

# KI-Systeme in Stromverteilnetzen

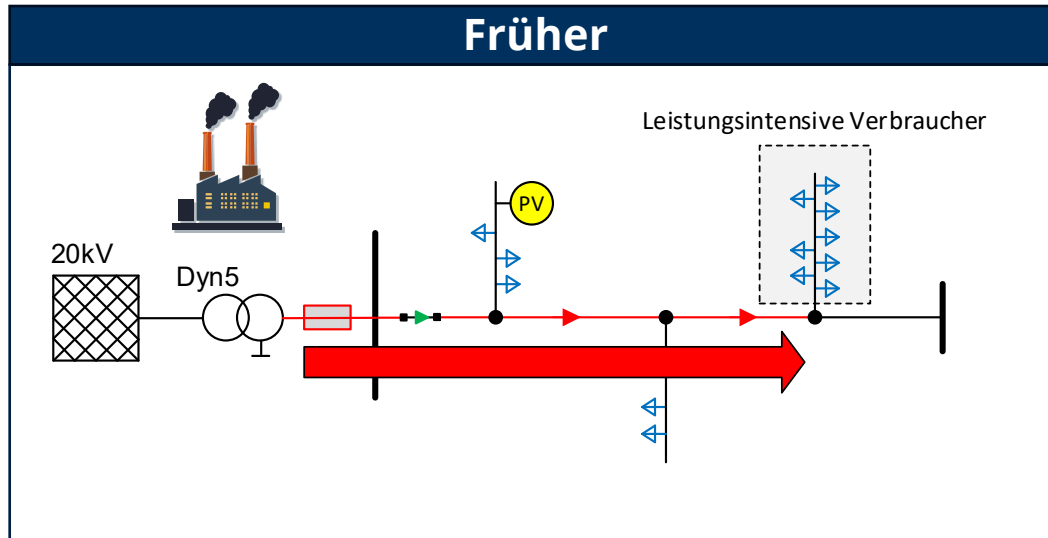
## Überwachung, Optimierung und Fehlerortung mit Künstlichen Neuronalen Netzen (KNN)



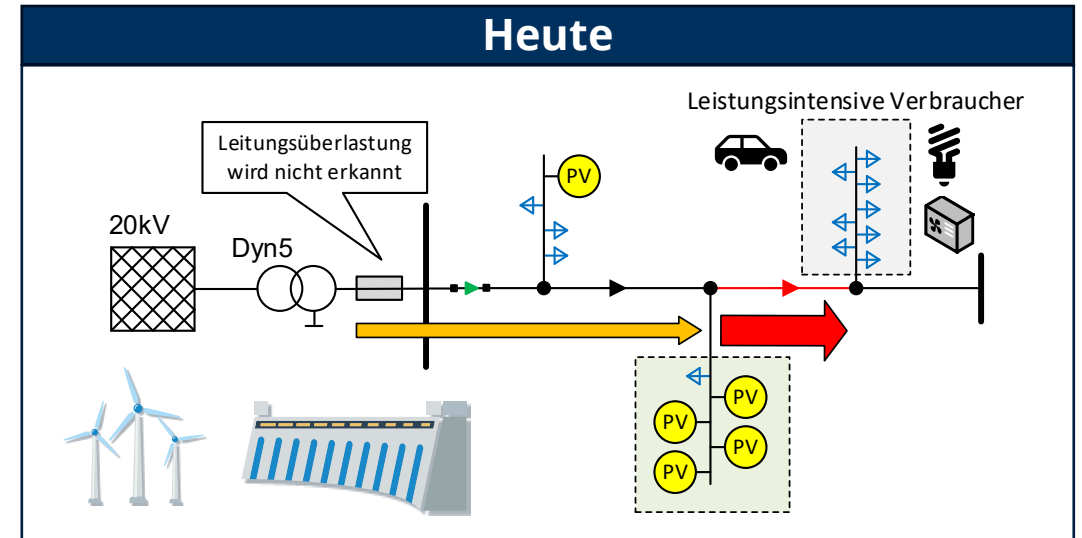
Technische Hochschule Mannheim, 9. Oktober 2025

Prof. Dr.-Ing. Michael Igel, Institut für elektrische Energiesysteme, htw saar

Dr.-Ing. Andreas Winter, energis-Netzgesellschaft mbH



- Leistungsfluss von der Netzeinspeisung zu den Verbrauchern
- Geringe Anzahl dezentraler Erzeugungsanlagen

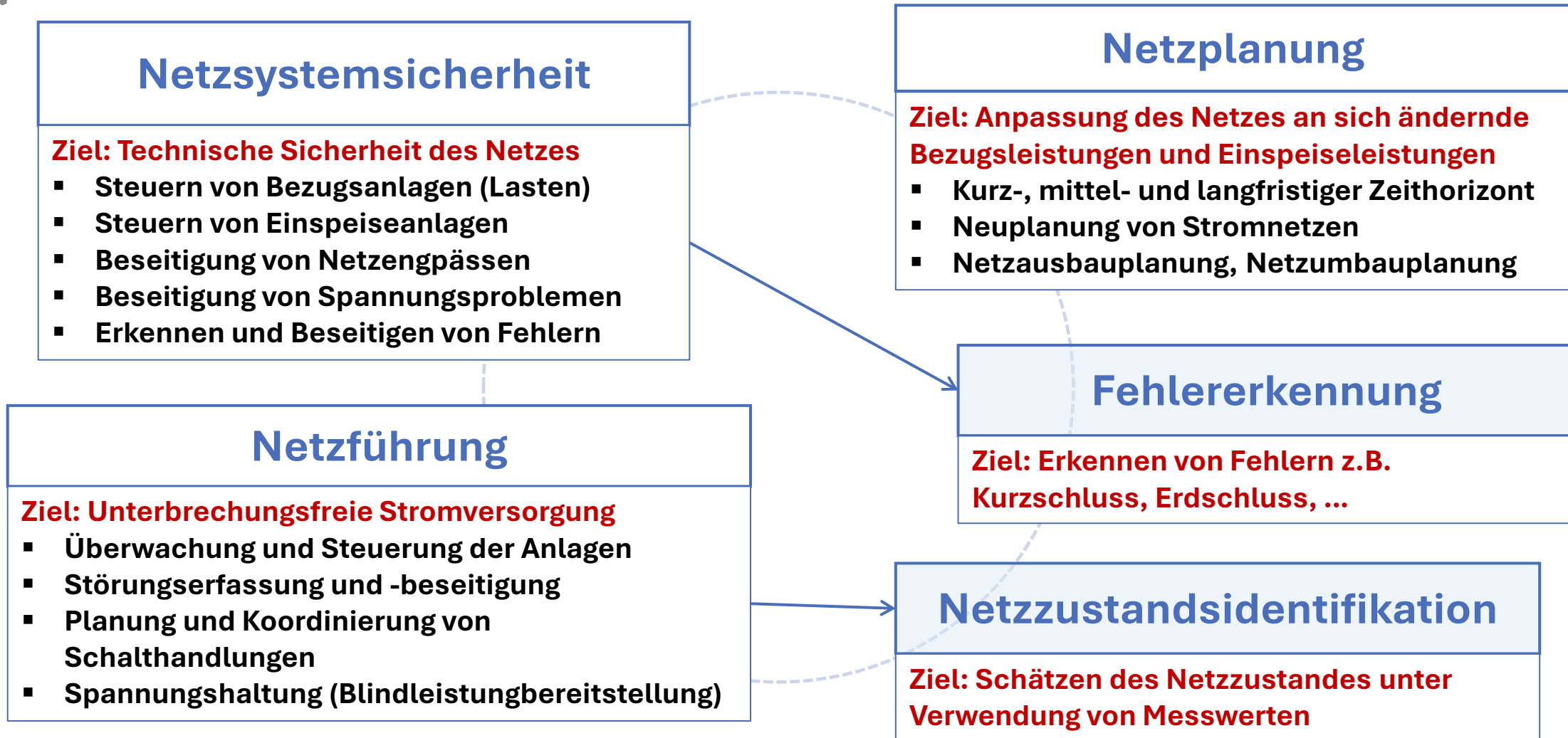


- Steigende Anzahl dezentraler Erzeugungsanlagen und neuer Verbraucher (E-Mobile und Wärmepumpen)
- Gefahr der inneren Überlastungen



Zukünftig sind hoch performante Verfahren erforderlich, um den Netzzustand eines elektrischen Verteilungsnetzes zu identifizieren und zu optimieren.





# Künstliche Intelligenz - Einige Grundlagen

## „KI markiert eine Epochenwende

Die Künstliche Intelligenz (KI) prägt unseren Alltag. So kommt sie beispielsweise bei der Sprachsteuerung von Navigationssystemen, der Gesichtserkennung zum Entsperren von Smartphones oder bei Produktvorschlägen beim Online-Shopping zum Einsatz.

KI kann komplexe Verkehrssituationen erkennen, beurteilen und Kollisionswahrscheinlichkeiten reduzieren. In der Industrie steuert KI Produktionsprozesse, lenkt Warenströmen und sagt einen drohenden Maschinenausfall voraus.

Auch die medizinische Diagnostik arbeitet mit KI: So können Machine-Learning-Algorithmen Tumore oder Schlaganfälle auf der Basis von CT-Scans erkennen oder Hautveränderungen klassifizieren.

Die Anwendungsmöglichkeiten der KI sind nahezu unbegrenzt. **Überall dort, wo große Datenmengen verarbeitet werden, kann KI zum Einsatz kommen und einen enormen Nutzen bringen.** Die technologische Tragweite der KI hat epochalen Charakter.“

Quelle: Technische Hochschule Würzburg-Schweinfurt

<https://ki.thws.de/thematik/anwendungsbereiche-der-ki>

ChatGPT

Google  
Gemini

Copilot

...

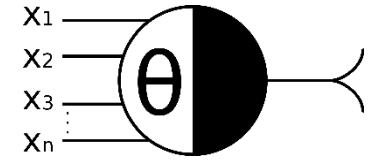
Generative KI

„Als **generative KI** werden Technologien wie ChatGPT oder Google Gemini, sogenannte Large Language Models (LLMs), bezeichnet. Sie haben die Fähigkeit, zum Beispiel Texte, Bilder, Audios oder Videos zu erzeugen. Generative KI basiert auf maschinellem Lernen, einem Teilgebiet der Künstlichen Intelligenz. Aus den Daten, mit der die KI gefüttert wird, kann sie neue Inhalte erstellen. Die Antworten der KI basieren dabei kurz gesagt auf Wahrscheinlichkeiten, die die KI aus dem eigenen Datenmaterial ermittelt.“

Einige Beispiele: **Hilfe beim Schreiben von Texten**

- Grußkarten, Reden, E-Mails, ...
- Übersetzen in andere Sprachen
- Verbessern von Texten
- ...

Quelle: ZDF heute <https://www.zdfheute.de/ratgeber/ki-kuenstliche-intelligenz-im-alltag-100.html>

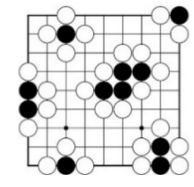


**McCulloch-Pitts-Neuron**



„AI“

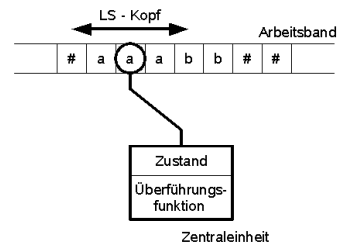
**KI schlägt Schachweltmeister**



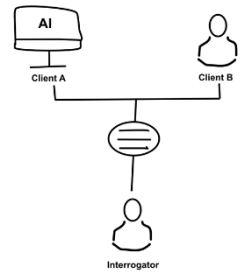
**KI schlägt Go-Meister**



**Turingmaschine**



**Turing-Test**



**ELIZA**

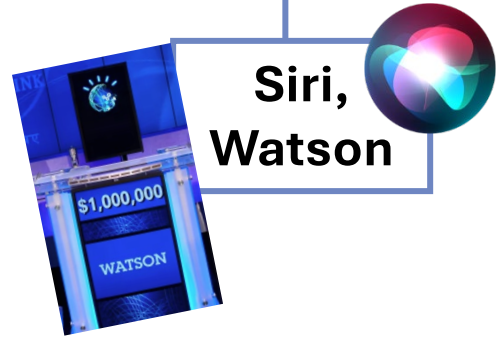
```

EEEEEE LL IIII ZZZZZZ AAAAA
EE LL II ZZ AA AA
EEEE LL II ZZ AAAAAA
EE LL II ZZ AA AA
EEEEEE LLLLLL IIII ZZZZZZ AA AA

rock Rogerian psychotherapist.
el program was described by Joseph Weizenbaum in 1966
mention by Norbert Landsteiner 2005.

something troubling you ?
re all alike.
is the connection, do you suppose ?
or always suspicious about something or other
    
```

**Siri, Watson**



- **„Künstliche Intelligenz ist die Fähigkeit einer Maschine, menschliche Fähigkeiten wie logisches Denken, Lernen, Planen und Kreativität zu imitieren.“** (Europäisches Parlament<sup>1</sup>)
- **„Künstliche Intelligenz (KI) ist ein Teilgebiet der Informatik. Sie imitiert menschliche kognitive Fähigkeiten, indem sie Informationen aus Eingabedaten erkennt und sortiert.“** (Fraunhofer-Institut für Kognitive Systeme IKS<sup>2</sup>)
- **„KI-Systeme sind Software- und Hardwaresysteme, die Künstliche Intelligenz nutzen, um in der physischen oder digitalen Welt "rational" zu handeln. Auf Grundlage von Wahrnehmung und Analyse ihrer Umgebung agieren sie mit einem gewissen Grad an Autonomie, um bestimmte Ziele zu erreichen.“** (Bundesamt für Sicherheit in der Informatik BSI<sup>3</sup>)

## Was ist Intelligenz ?

<sup>1</sup> <https://www.europarl.europa.eu/news/de/headlines/society/20200827STO85804/was-ist-kuenstliche-intelligenz-und-wie-wird-sie-genutzt>

<sup>2</sup> <https://www.iks.fraunhofer.de/de/themen/kuenstliche-intelligenz.html>

<sup>3</sup> [https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Kuenstliche-Intelligenz/kuenstliche-intelligenz\\_node.html](https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Kuenstliche-Intelligenz/kuenstliche-intelligenz_node.html)

## Künstliche Intelligenz



### Starke KI

➔ Superintelligenz

Besitzt die gleichen intellektuellen Fähigkeiten wie ein Mensch

- Logisches Denken
- Entscheidungsfähigkeit trotz Unsicherheit
- Planen und lernen
- Kommunikation in natürlicher Sprache
- Erreichen eines Ziels durch Kombination aller Fähigkeiten

➔ bis heute nicht entwickelt

### Schwache KI

Finden von konkreten Lösungen für bestimmte Probleme

➔ Schwach ≠ schlecht

➔ Nutzung von biologisch-medizinischen Erkenntnissen zur Erschaffung von effizienteren, kostengünstigeren Methoden

➔ Grundlage sind mathematisch-algorithmische Verfahren

➔ **Anforderung:** Zahlen sollen mit einer „Maschine“ addiert werden

## Mathematisch-algorithmische Lösung

- Die Lösung wird durch Mathematik beschrieben.

