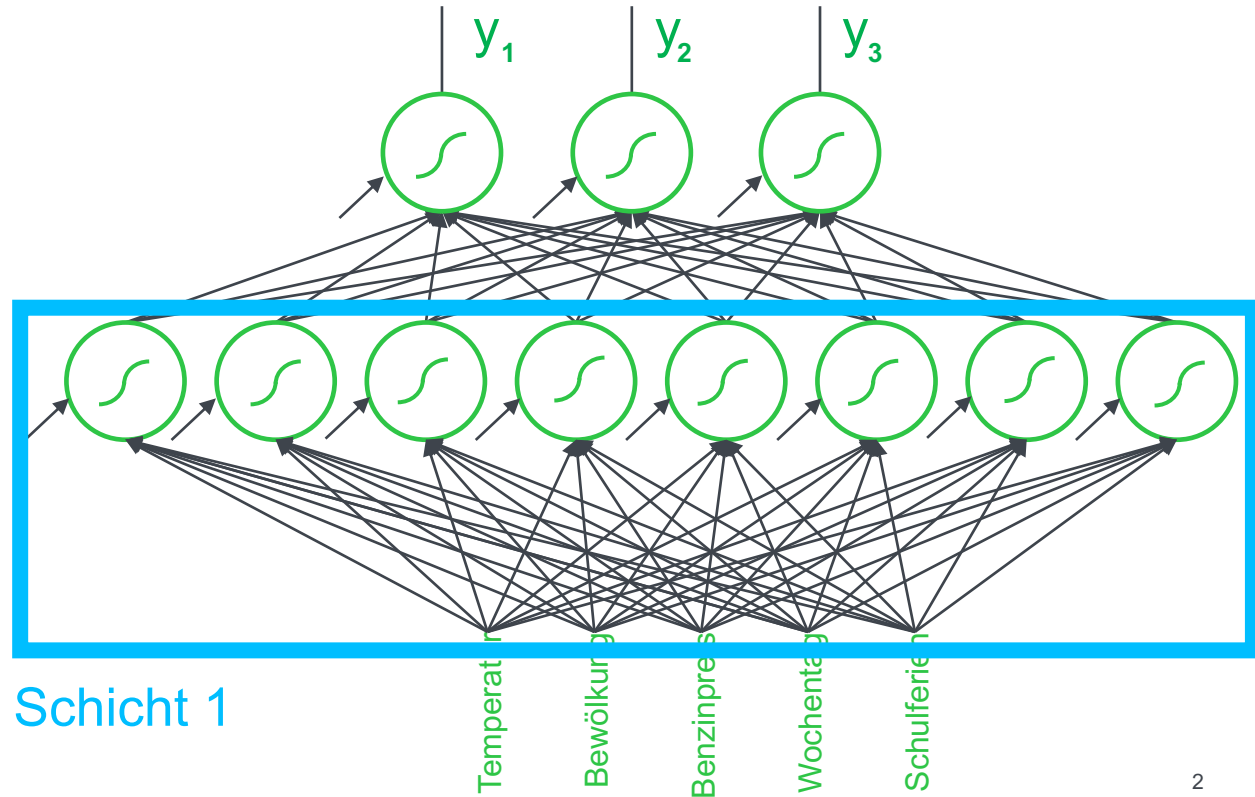


# Parameter in neuronalen Netzen

... und der Forward Pass

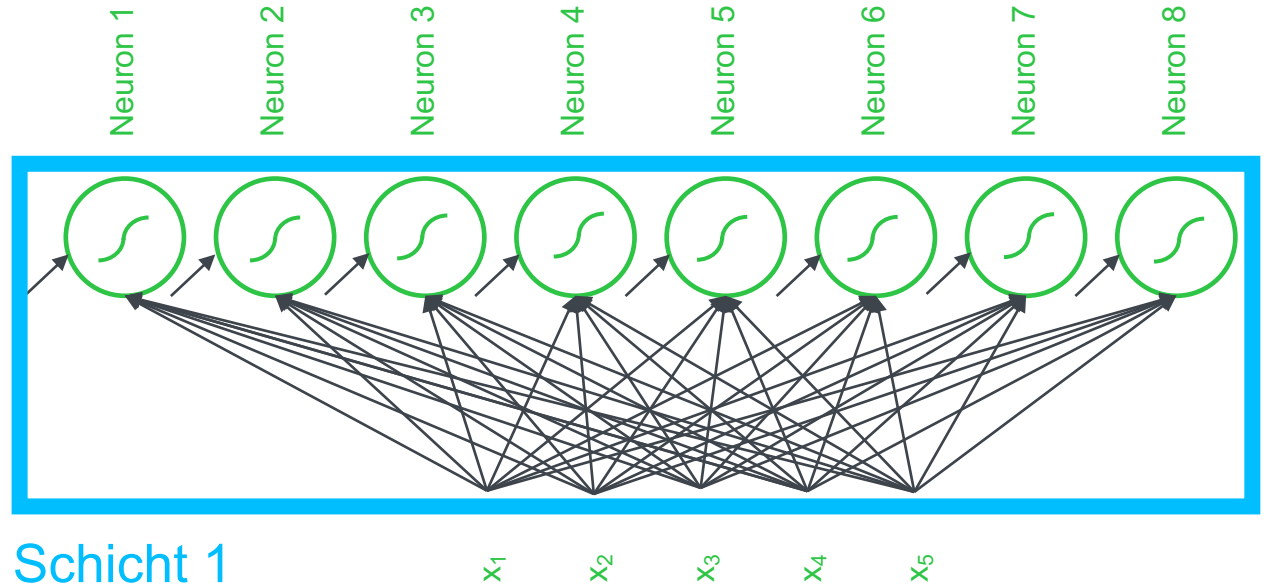
## Welche zu lernenden Parameter gibt es in Neuronalen Netzen?

