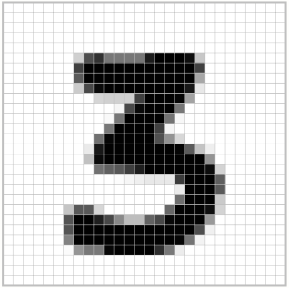


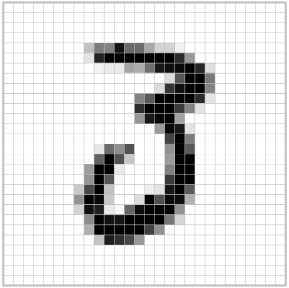




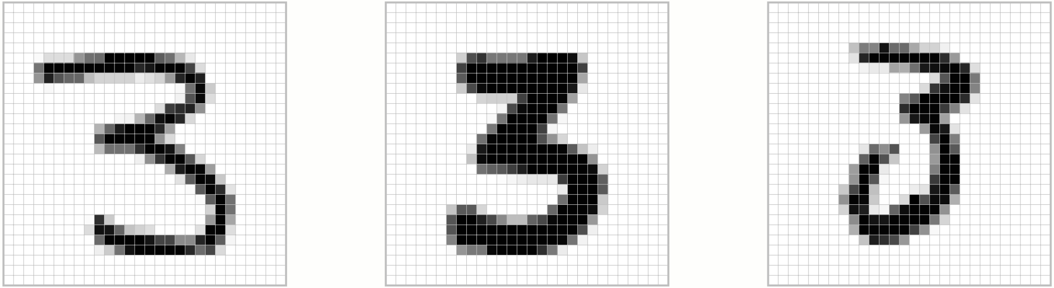
2.2 Modellierung des Problems





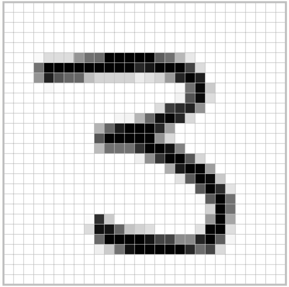


2.3 Modellierung des Problems



0,00 | 0,25 | 0,50 | 0,75 | 1,00

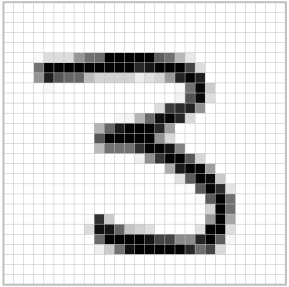
Überführung auf eine Netzstruktur



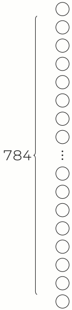
28px x 28px

3.2

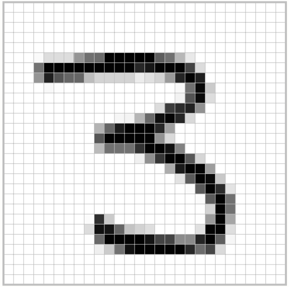
Überführung auf eine Netzstruktur



28px x 28px

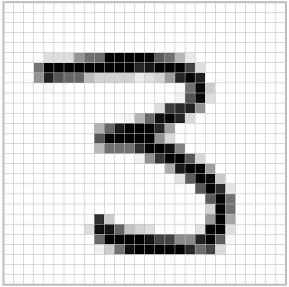


3.3 Überführung auf eine Netzstruktur

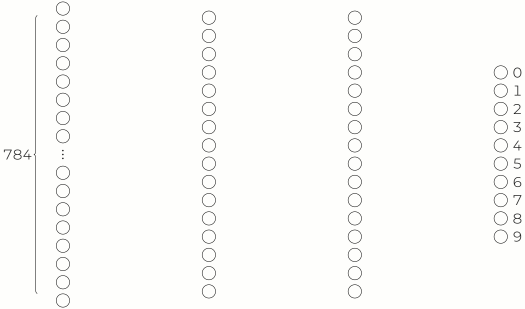


28px x 28px

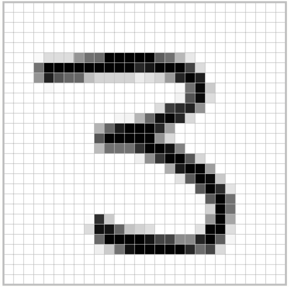
3.4 Überführung auf eine Netzstruktur



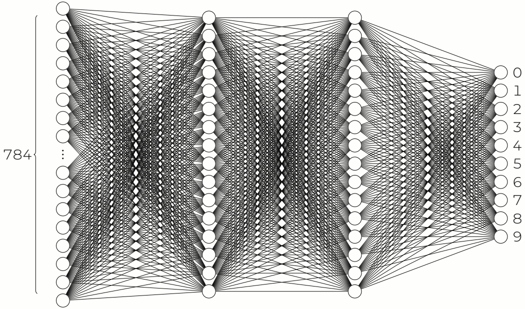
28px x 28px



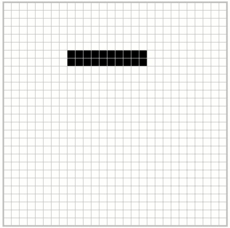
Überführung auf eine Netzstruktur



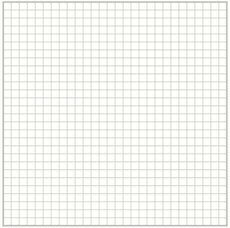