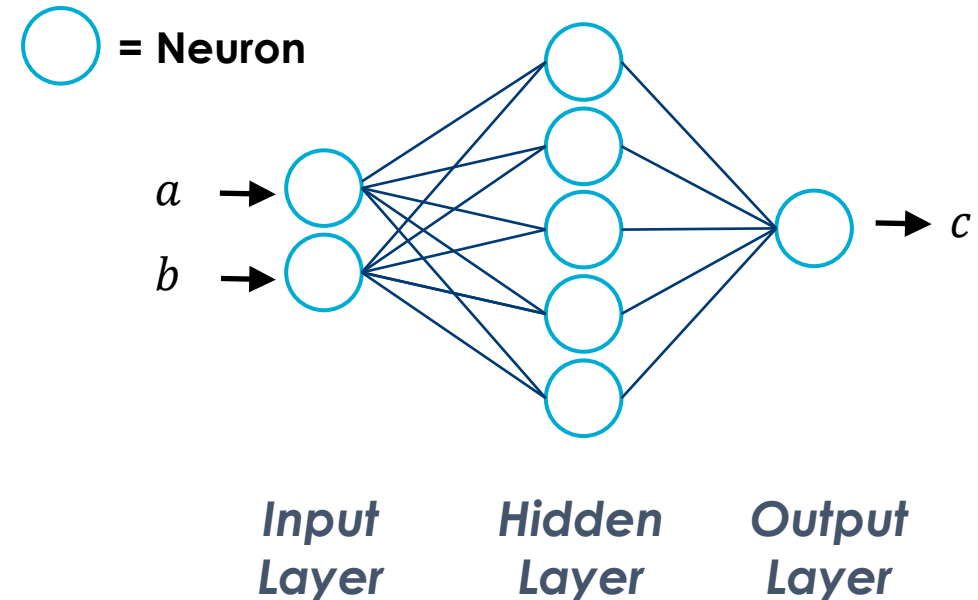
$$c = a + b$$

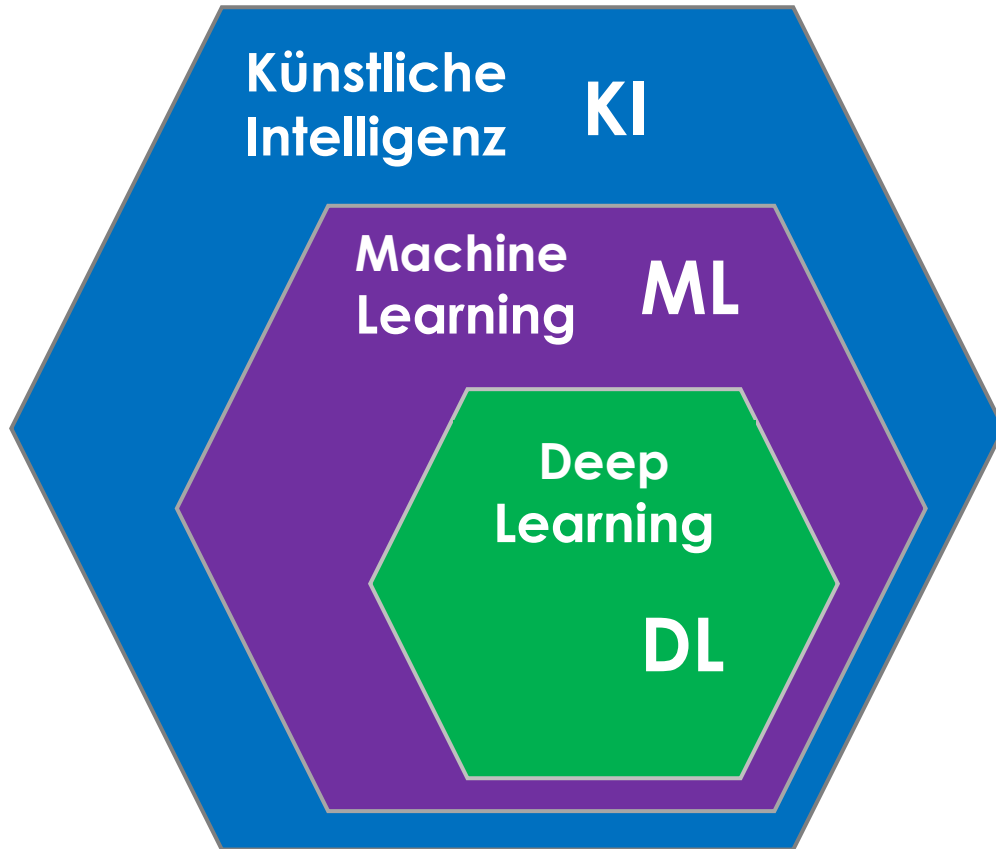
- Ein Algorithmus wird mit Hilfe einer Programmiersprache implementiert.

```
double a = 5;
double b = 3;
double c = a + b;
printf ("c = %f", c);
```

## Datenbasierte Lösung

- Die Lösung erfolgt durch ein **Künstliches Neuronales Netz**.





- Die **Künstliche Intelligenz (KI)** ist ein **Teilgebiet der Informatik** und verfolgt das Ziel bestimmte **intelligente Entscheidungsstrukturen von Menschen automatisiert nachzubilden**.
- **Machine Learning (ML)** ist ein Oberbegriff für die **maschinelle Generierung von Wissen** aus bereits vorliegenden Daten und Erfahrungen.
- **Deep Learning (DL)** ist eine Methode des ML, die an der **Funktionsweise des menschlichen Gehirns** inspiriert ist.

## Bestärkendes Lernen

### ➔ Keine Trainingsdaten

- Entwickeln einer **Lösungsstrategie** für ein Problem durch Interaktion mit der Umgebung
- Trial & Error: Lernen durch Belohnung und Bestrafung
- ➔ **Lernen durch Feedback**
- Vergleichbar mit dem menschlichen Lernen

## Unüberwachtes Lernen

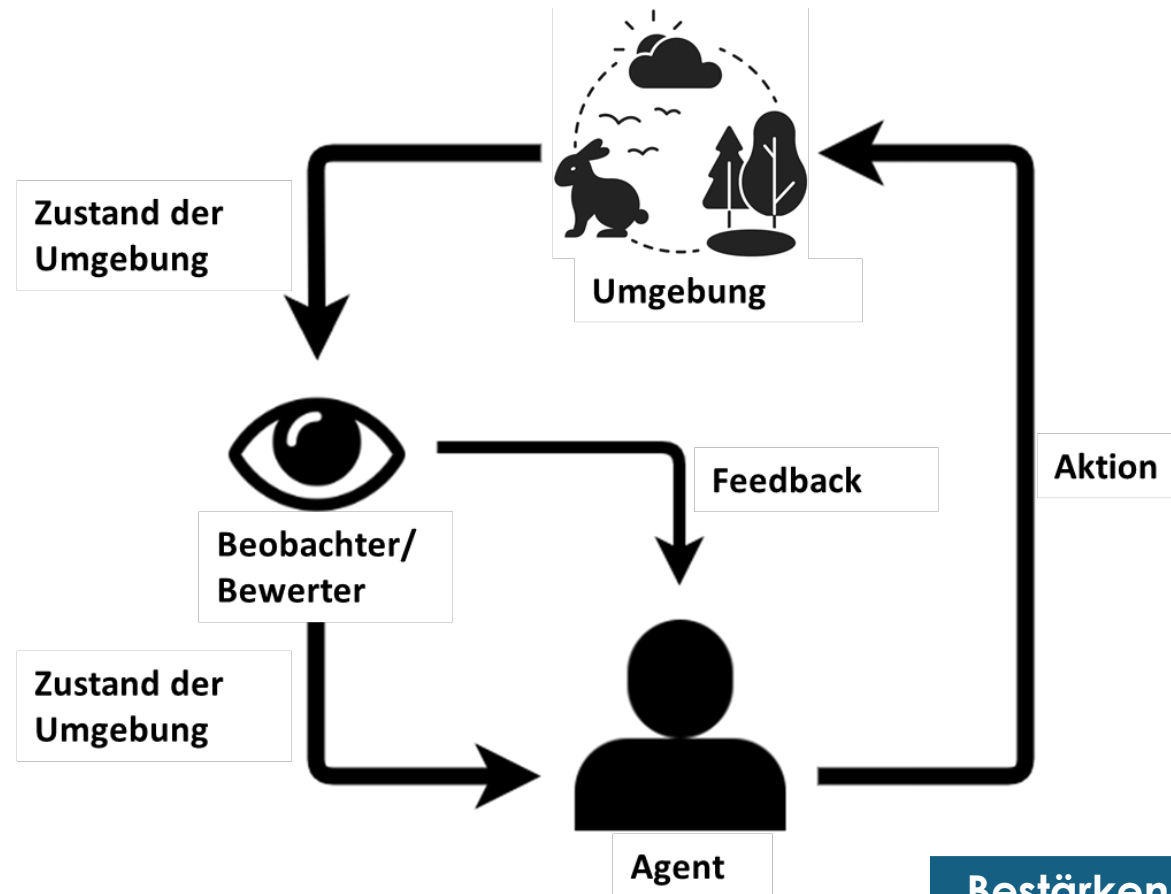
- Eingangsdaten: nicht-kategorisierte Daten  
➔ **ohne Lösungsmenge**
- Suchen von **Mustern** und **Zusammenhängen**
- **Clustering**: Finden des Musters zur Einteilung der Daten
- **Assoziationsanalyse**: Extraktion von Regeln aus Datensätzen
- **Dimensionsreduktion**: Verringern der Komplexität des Datensatzes ohne Verlust an Informationsgehalt

## Überwachtes Lernen

- Trainingsdatensatz: Input-/Output-Datenpaare  
➔ **Lösung ist vorgegeben**
- Suchen einer **Funktion  $y=f(x)$** , um Inputvariablen mit Outputvariablen zu verknüpfen
- ➔ **Klassifikation**: Ausgabe ist diskret, Unterteilung in Klassen
- **Regression**: Ausgabevariable ist numerisch

## Bestärkendes Lernen

- Keine Trainingsdaten
- Entwickeln einer Lösungsstrategie für ein Problem durch Interaktion mit der Umgebung
- Trial & Error: Lernen durch Belohnung und Bestrafung
- ➔ **Lernen durch Feedback**
- Vergleichbar mit dem menschlichen Lernen