- Gewichte für Verbindungen zwischen Neuronen
- Bias
- Netzwerkstruktur hier:
  - Feed Forward Neural Network: nicht zirkulär (keine Schleifen)
  - Fully connected (alle Neuronen einer Schicht mit allen Neuronen der nächsten Schicht verbunden)



## Beispiel: Schicht 1

- Gewichte der Verbindungen der 1. Schicht
- Abkürzung:  $w_{ij}^1$   
engl.: weight



## Beispiel: Schicht 1

- Gewichte der Verbindungen der 1. Schicht

- Abkürzung:  $w_{ij}^1$   
engl.: weight

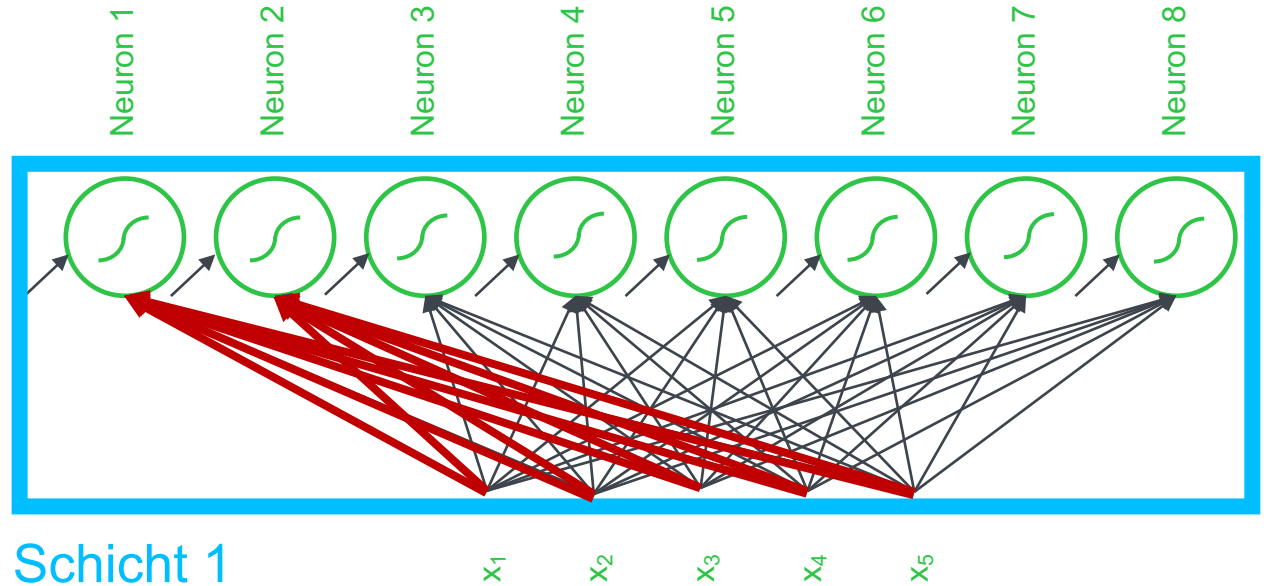
- zu Neuron 1:

$$w_{11}^1, w_{12}^1, w_{13}^1, w_{14}^1, w_{15}^1$$

- zu Neuron 2:

$$w_{21}^1, w_{22}^1, w_{23}^1, w_{24}^1, w_{25}^1$$

- USW.



## Gewichtematrix für Schicht 1

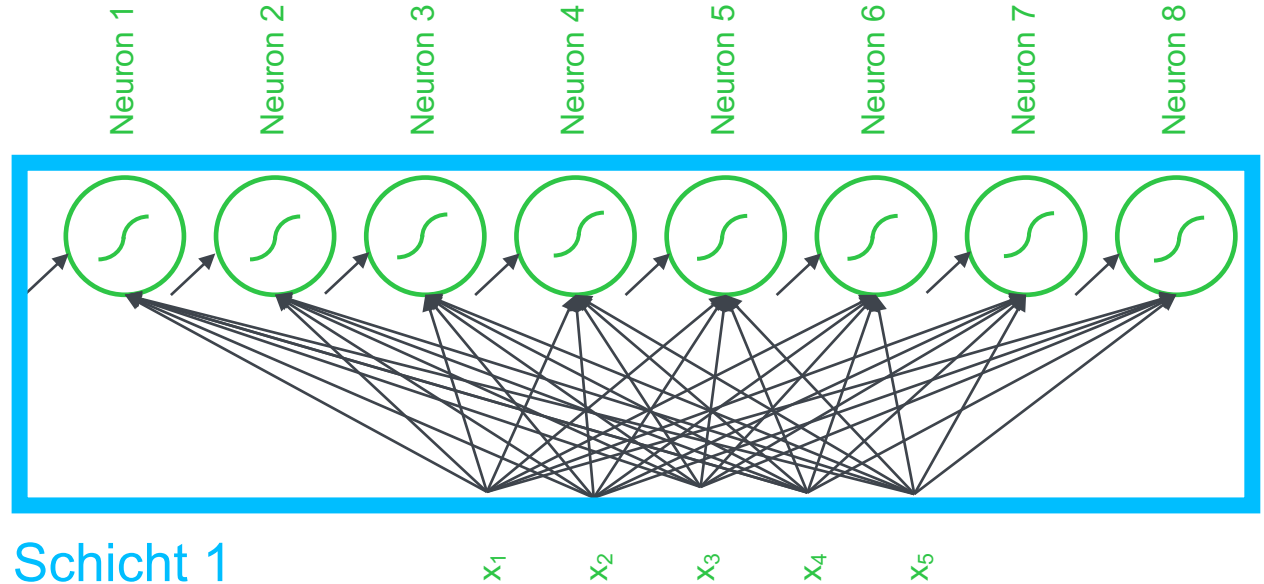
- Alle Gewichte der 1. Schicht
- Ergeben Matrix  $W^1$
- Mit Gewichten  $w_{ij}^1$  für Verbindungen von  $x_j$  zu Neuron  $i$