28px × 28px



Gewichtungen setzen



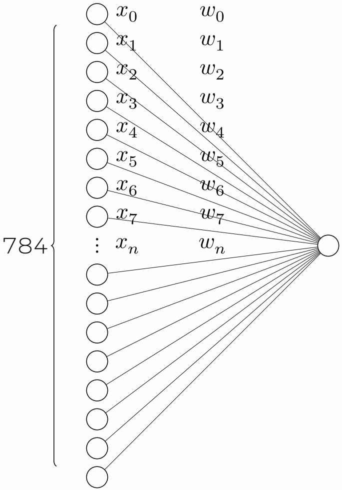
Inputs



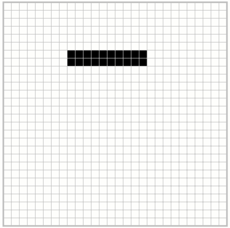
Gewichte

Linearkombination

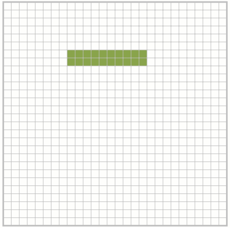
$$w_0 x_0 + w_1 x_1 + \cdots + w_n x_n$$



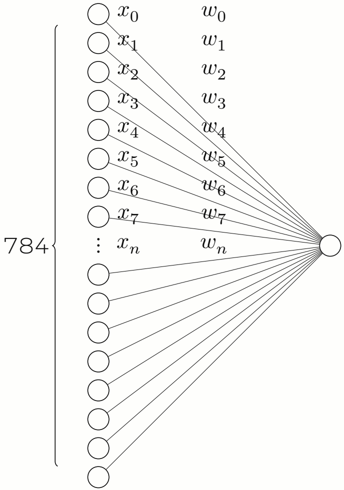
4.2 Gewichtungen setzen



Inputs



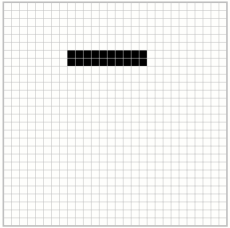
Gewichte



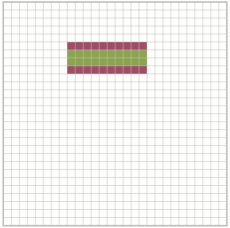
Linearkombination

$$w_0 x_0 + w_1 x_1 + \cdots + w_n x_n$$

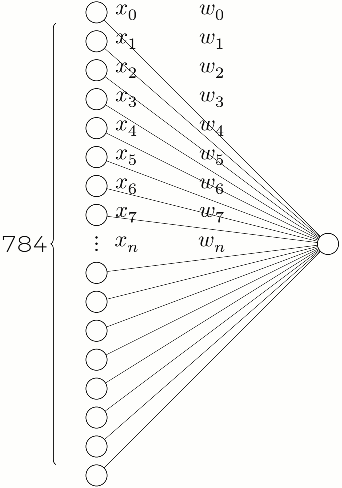
4.3 Gewichtungen setzen



Inputs



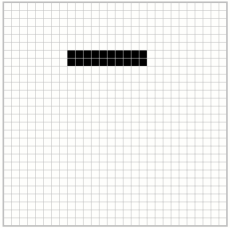
Gewichte



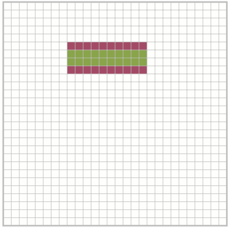
Linearkombination

$$w_0 x_0 + w_1 x_1 + \cdots + w_n x_n$$

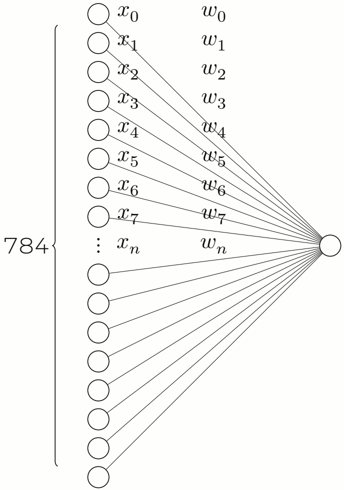
4.4 Gewichtungen setzen



Inputs



Gewichte



Linearkombination

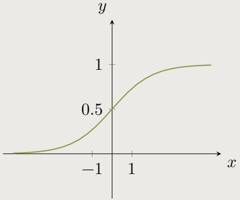
$$w_0 x_0 + w_1 x_1 + \cdots + w_n x_n - b$$