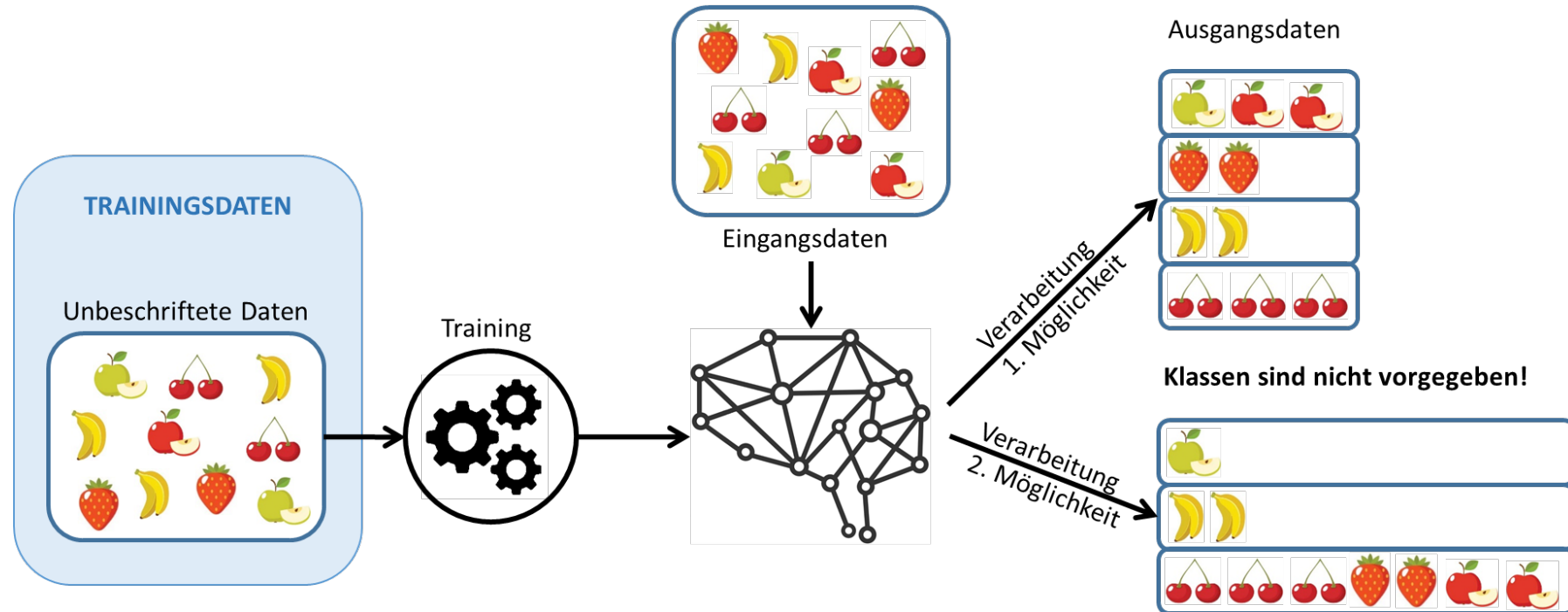


## Bestärkendes Lernen

**Anwendung:** Einparkhilfe,  
Dressur, autonomes Fahren

## Unüberwachtes Lernen

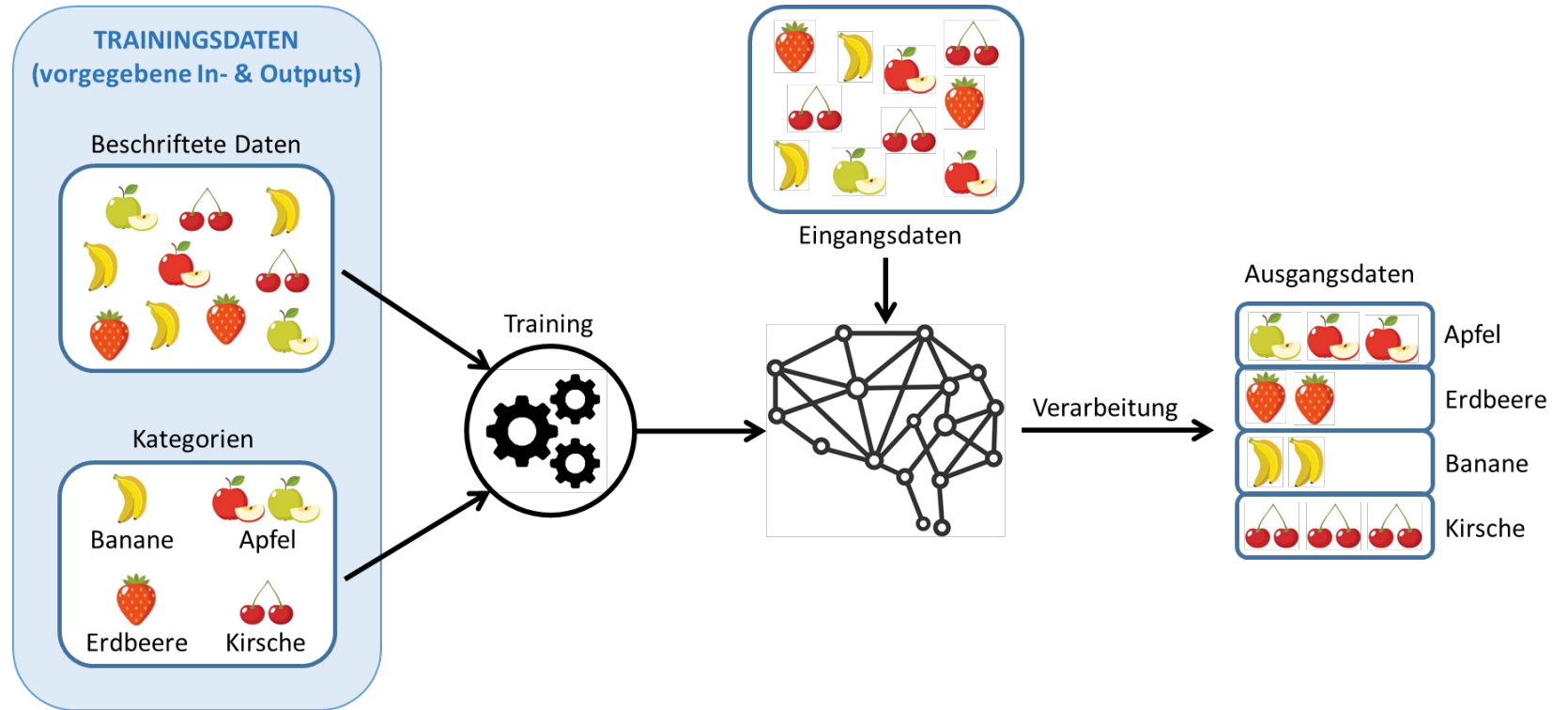
- Eingangsdaten: nicht-kategorisierte Daten  
 ➔ **ohne Lösungsmenge**
- Suchen von Mustern und Zusammenhängen
- **Clustering:** Finden des Musters zur Einteilung der Daten
- **Assoziationsanalyse:** Extraktion von Regeln aus Datensätzen
- **Dimensionsreduktion:** Verringern der Komplexität des Datensatzes ohne Verlust an Informationsgehalt



Clustering	Assoziationsanalyse	Dimensionsreduktion
<p><b>Anwendung:</b> Finden von Anomalien, genbasierte Krebsstudien, Kundengruppierungen für gezielte Werbung</p>	<p><b>Beispiel:</b> 100 Kinobesucher, 50 davon kaufen Popcorn, 25 der Popcornkäufer kaufen auch Getränke</p> <p><b>Anwendung:</b> „wird häufig zusammen gekauft“, Marketingstrategien</p>	<p><b>Anwendung:</b> Datenaufbereitung, Herausfiltern von redundanten bzw. nutzlosen Daten</p>

## Überwachtes Lernen

- Trainingsdatensatz: Input-/ Output-Datenpaare  
 ➔ **Lösung ist vorgegeben**
- Suchen einer Funktion  $y = f(x)$ , um Inputvariablen mit Outputvariablen zu verknüpfen
- **Klassifikation:** Ausgabe ist diskret, Unterteilung in Klassen
- **Regression:** Ausgabevariable ist numerisch

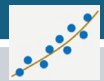


### Klassifikation



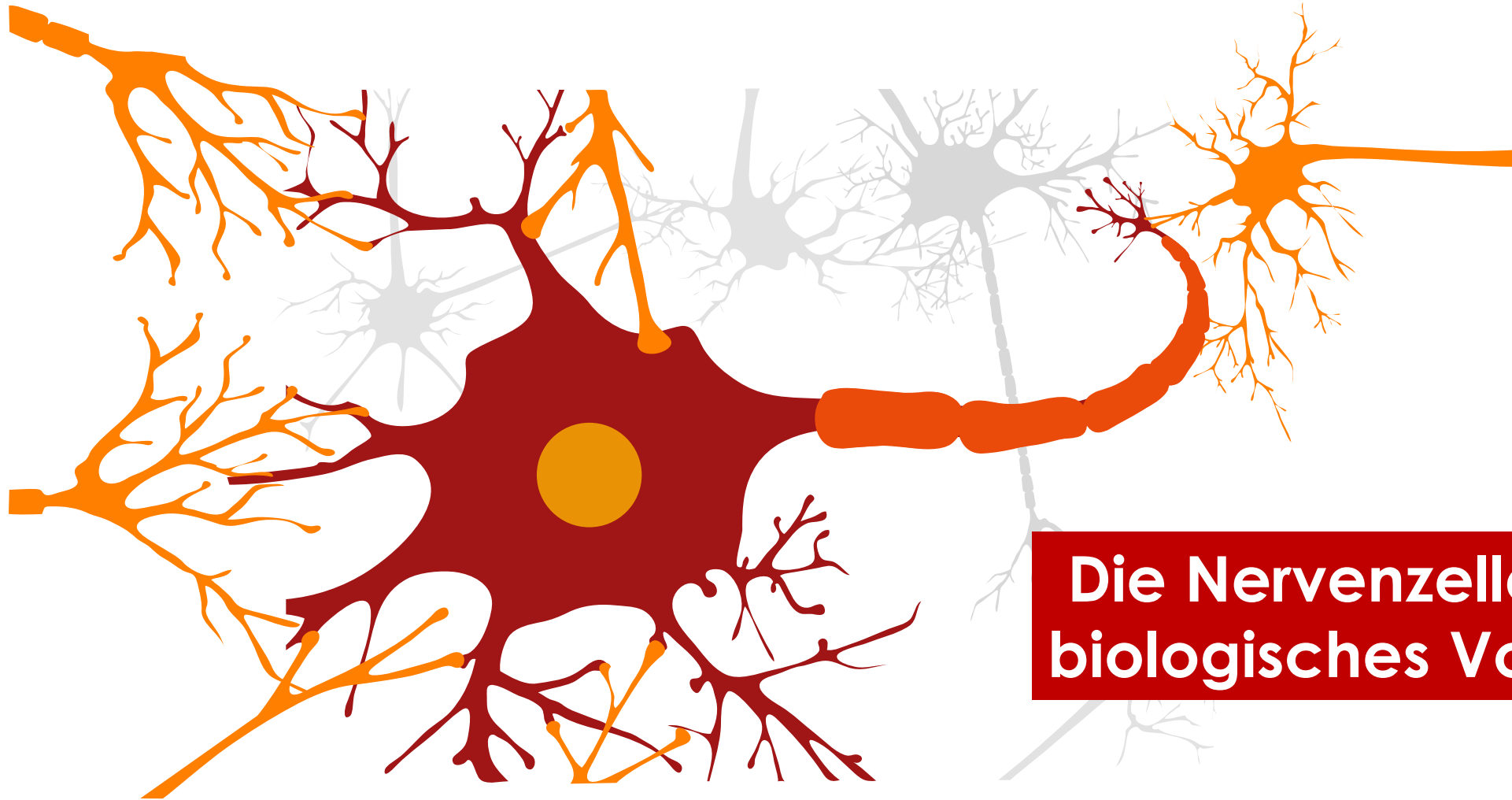
- **Ausgabevariable:** diskret
- **Daten:** diskret, linear separabel
- **gesucht:** Entscheidungsgrenze zur Unterteilung der Daten
- **Anwendung:** Spamfilter, Bild-/ Krebserkennung