	Von $x_1$	Von $x_2$	Von $x_3$	Von $x_4$	Von $x_5$
Zu Neuron 1	$w_{11}^1$	$w_{12}^1$	$w_{13}^1$	$w_{14}^1$	$w_{15}^1$
Zu Neuron 2	$w_{21}^1$	$w_{22}^1$	$w_{23}^1$	$w_{24}^1$	$w_{25}^1$
Zu Neuron 3	$w_{31}^1$	$w_{32}^1$	$w_{33}^1$	$w_{34}^1$	$w_{35}^1$
Zu Neuron 4	$w_{41}^1$	$w_{42}^1$	$w_{43}^1$	$w_{44}^1$	$w_{45}^1$
Zu Neuron 5	$w_{51}^1$	$w_{52}^1$	$w_{53}^1$	$w_{54}^1$	$w_{55}^1$
Zu Neuron 6	$w_{61}^1$	$w_{62}^1$	$w_{63}^1$	$w_{64}^1$	$w_{65}^1$
Zu Neuron 7	$w_{71}^1$	$w_{72}^1$	$w_{73}^1$	$w_{74}^1$	$w_{75}^1$
Zu Neuron 8	$w_{81}^1$	$w_{82}^1$	$w_{83}^1$	$w_{84}^1$	$w_{85}^1$

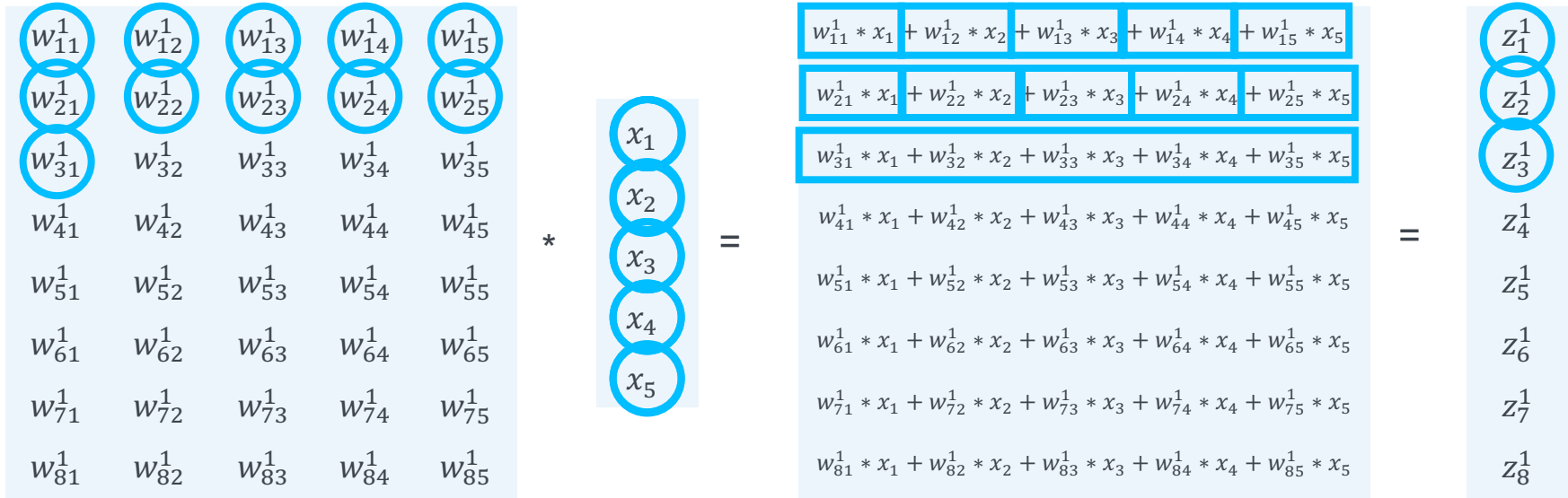
## Logits in Schicht 1 (ohne Bias)

- Neuron 1:  $z_1^1 = w_{11}^1 * x_1$   
 $+ w_{12}^1 * x_2$   
 $+ w_{13}^1 * x_3$   
 $+ w_{14}^1 * x_4$   
 $+ w_{15}^1 * x_5$

- Neuron 2:  $z_2^1 = w_{21}^1 * x_1$   
 $+ w_{22}^1 * x_2$   
 $+ w_{23}^1 * x_3$   
 $+ w_{24}^1 * x_4$   
 $+ w_{25}^1 * x_5$



# Gewichtematrix für Schicht 1



Gewichte (Matrix)