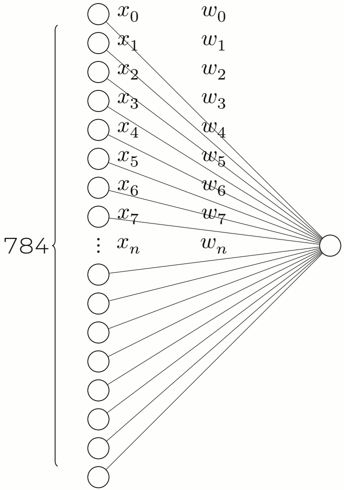
5

Zahlenbereich begrenzen

Sigmoidfunktion

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

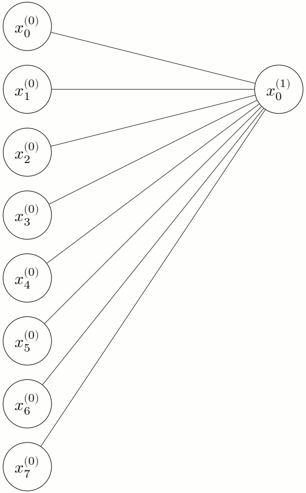




Alles zusammen setzen

Aktivierungsfunktion

$$x_0^{(1)} = \sigma(w_0^{(0)} x_0 + w_1^{(0)} x_1 + \cdots + w_n^{(0)} x_n - b)$$

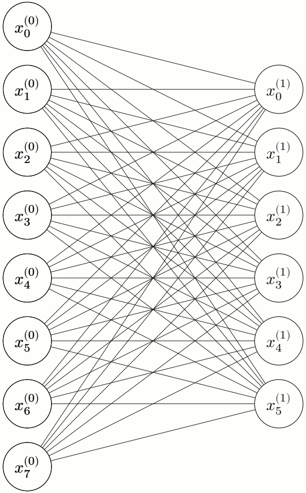


Alles zusammen setzen

Aktivierungsfunktion

$$x_0^{(1)} = \sigma(w_0^{(0)} x_0 + w_1^{(0)} x_1 + \cdots + w_n^{(0)} x_n - b)$$

$$\begin{bmatrix} x_0^{(1)} \\ x_1^{(1)} \\ \vdots \\ x_n^{(1)} \end{bmatrix} = \sigma \left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} x_0^{(0)} \\ x_1^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} \right)$$

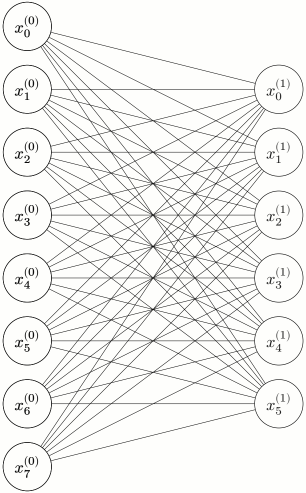


Alles zusammen setzen

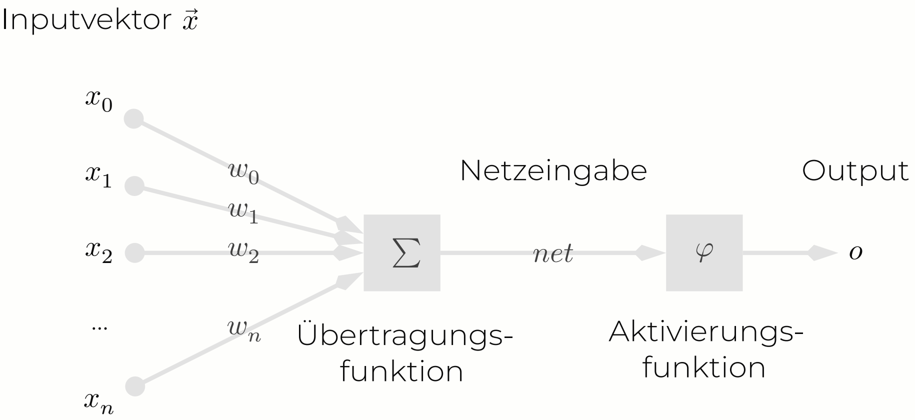
Aktivierungsfunktion

$$\vec{x^{(1)}} = \sigma(W\vec{x^{(0)}} + \vec{b})$$

$$\begin{bmatrix} x_0^{(1)} \\ x_1^{(1)} \\ \vdots \\ x_n^{(1)} \end{bmatrix} = \sigma\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} x_0^{(0)} \\ x_1^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix}\right)$$