### Regression

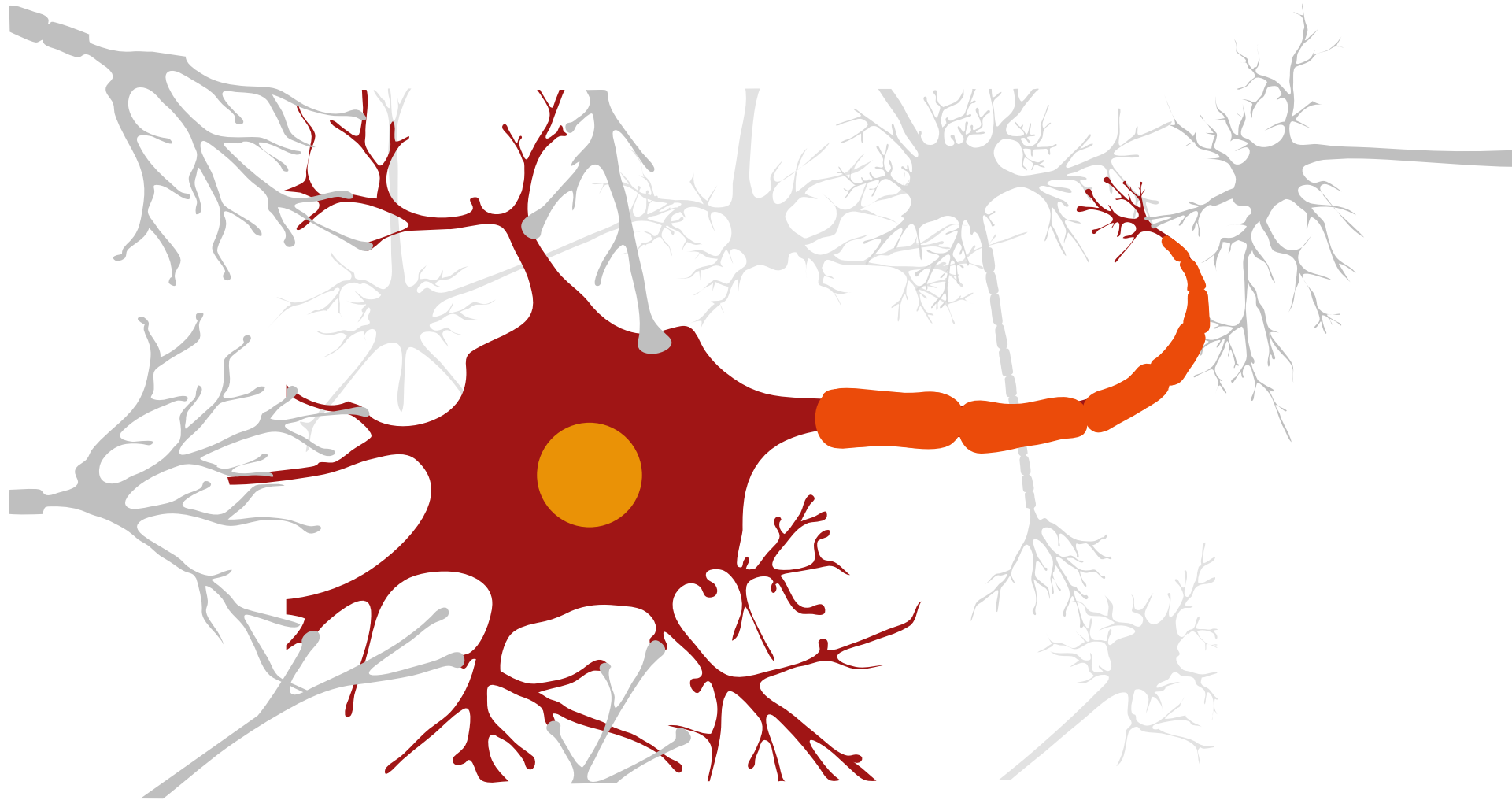


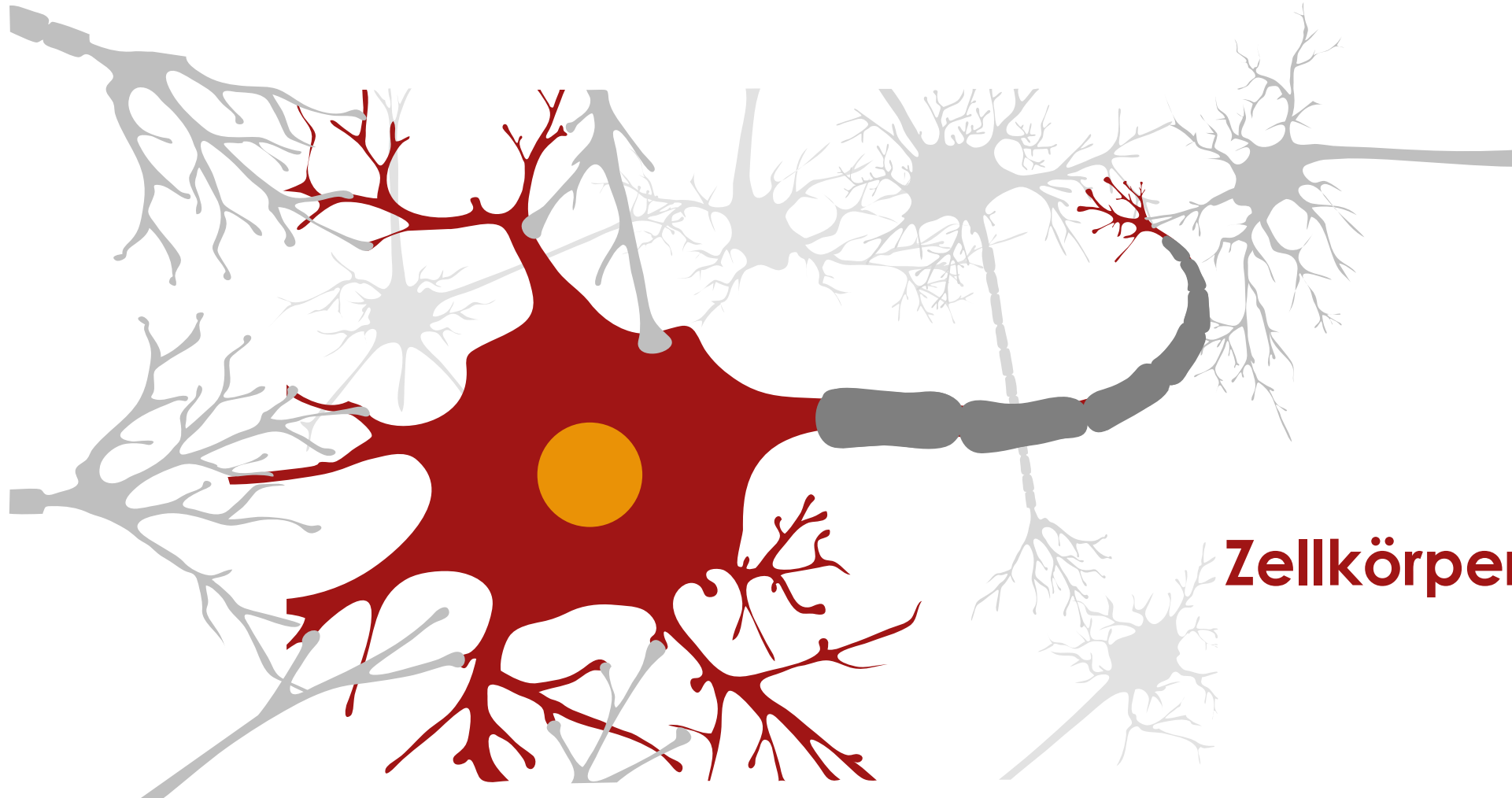
- **Ausgabevariable:** numerisch
- **Daten:** stetig
- **gesucht:** Kurve, die die Ergebnisse am besten vorhersagen kann
- **Anwendung:** Wetter-/ Stromverbrauchsvorhersage, Ausfallwahrscheinlichkeiten