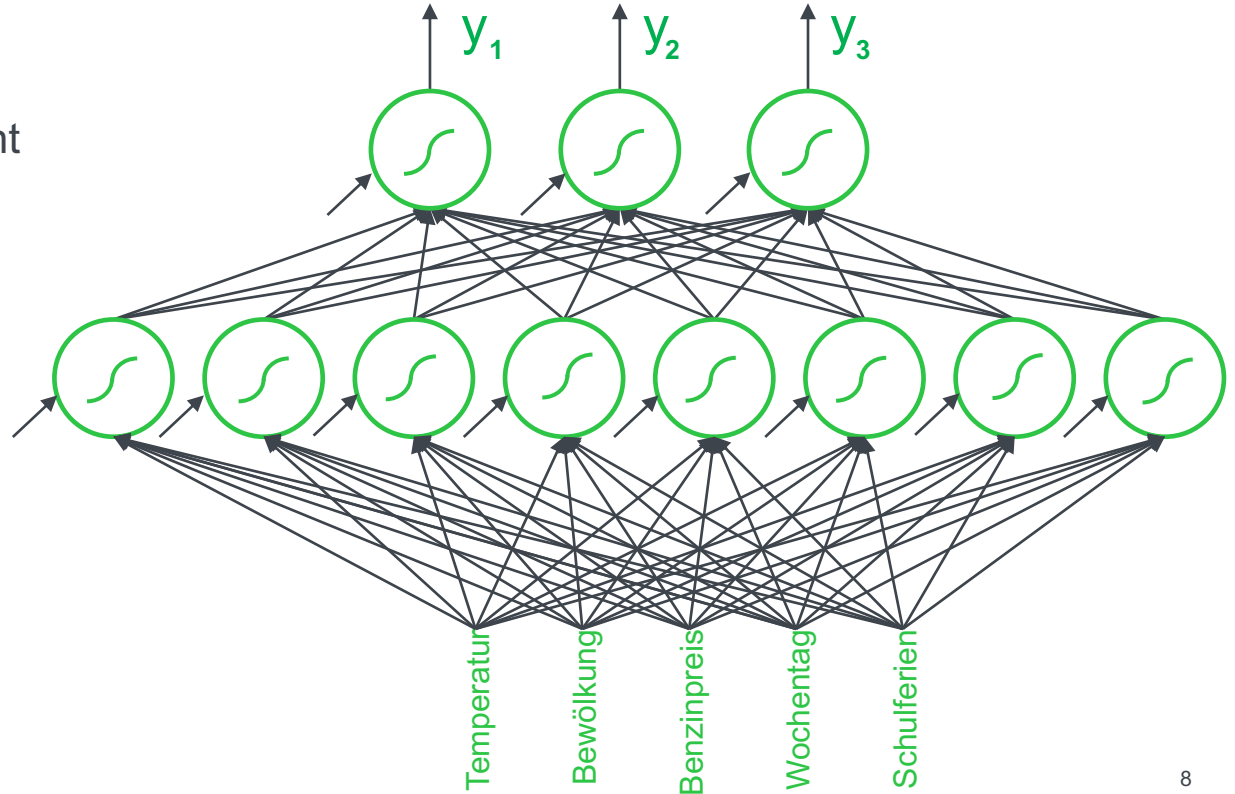
Inputs (Vektor)

Logits (Vektor)

## Logits in Schicht 1

- Bisher:
- Logits  $z^1$  der ersten Schicht  

$$z^1 = W^1 * x$$
- Matrixmultiplikation
- Es fehlt: der Bias  $b_i^1$





# Logits in Schicht 1

$$W^1 * x + b^1 =$$

$$w_{11}^1 * x_1 + w_{12}^1 * x_2 + w_{13}^1 * x_3 + w_{14}^1 * x_4 + w_{15}^1 * x_5$$

$$w_{21}^1 * x_1 + w_{22}^1 * x_2 + w_{23}^1 * x_3 + w_{24}^1 * x_4 + w_{25}^1 * x_5$$

$$w_{31}^1 * x_1 + w_{32}^1 * x_2 + w_{33}^1 * x_3 + w_{34}^1 * x_4 + w_{35}^1 * x_5$$

$$w_{41}^1 * x_1 + w_{42}^1 * x_2 + w_{43}^1 * x_3 + w_{44}^1 * x_4 + w_{45}^1 * x_5$$

$$w_{51}^1 * x_1 + w_{52}^1 * x_2 + w_{53}^1 * x_3 + w_{54}^1 * x_4 + w_{55}^1 * x_5$$

$$w_{61}^1 * x_1 + w_{62}^1 * x_2 + w_{63}^1 * x_3 + w_{64}^1 * x_4 + w_{65}^1 * x_5$$

$$w_{71}^1 * x_1 + w_{72}^1 * x_2 + w_{73}^1 * x_3 + w_{74}^1 * x_4 + w_{75}^1 * x_5$$

$$w_{81}^1 * x_1 + w_{82}^1 * x_2 + w_{83}^1 * x_3 + w_{84}^1 * x_4 + w_{85}^1 * x_5$$

$$b_1^1$$

$$b_2^1$$

$$b_3^1$$

$$b_4^1$$

$$b_5^1$$

$$b_6^1$$

$$b_7^1$$

$$b_8^1$$

+

=

$$z_1^1$$

$$z_2^1$$

$$z_3^1$$

$$z_4^1$$

$$z_5^1$$

$$z_6^1$$

$$z_7^1$$

$$z_8^1$$

$$z^1 = W^1 * x + b^1$$

**Die Logits einer Schicht werden mithilfe einer Matrixmultiplikation berechnet.**

**Die Gewichtematrix der Schicht wird mit dem Vektor aller Eingänge dieser Schicht multipliziert. Anschließend wird noch der Vektor für die Bias addiert.**

## Parameter der 1. Schicht

- Anzahl der Verbindungen von Input zu Neuron:
  - Anzahl Inputs \* Anzahl Neuronen
- Anzahl der Bias:
  - Anzahl Neuronen

Anzahl Parameter  
(n Neuronen, m Inputs)

$$n * m + n = (m + 1) * n$$

## Ausgänge von Schicht 1

- Vektor, hier Bezeichnung  $a^1$   
(um von den Ausgängen  $y$  des ganzen Netzwerks zu unterscheiden)
- Entsteht durch Anwendung der Aktivierungsfunktion auf die Logits  $z^1$ , also z.B. bei Sigmoid-Aktivierung:

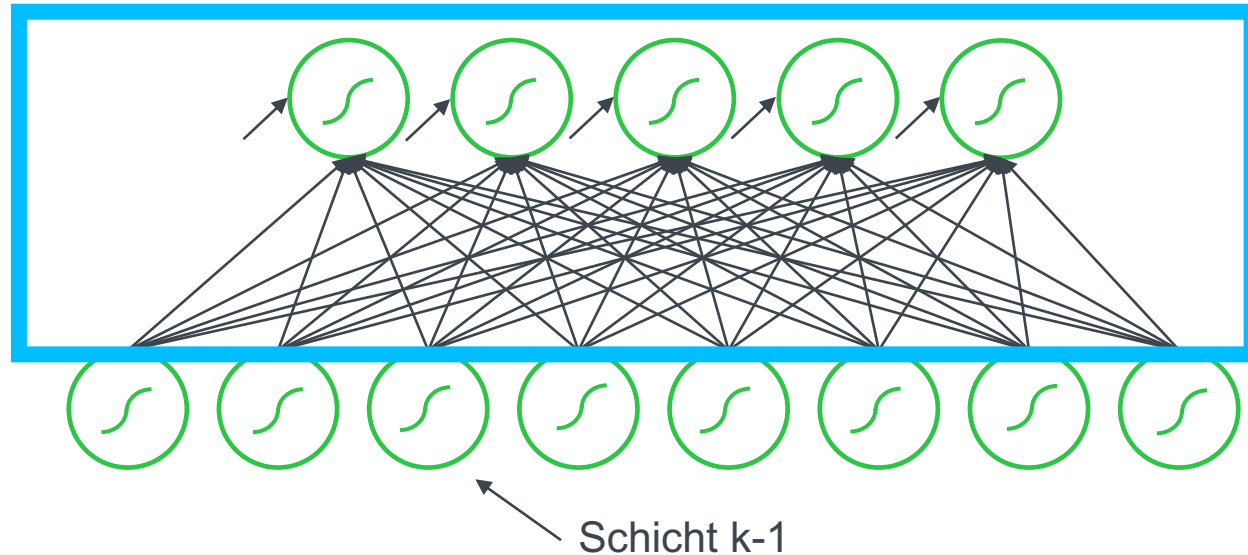
$$a^1 = \sigma(z^1) = \sigma(W^1x + b^1)$$

- Bildet den Eingangsvektor für die nächste Schicht
- Aktivierungsfunktion enthält keine weiteren lernbaren Parameter

## Höhere Schichten

- Schicht k
- Eingänge kommen aus voriger Schicht
- Gewichtematrix  $W^k$
- Mit Gewichten  $w_{ij}^k$  für Verbindungen von Neuron j aus Schicht k-1 zu Neuron i
- Bias in Vektor  $b^k$

### Schicht k



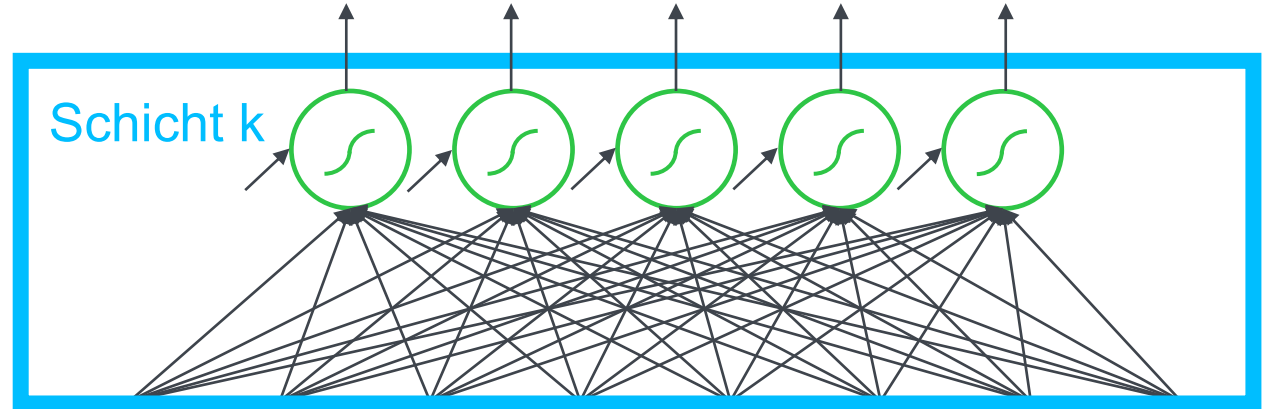
## Parameter in Schicht k

- Wie Schicht 1
- Andere Form der Matrix
  - 5 Zeilen (Anzahl Neuronen in Schicht k)
  - 8 Spalten (Anzahl Neuronen in Schicht k-1)
- Bias-Vektor mit 5 Zeilen (Anzahl Neuronen in Schicht k)

	Von Neuron 1	Von Neuron 2	Von Neuron 3	Von Neuron 4	Von Neuron 5	Von Neuron 6	Von Neuron 7	Von Neuron 8	
Zu Neuron 1	$w_{11}^k$	$w_{12}^k$	$w_{13}^k$	$w_{14}^k$	$w_{15}^k$	$w_{16}^k$	$w_{17}^k$	$w_{18}^k$	$b_1^k$
Zu Neuron 2	$w_{21}^k$	$w_{22}^k$	$w_{23}^k$	$w_{24}^k$	$w_{25}^k$	$w_{26}^k$	$w_{27}^k$	$w_{28}^k$	$b_2^k$
Zu Neuron 3	$w_{31}^k$	$w_{32}^k$	$w_{33}^k$	$w_{34}^k$	$w_{35}^k$	$w_{36}^k$	$w_{37}^k$	$w_{38}^k$	$b_3^k$
Zu Neuron 4	$w_{41}^k$	$w_{42}^k$	$w_{43}^k$	$w_{44}^k$	$w_{45}^k$	$w_{46}^k$	$w_{47}^k$	$w_{48}^k$	$b_4^k$
Zu Neuron 5	$w_{51}^k$	$w_{52}^k$	$w_{53}^k$	$w_{54}^k$	$w_{55}^k$	$w_{56}^k$	$w_{57}^k$	$w_{58}^k$	$b_5^k$