Aufbau eines Perzeptrons



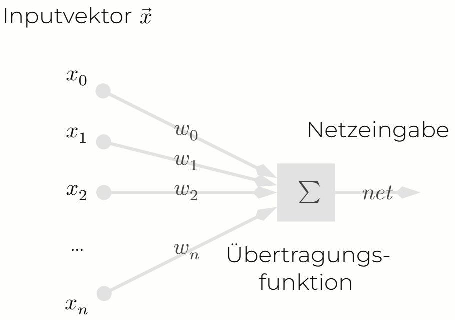
Übertragungsfunktion

Linearkombination

$$net = x_0 w_0 + x_1 w_1 + x_2 w_2 + \ldots + x_n w_n$$

oder

$$net = \sum_{i=0}^{n} x_i w_i$$



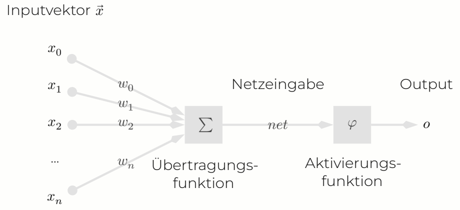
Fehlerfunktion

Dataset

$$X = \Big\{(\vec{x}_0, y_0); (\vec{x}_1, y_1); (\vec{x}_2, y_2); (\dots, \dots); (\vec{x}_n, y_n)\Big\}$$

Mean Squared Error

$$E = \frac{1}{2}\sum_{i=0}^{n}(y_i - o_i)^2$$



Dataset

$$X = \left\{ (\vec{x}_0, y_0); (\vec{x}_1, y_1); (\vec{x}_2, y_2); (\dots, \dots); (\vec{x}_n, y_n) \right\}$$

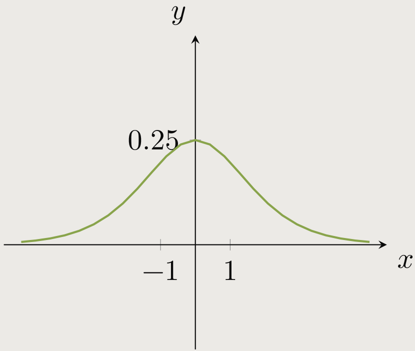
Ableitung der Aktivierungsfunktion

Ableitung der Sigmoidfunktion

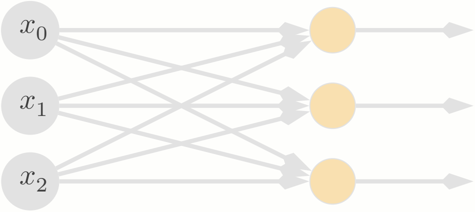
$$\varphi'(x) = \frac{1}{1+e^{-x}} \cdot \left(1 + \frac{1}{1+e^{-x}}\right)$$

oder

$$\varphi'(x) = \varphi(x) \cdot (1 + \varphi(x))$$

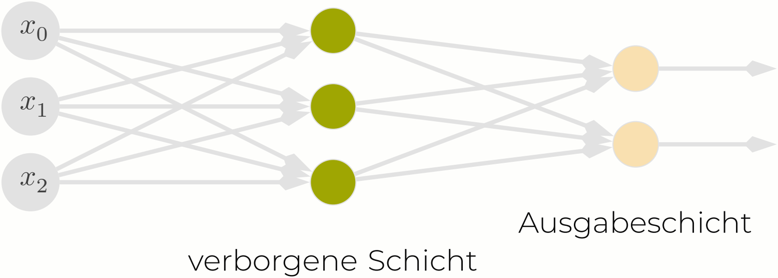


Einschichtiges feedforward-Netz

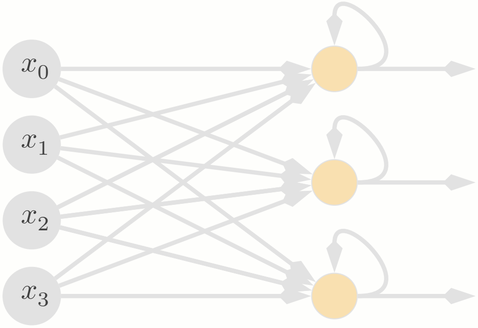


Ausgabeschicht

Mehrschichtiges feedforward-Netz



Rekurrentes Netz



Ausgabeschicht

Jasper Gude

Hockenheim, 28. November 2023