# Das Neuron als biologisches Vorbild

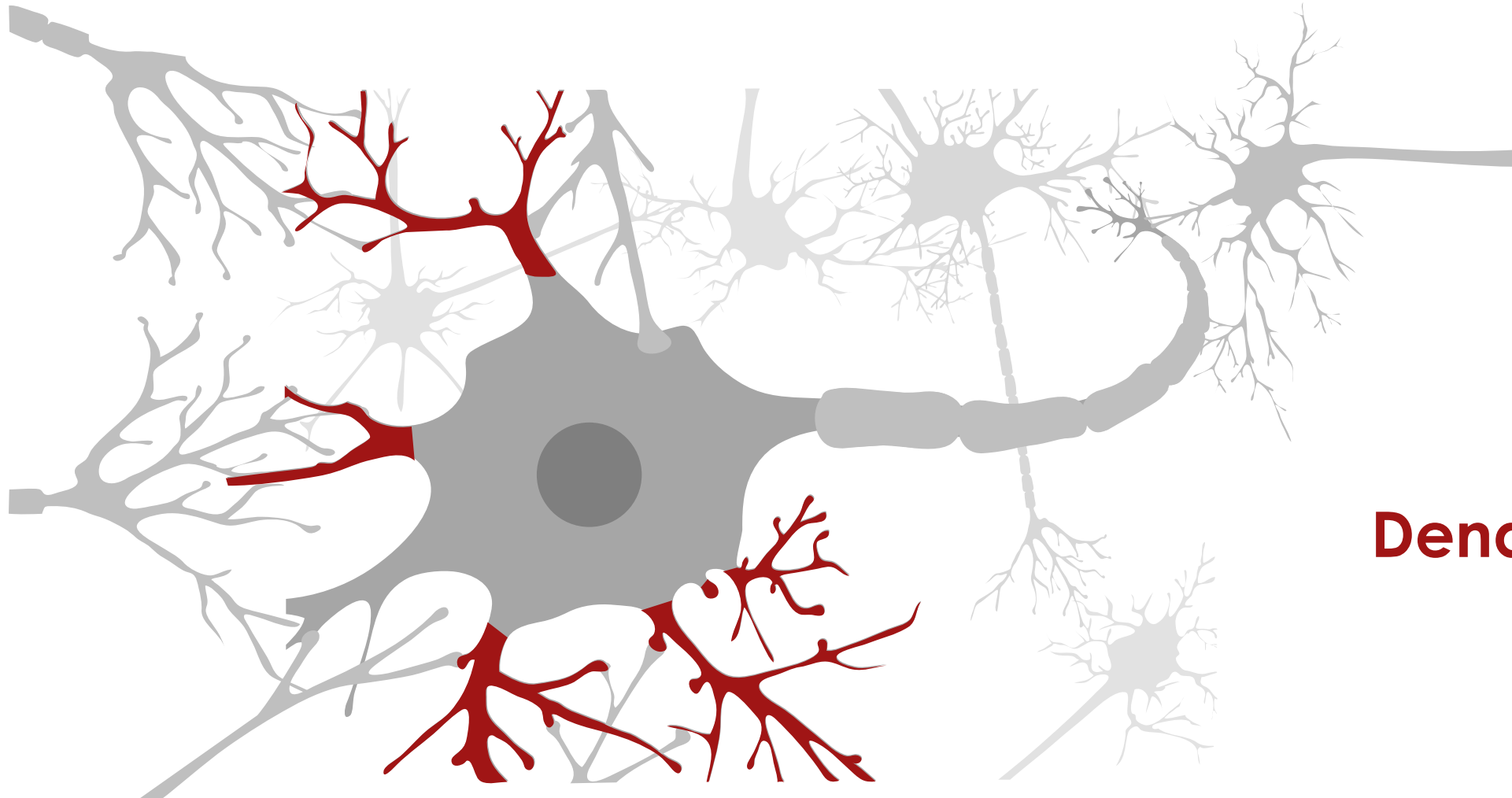


**Die Nervenzelle als  
biologisches Vorbild**

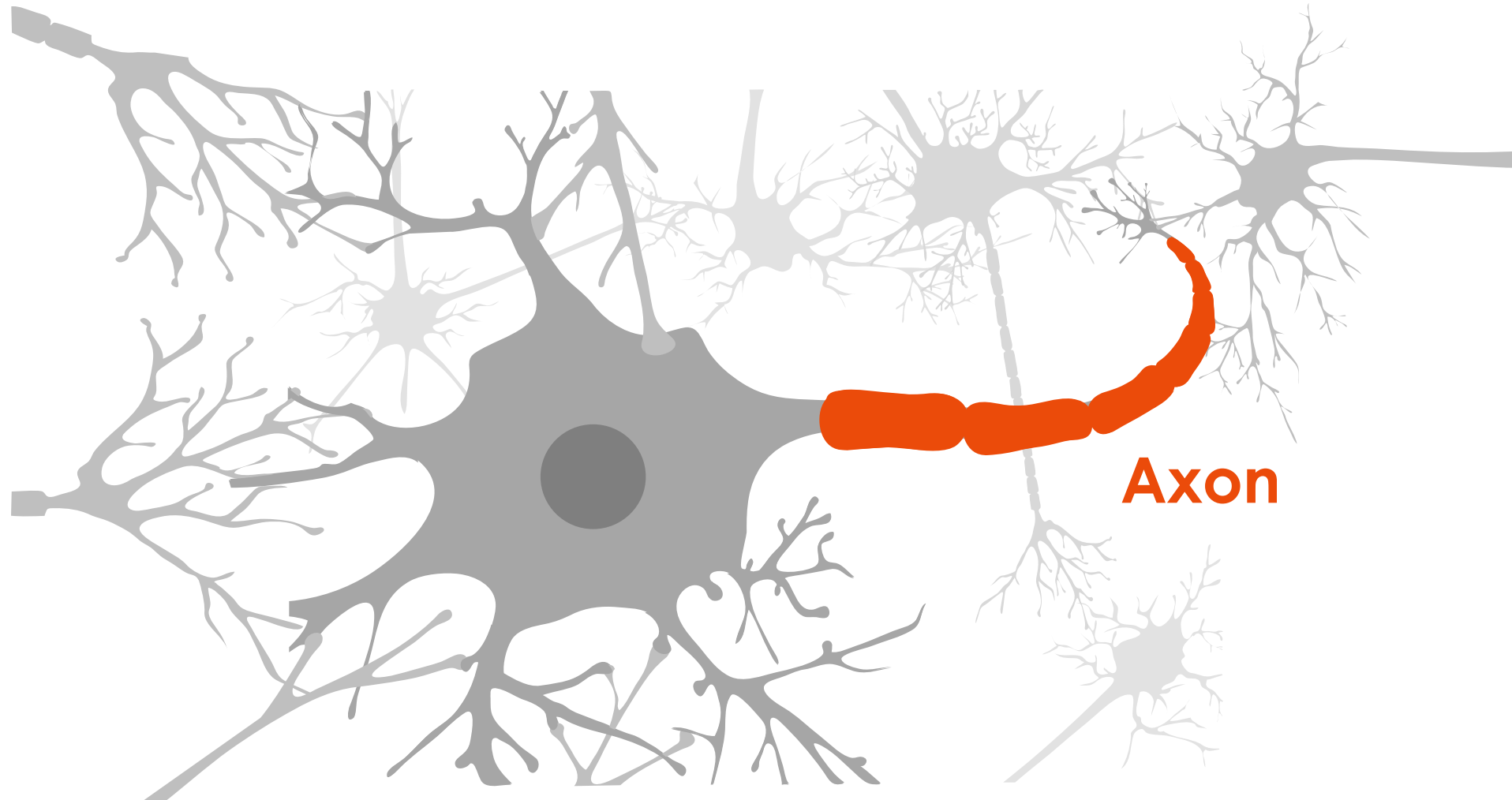


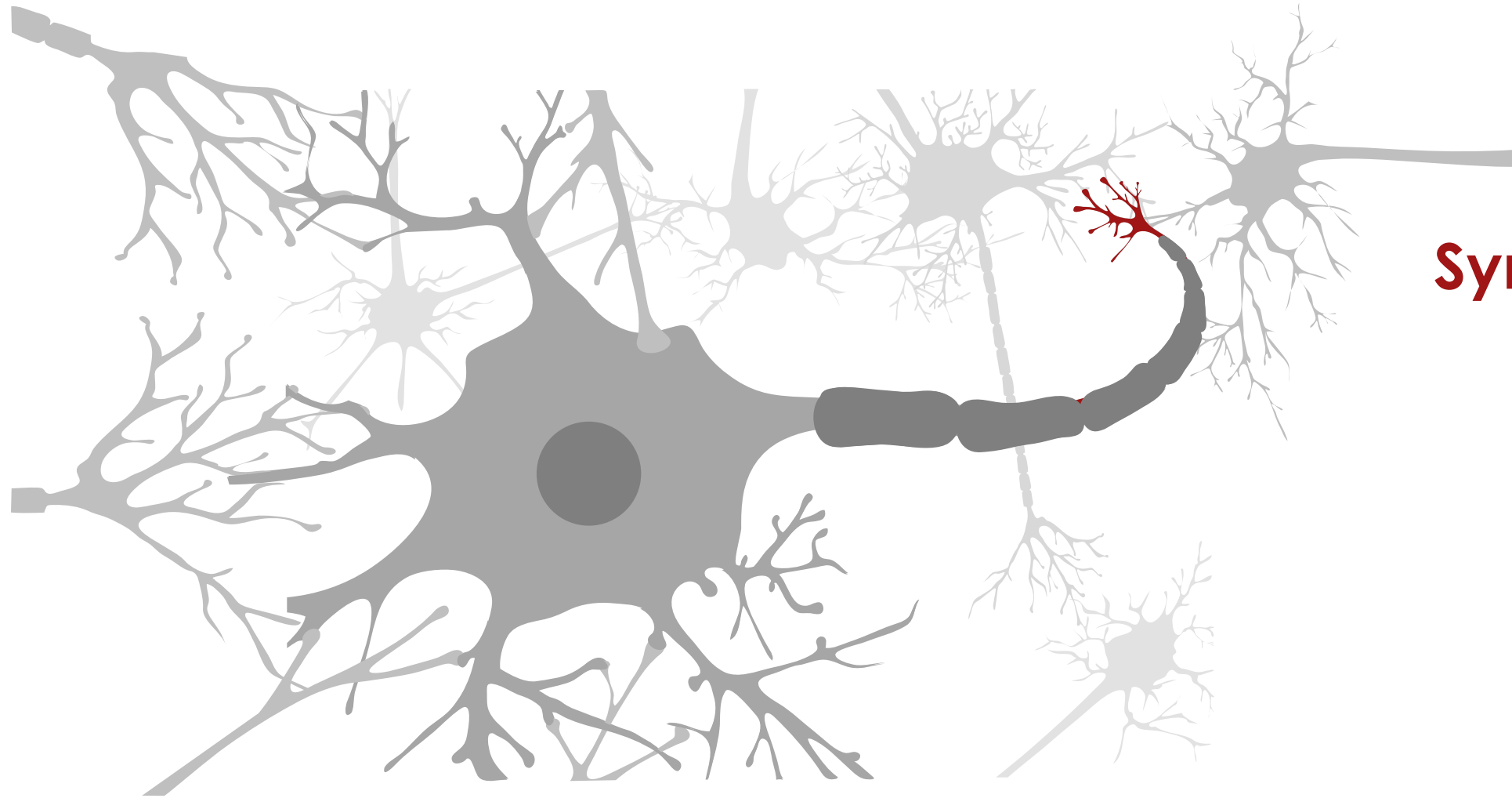


**Zellkörper = Soma**

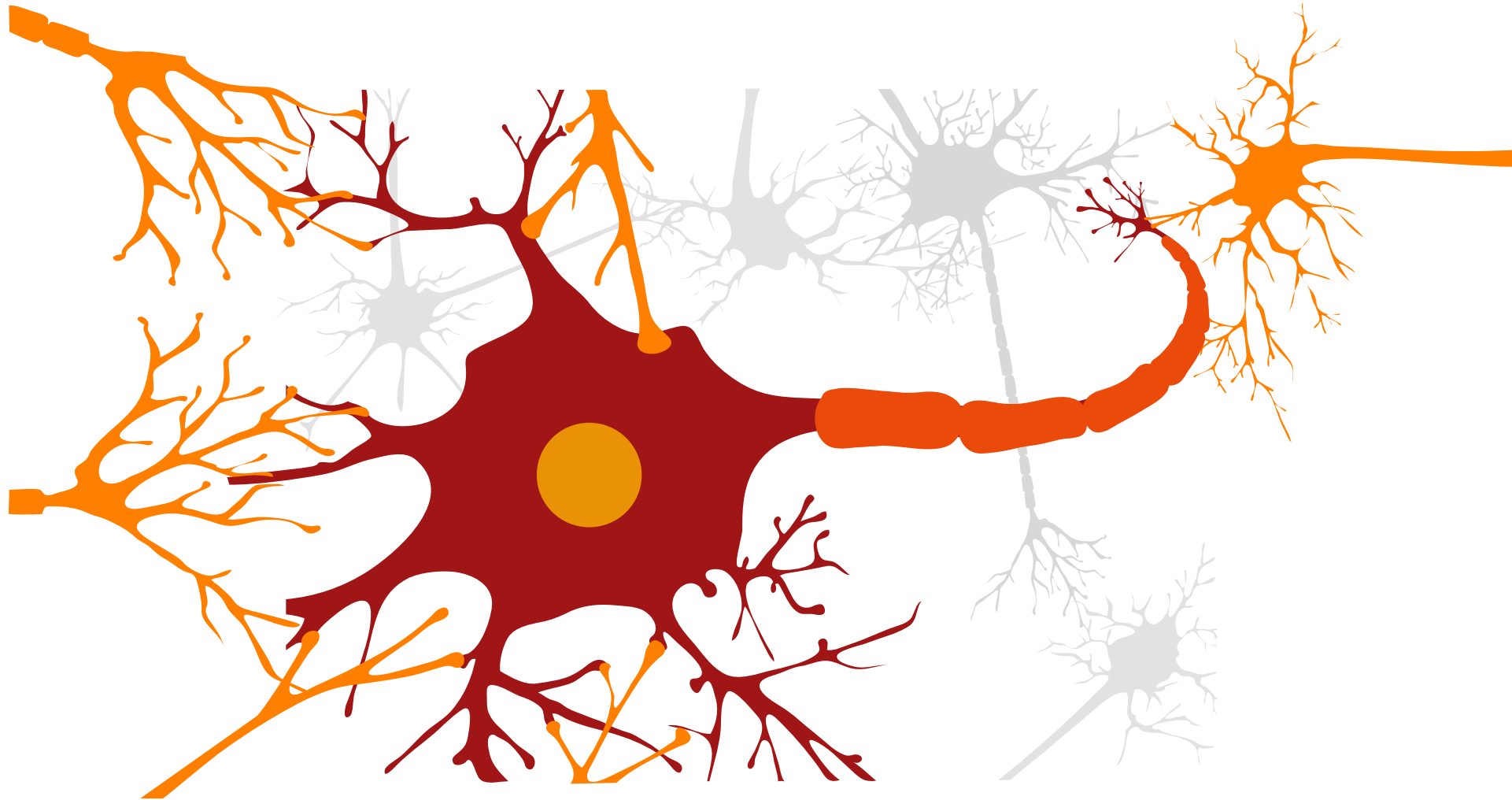