Anzahl Parameter  
(n Neuronen, m Inputs)  
 $n * m + n = (m + 1) * n$

## Ausgänge von Schicht k



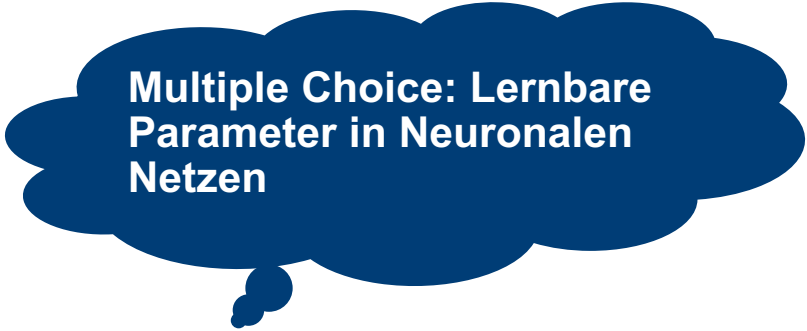
- Logits

$$z^k = W^k * a^{k-1} + b^k$$

- Ausgänge

(z.B. bei Sigmoid-Aktivierung)

$$a^k = \sigma(z^k) = \sigma(W^k * a^{k-1} + b^k)$$



**Multiple Choice: Lernbare  
Parameter in Neuronalen  
Netzen**





**Single Choice: Anzahl von  
Parametern in einer Schicht**



**Drag and Drop: Welche Matrix  
passt zu welcher Schicht?**

## Anwendung des Netzwerkes: der Forward Pass

- Ausgänge Schicht 1:

$$a^1 = \sigma(W^1x + b^1)$$

- Logits Schicht 2:

$$z^2 = W^2 * a^1 + b^2$$

$$= W^2 * \sigma(W^1x + b^1) + b^2$$

- Ausgang Schicht 2

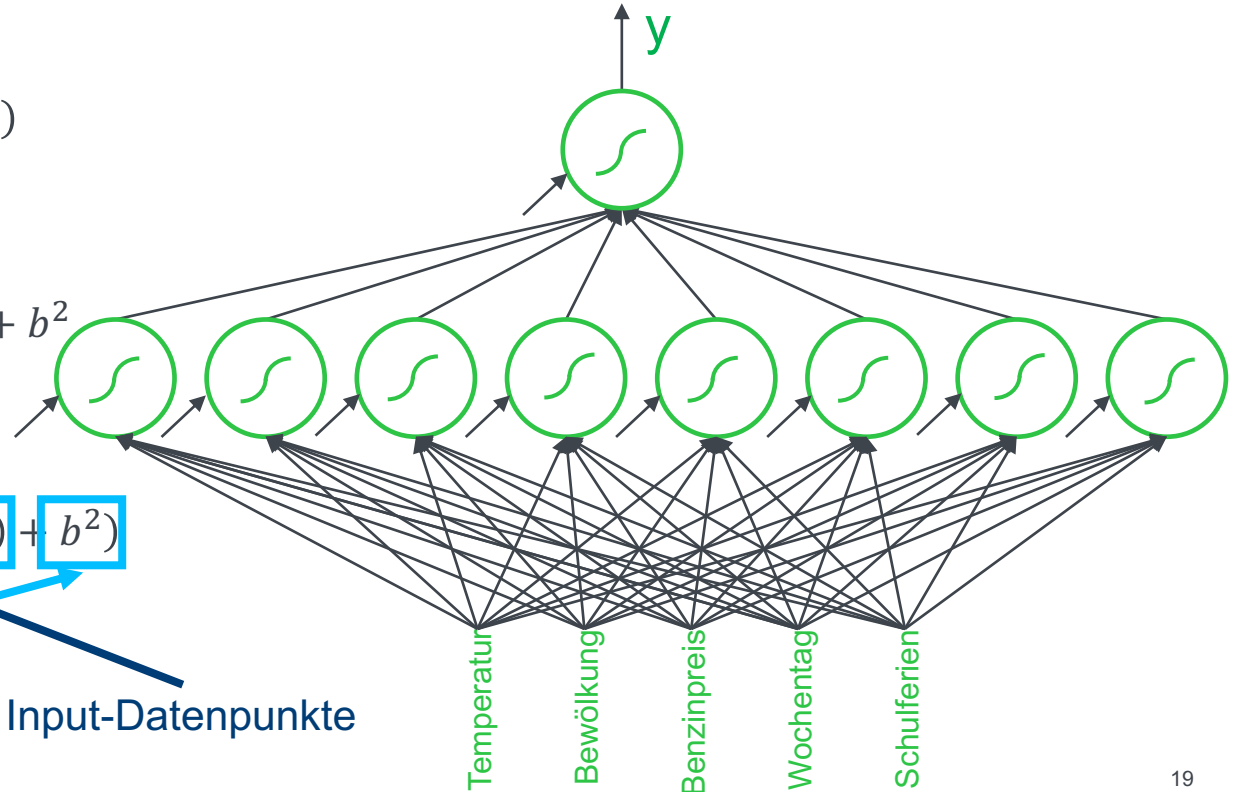
$$y = \sigma(z^2)$$

$$= \sigma(W^2 * \sigma(W^1 * x + b^1) + b^2)$$

Vorhersage

Gelernte Parameter

Input-Datenpunkte



Mehr zu Softmax: siehe Aktivierungsfunktionen

# Forward Pass an einem Beispiel

0	1	0	-1
1	-1	1	0

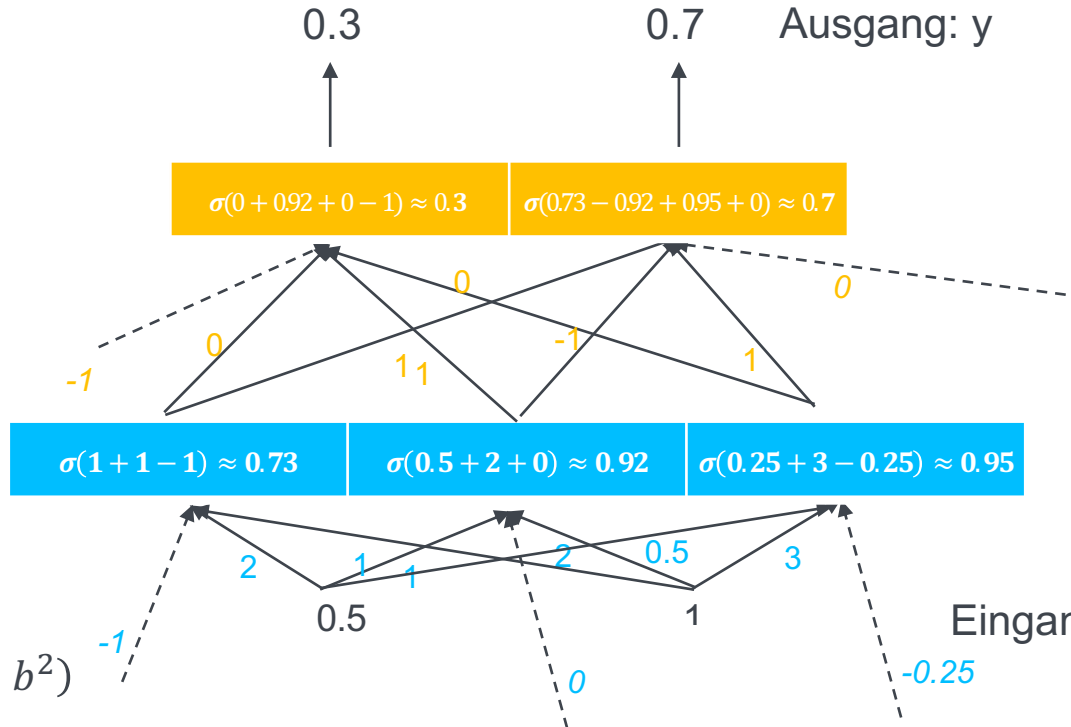
$W^2$        $b^2$

(Softmax-)  
Aktivierung  
Schicht 2  
Logits

2	1	-1
1	2	0
0.5	3	-0.25

$W^1$        $b^1$

(Sigmoid-)  
Aktivierung  
Schicht 1  
Logits



$$y = \sigma_{Softmax}(W^2 \sigma(W^1 x + b^1) + b^2)$$

**Der Forward Pass ist die schrittweise Berechnung der Ausgabe eines Neuronalen Netzes. Die Eingangsdaten sind Input für die unterste Schicht.**

**Für jede Schicht wird erst der Logit berechnet. Durch Anwendung der Aktivierungsfunktion bekommt man dann die Aktivierungswerte aller Neuronen der Schicht. Diese gehen als Input in die nächsthöhere Schicht.**

**Die Aktivierungswerte der obersten Schicht liefern das Gesamtergebnis.**

**Der Forward Pass kann durch eine mathematische Formel ausgedrückt werden – wie z.B. auch das Ergebnis einer linearen Regression.**

**Allerdings ist die Formel für den Forward Pass komplizierter: Sie beinhaltet mehrfache Multiplikationen mit Matrizen sowie die Anwendung nicht-linearer Funktionen.**