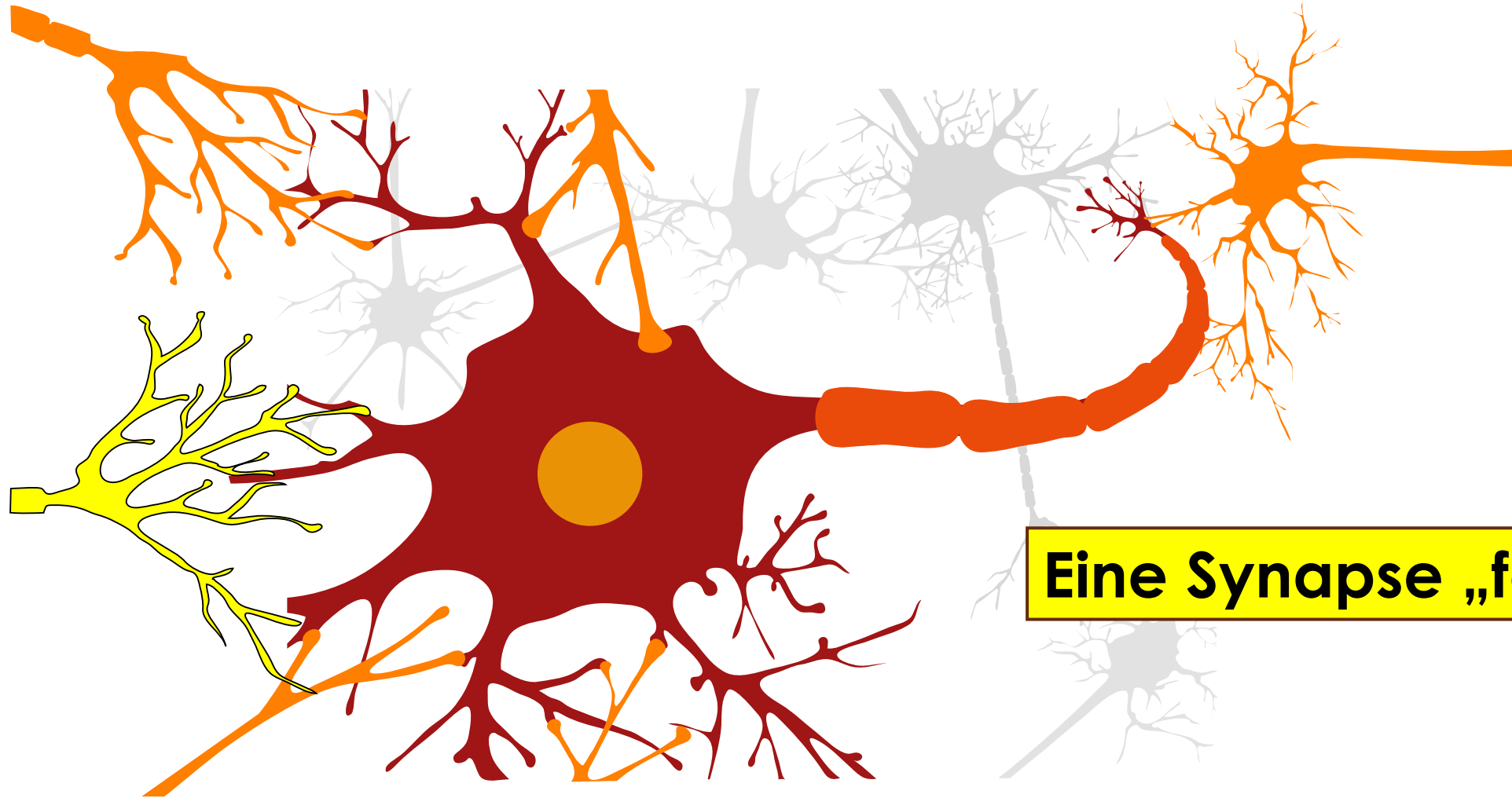
**Dendriten**

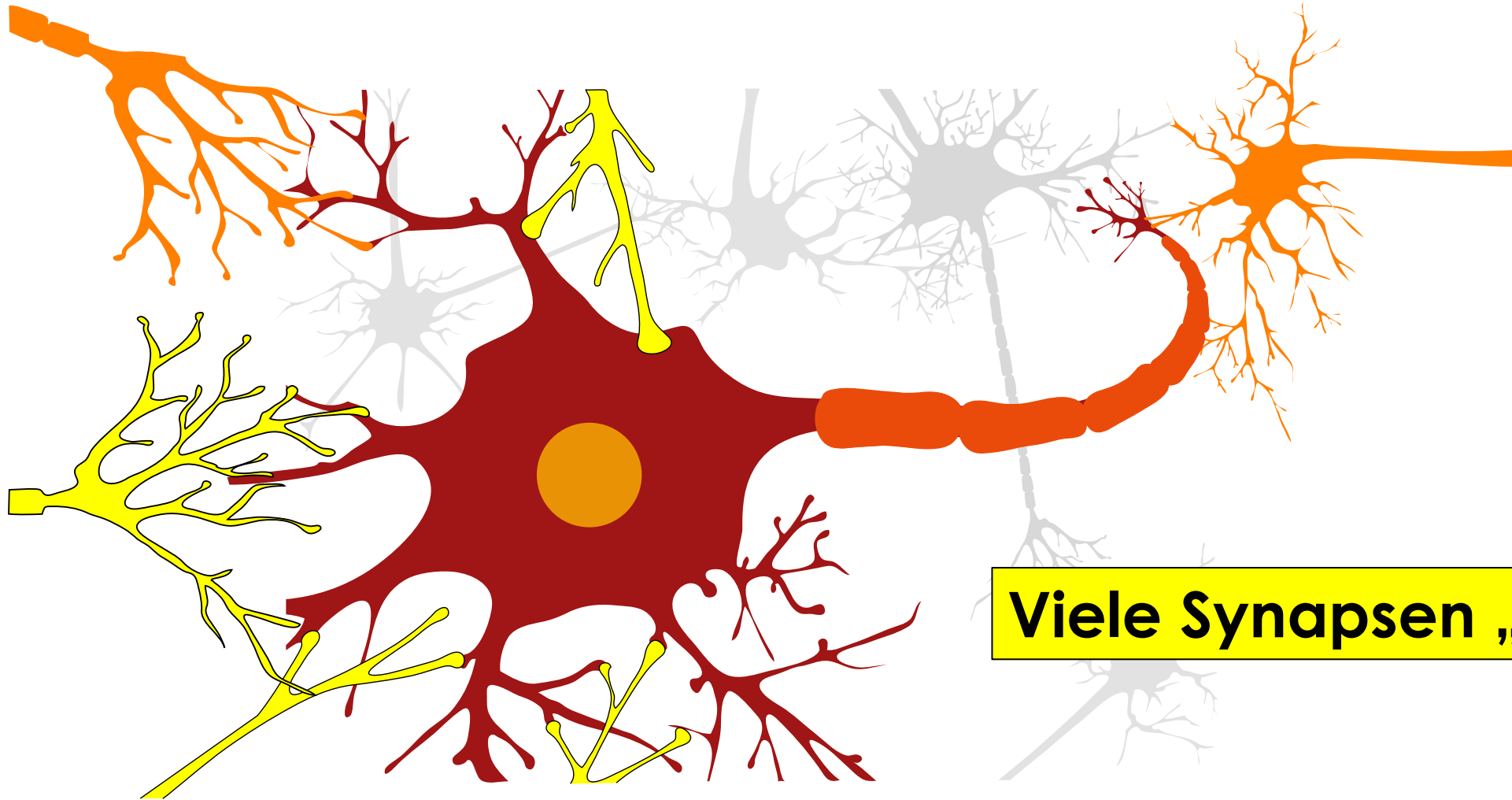




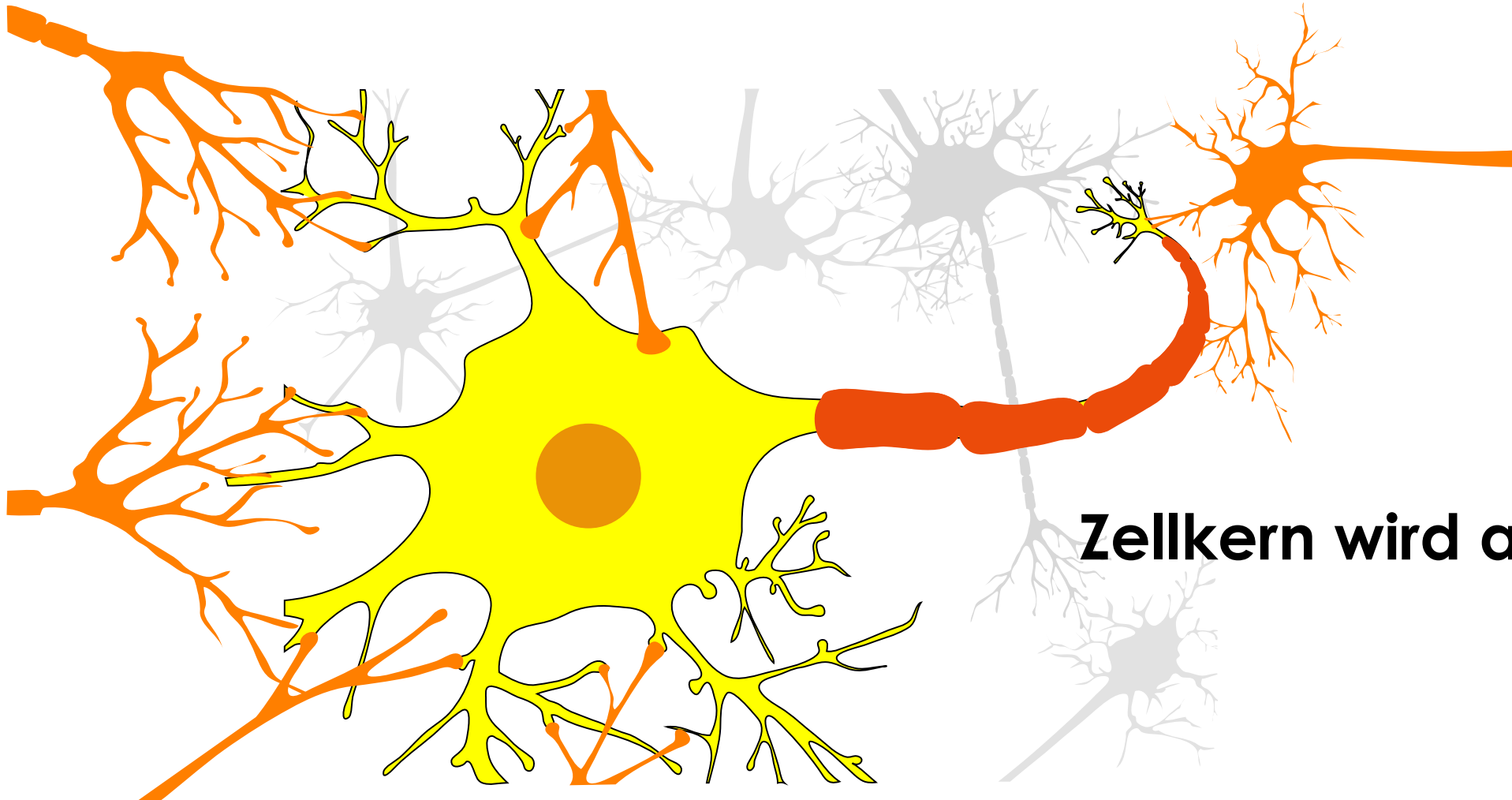
**Synapse**



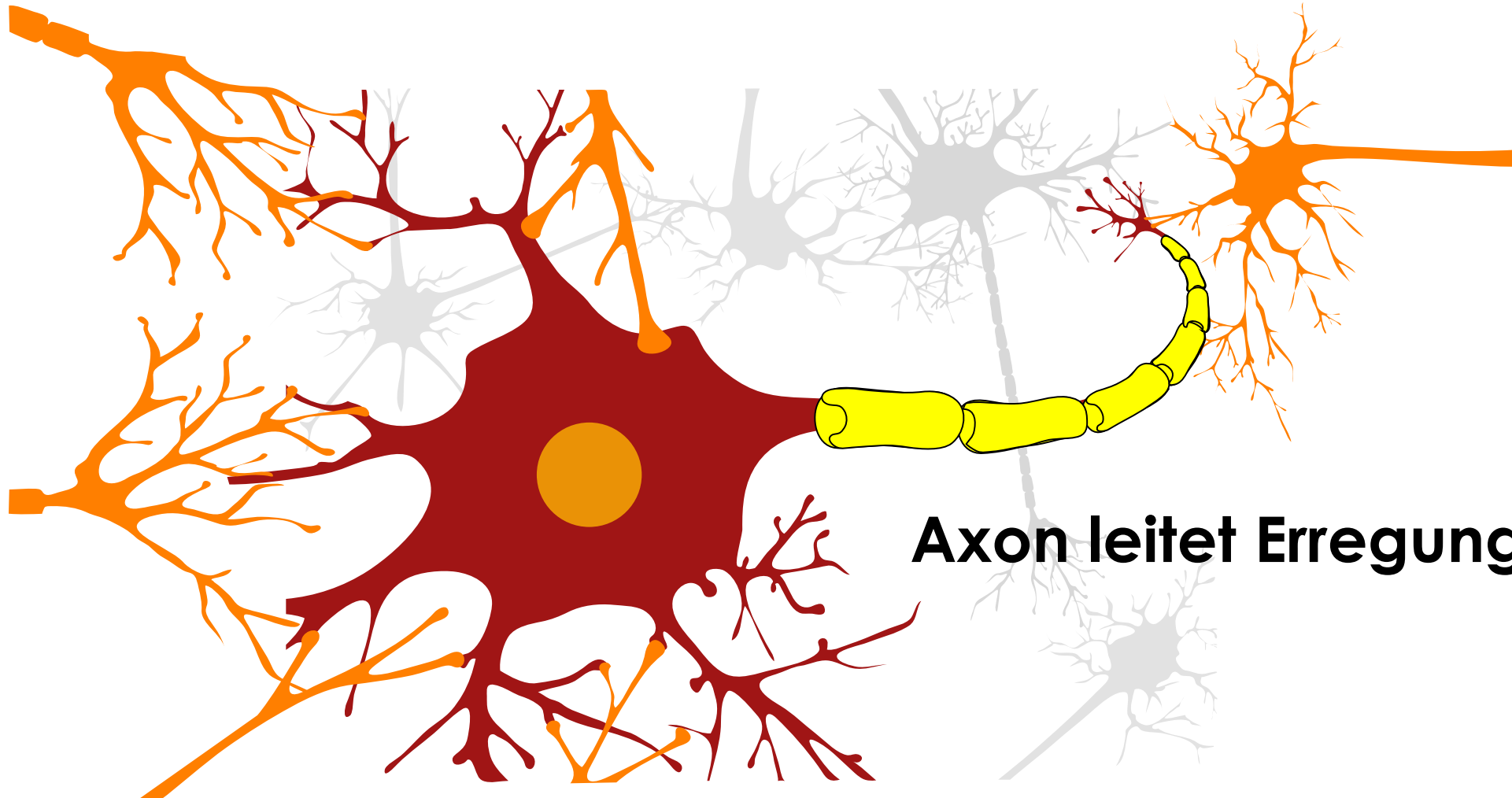




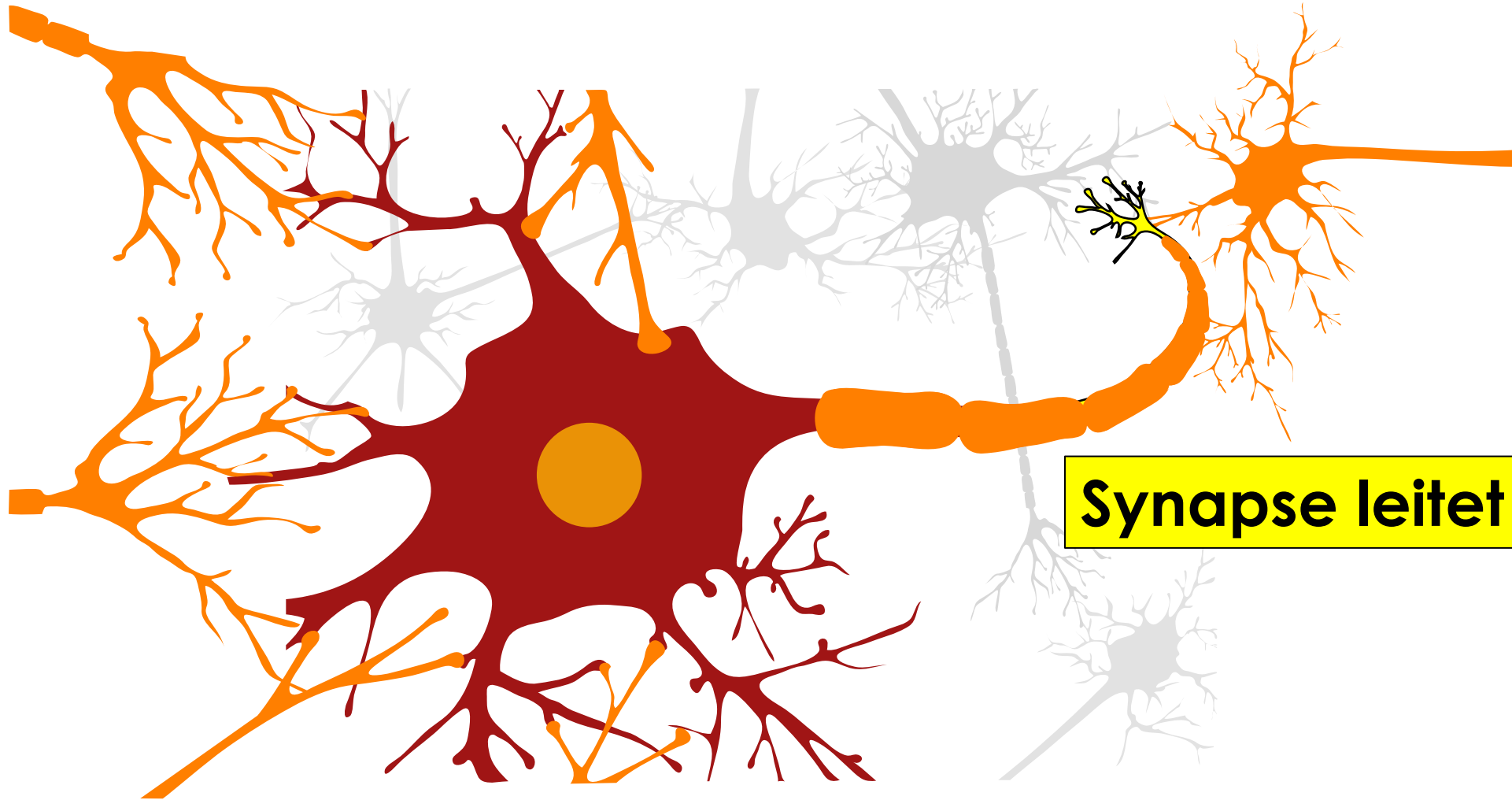
**Viele Synapsen „feuern“**



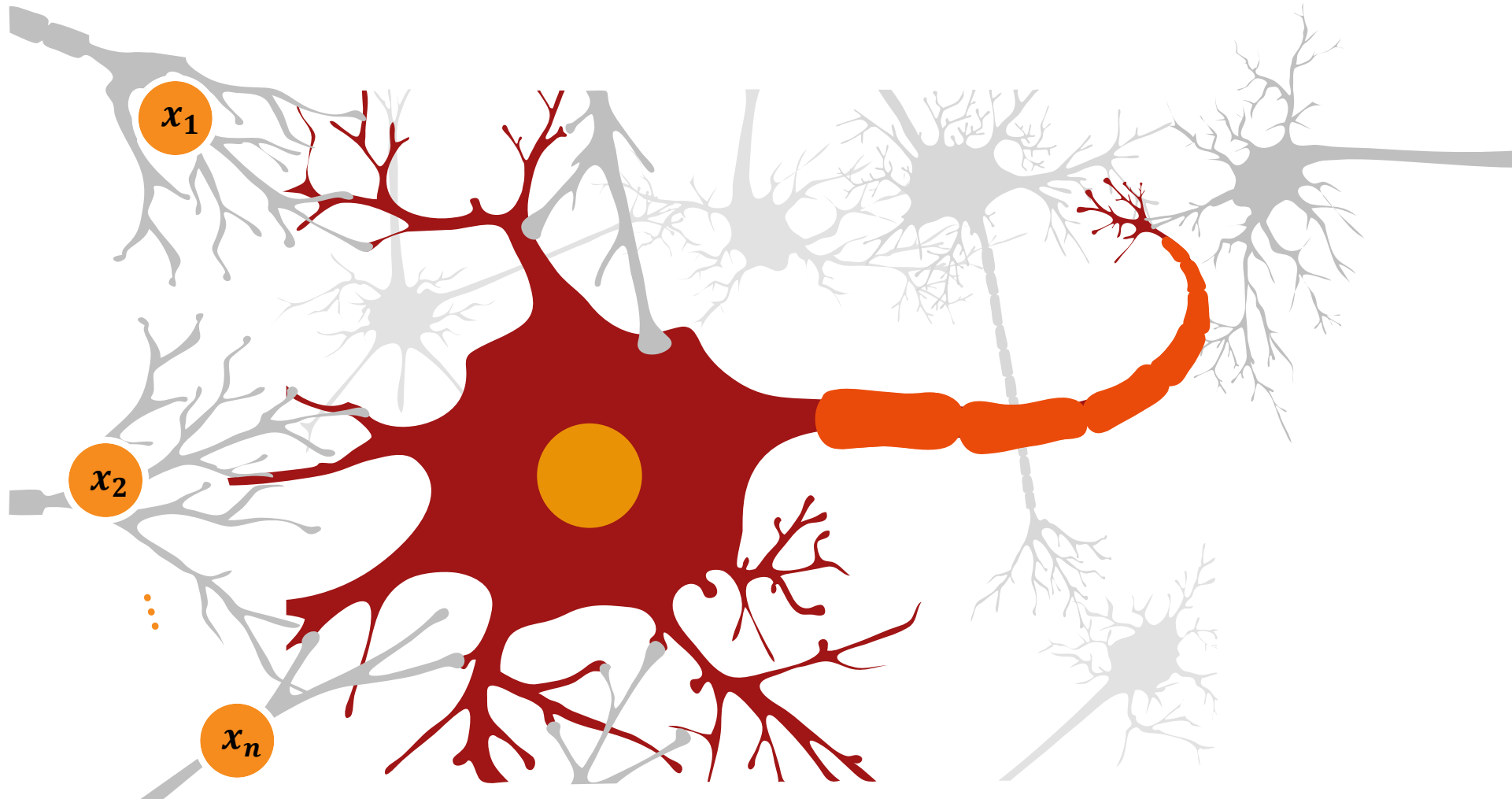
**Zellkern wird angeregt**



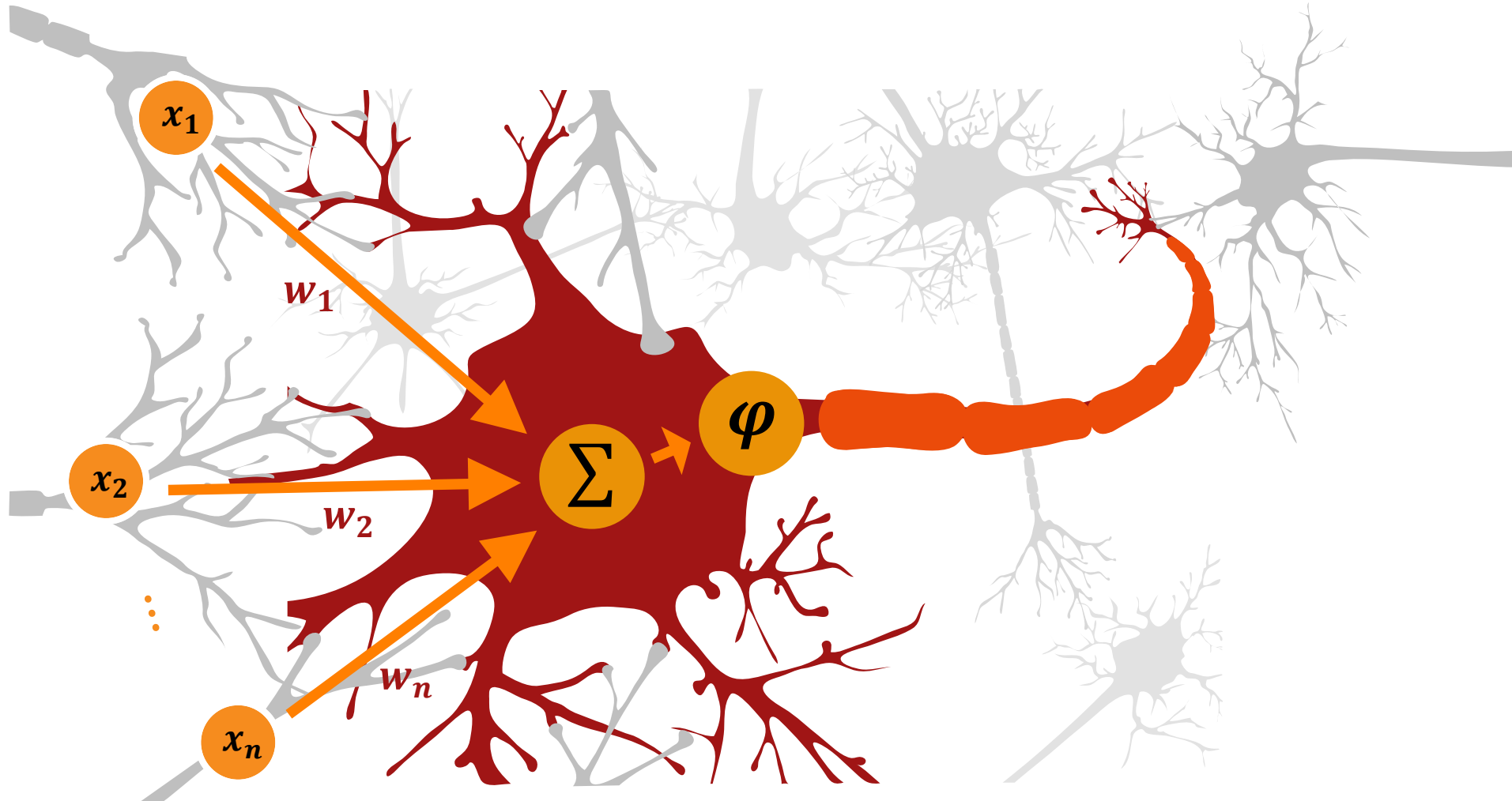
**Axon leitet Erregung weiter**

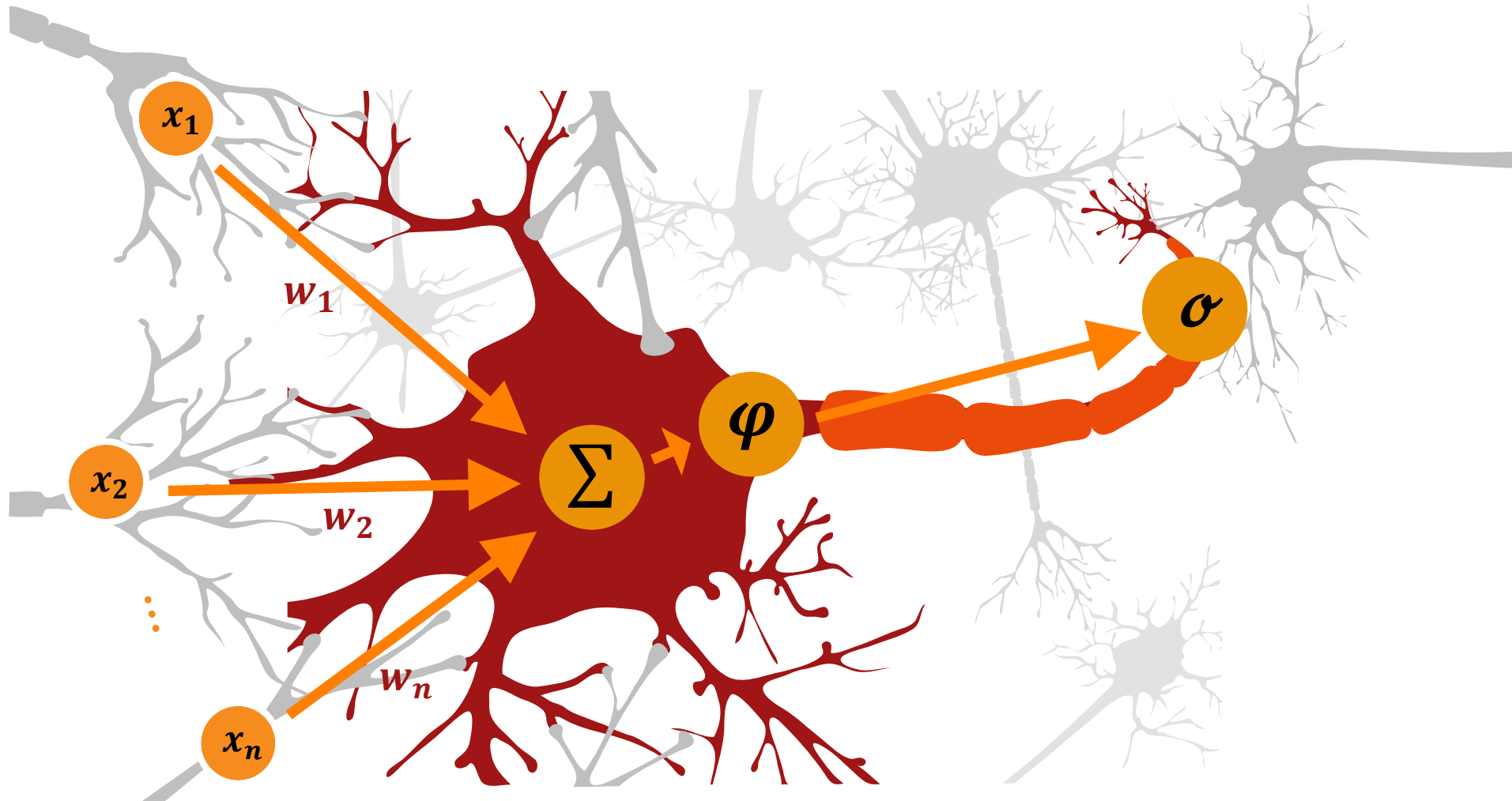


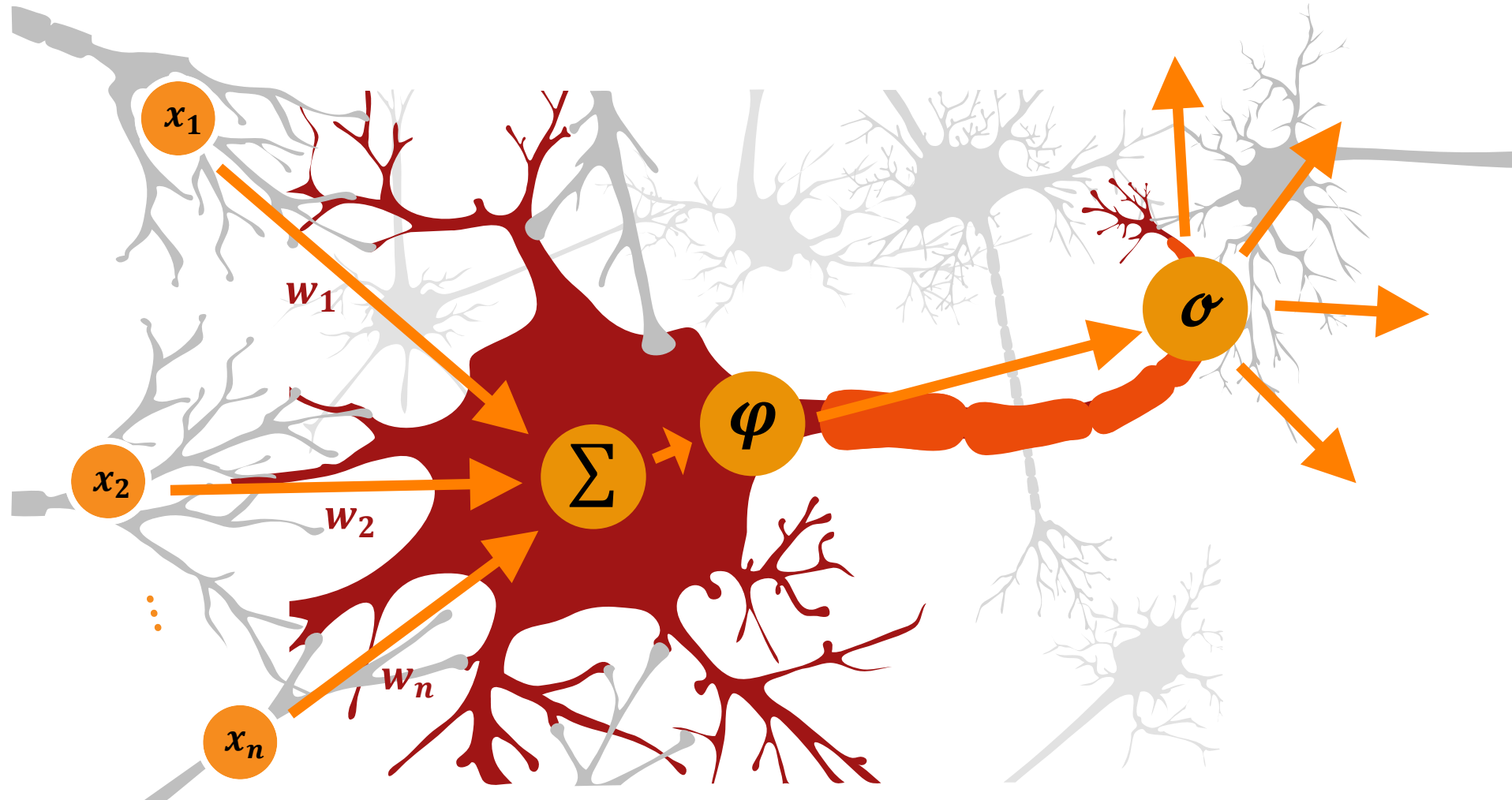