**Drag and Drop: Was ist was in der Formel zur Berechnung der Aktivierung?**

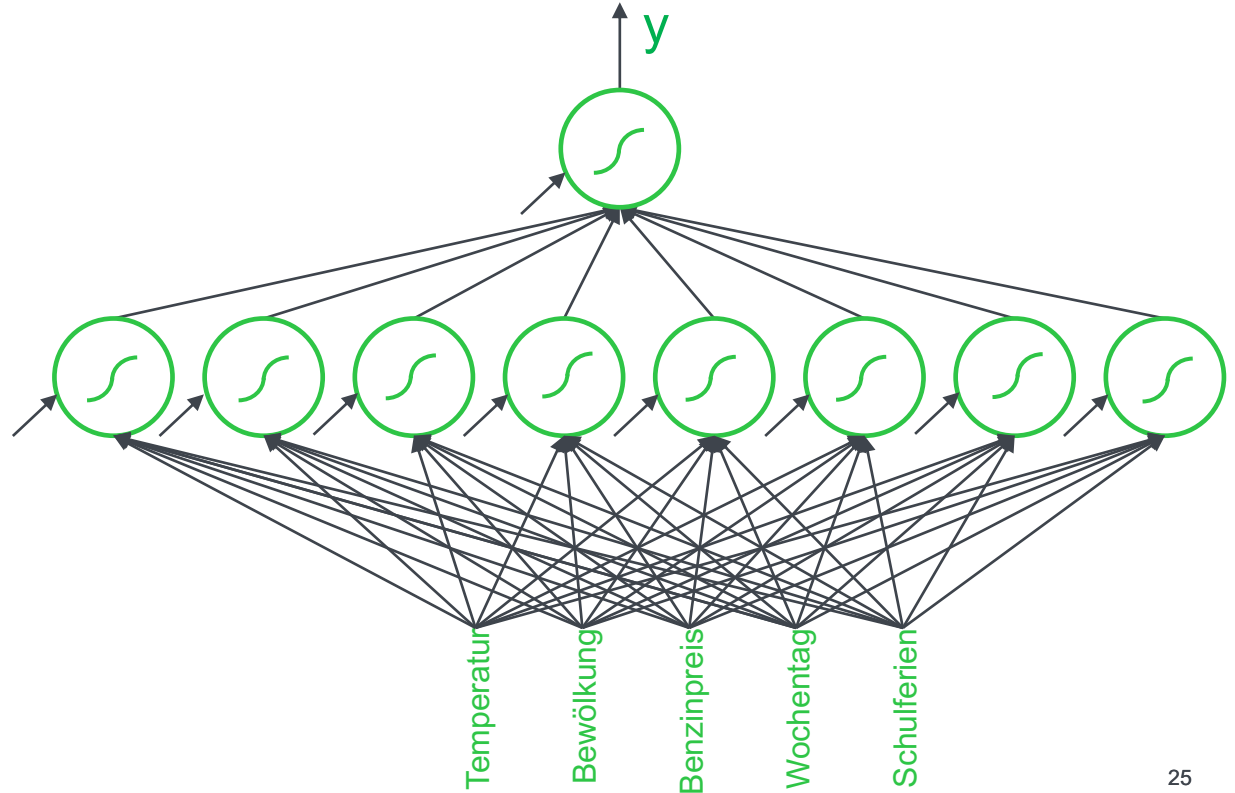


**Question Set: Forward pass**



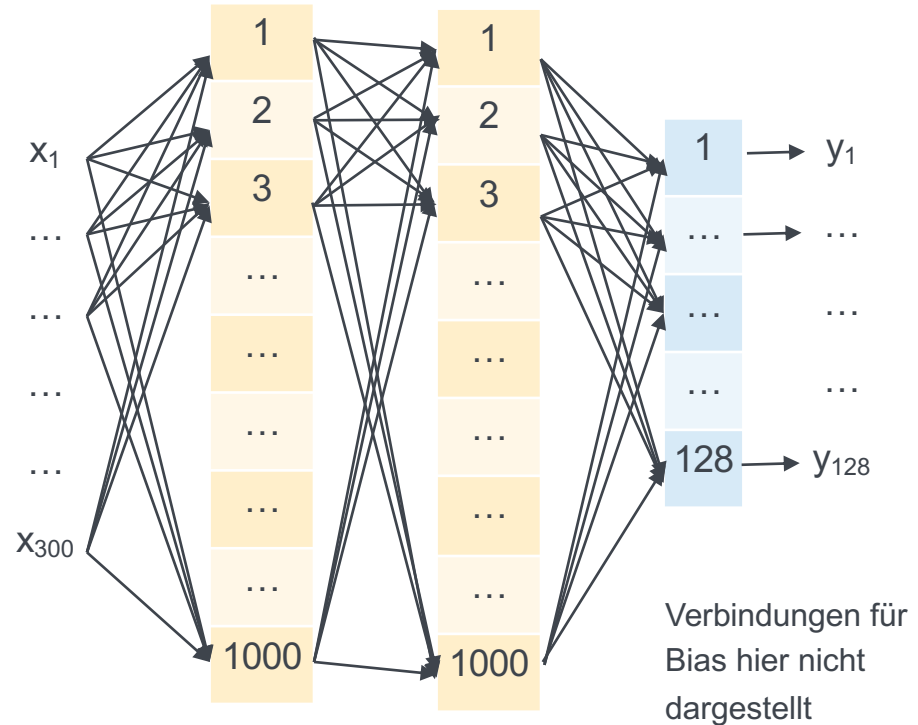
## Anzahl der Parameter im Beispiel

- Schicht 1
  - 5 Inputs, Bias, 8 Neuronen
  - $6 \cdot 8 = 48$  Parameter
- Schicht 2
  - 8 Inputs, Bias, 1 Neuron
  - $9 \cdot 1 = 9$  Parameter
- Insgesamt  $48 + 9 = 57$  Parameter



## Anzahl der Parameter in realistischerem Netzwerk

- Schicht 1
  - 300 Inputs, Bias, 1000 Neuronen
  - $301 \cdot 1000 = 301000$  Parameter
- Schicht 2
  - 1000 Inputs, Bias, 1000 Neuronen
  - $1001 \cdot 1000 = 1001000$  Parameter
- Schicht 3
  - 1000 Inputs, Bias, 128 Neuronen
  - $1001 \cdot 128 = 128128$  Parameter
- Insgesamt 1 430 128 Parameter



**In Neuronalen Netzen hat man häufig mehrere Millionen Parameter, die gelernt werden müssen.**

## Dr. Antje Schweitzer

Universität Stuttgart  
Institut für Maschinelle Sprachverarbeitung



**Universität Stuttgart**

Institut für Maschinelle Sprachverarbeitung  
Institut für Software Engineering



**IHK** Industrie- und Handelskammer  
Reutlingen

Reutlingen | Tübingen | Zollernalb



**IHK** Region Stuttgart



**IHK** Industrie- und Handelskammer  
Karlsruhe



# Lizenzbestimmungen

“Parameter in Neuronalen Netzen” von Antje Schweitzer, KI B<sup>3</sup> / Uni Stuttgart

Das Werk - mit Ausnahme der folgenden Elemente:

- Logos der Verbundpartner und des Förderprogramms
- im Quellenverzeichnis aufgeführte Medien

ist lizenziert unter:

 [CC BY 4.0 \(https://creativecommons.org/licenses/by/4.0/deed.de\)](https://creativecommons.org/licenses/by/4.0/deed.de)

(Namensnennung 4.0 International)

## Quellenverzeichnis

Titelfoto: [Diane Picchiottino \(https://unsplash.com/@diane\\_soko\)](https://unsplash.com/@diane_soko), ohne Titel, auf [Unsplash \(https://unsplash.com/photos/itHFvqW09yM\)](https://unsplash.com/photos/itHFvqW09yM), ist lizenziert unter [Unsplash-Lizenz \(https://unsplash.com/license\)](https://unsplash.com/license).

Bildausschnitt verändert.