**Synapse leitet weiter**



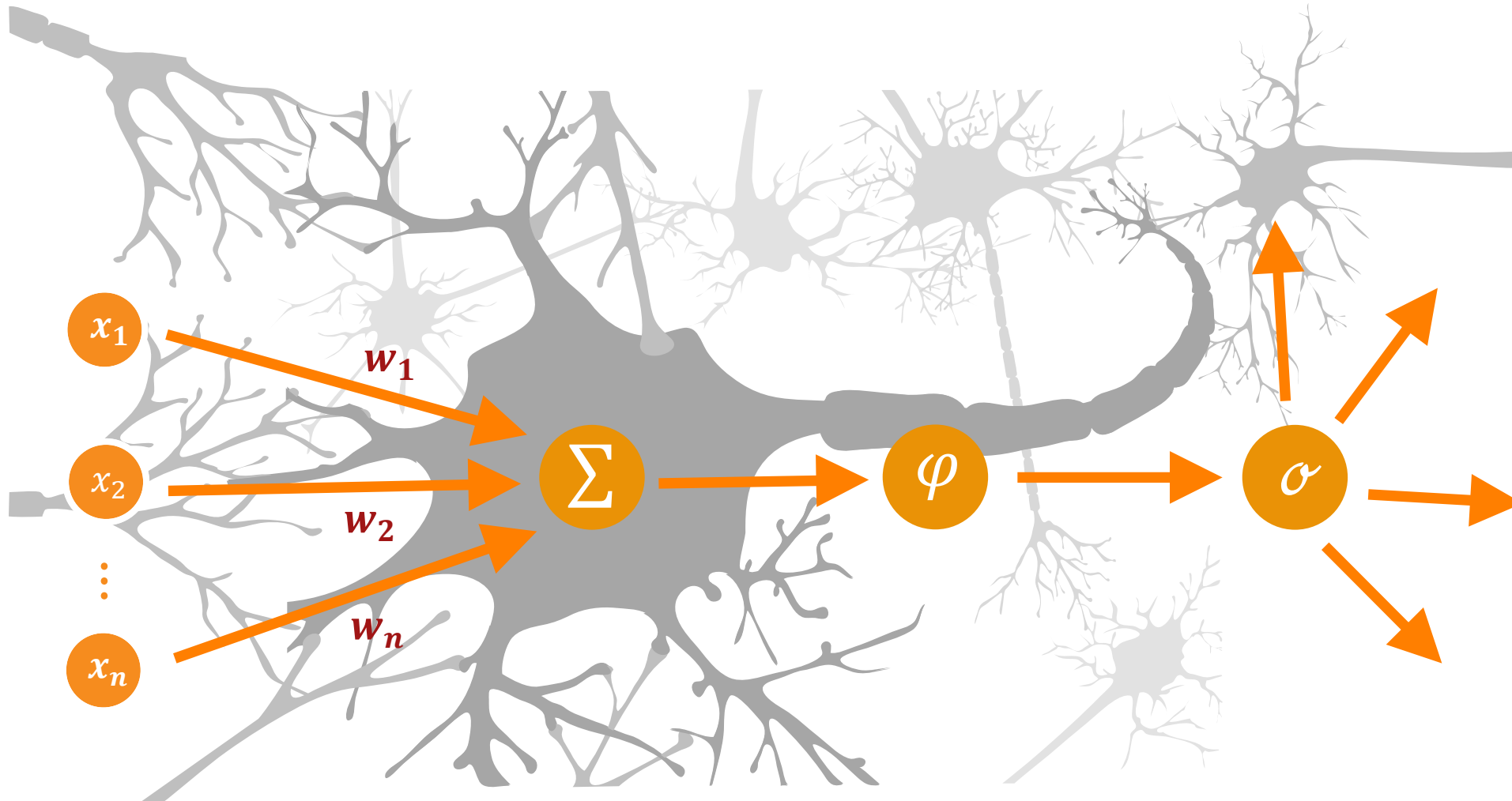


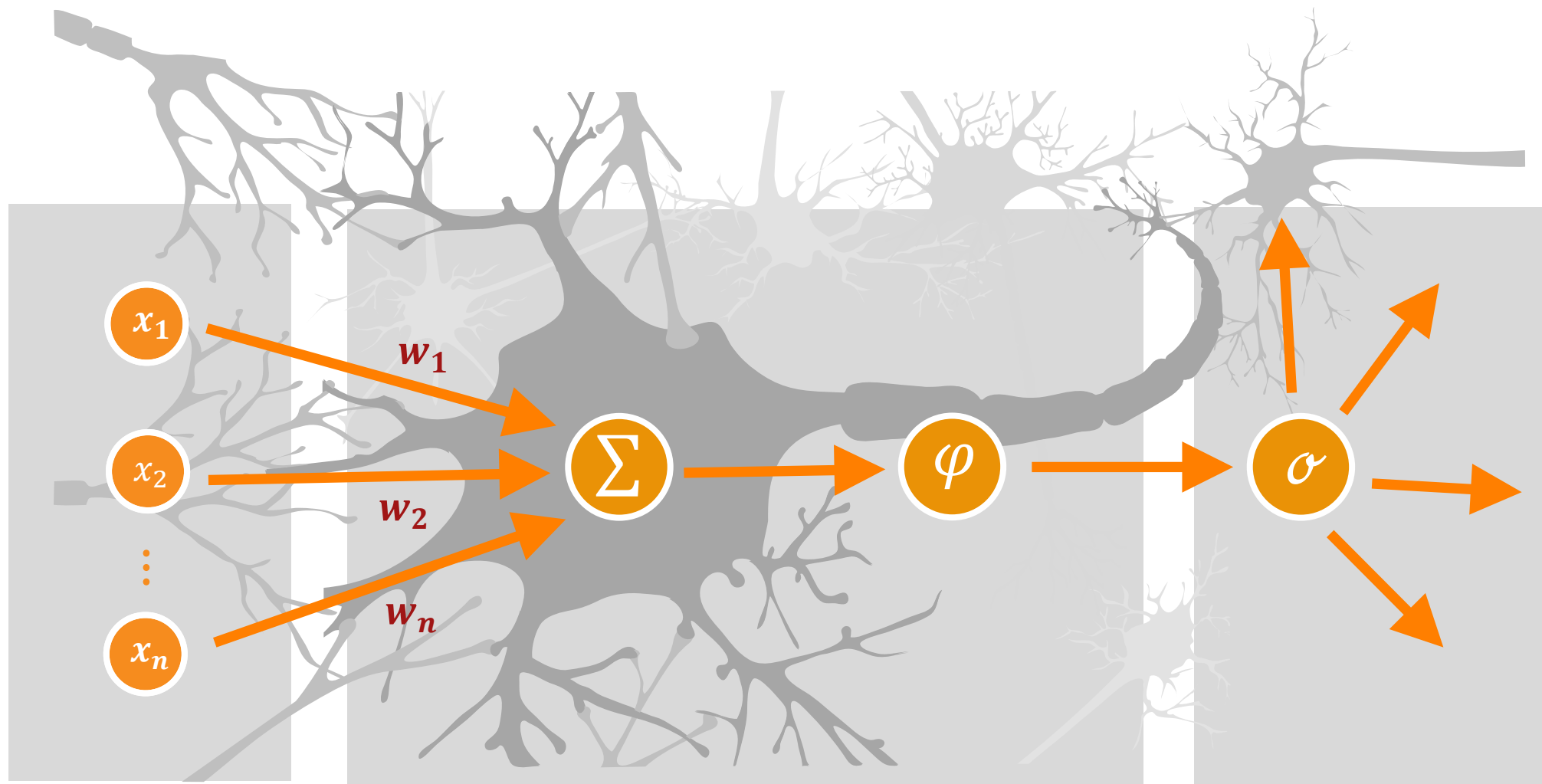


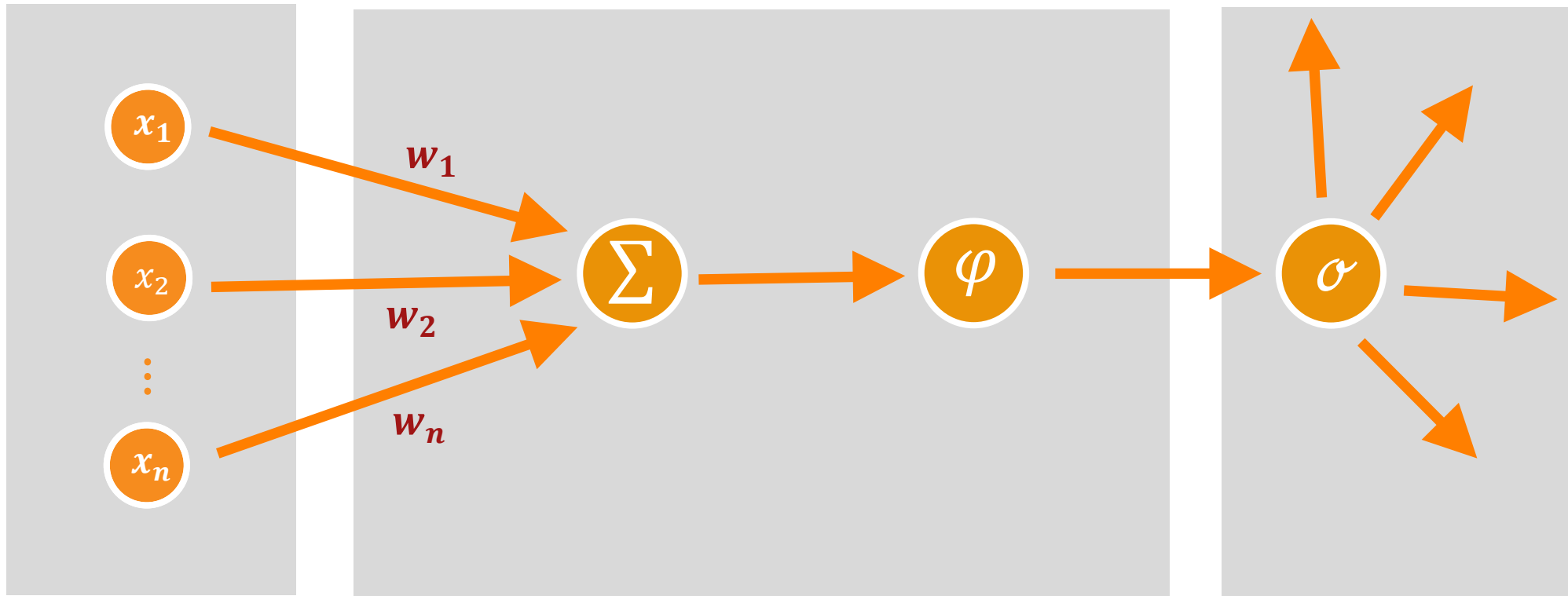


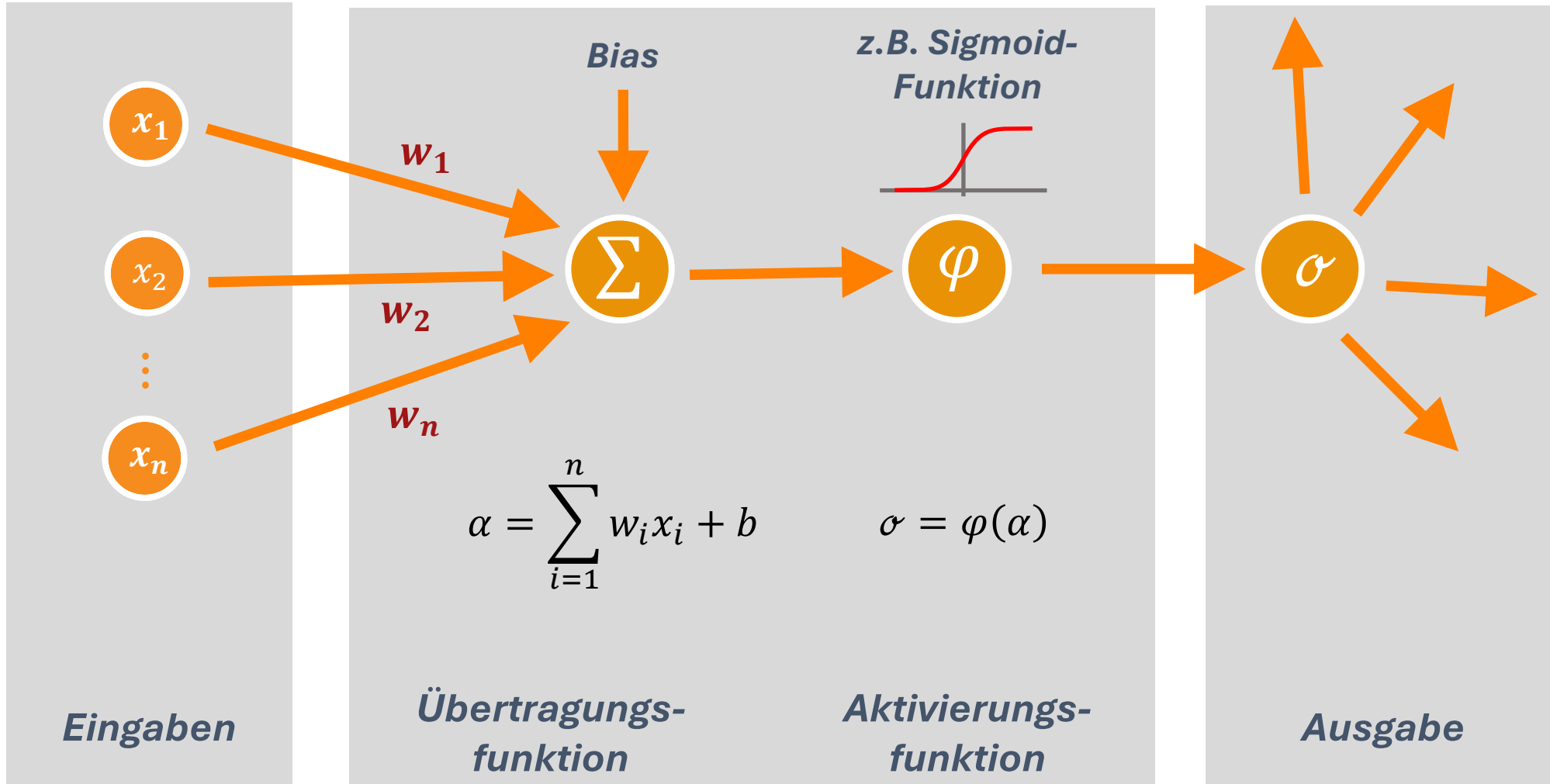


# Vom Künstlichen Neuron zum Künstlichen Neuronalen Netz

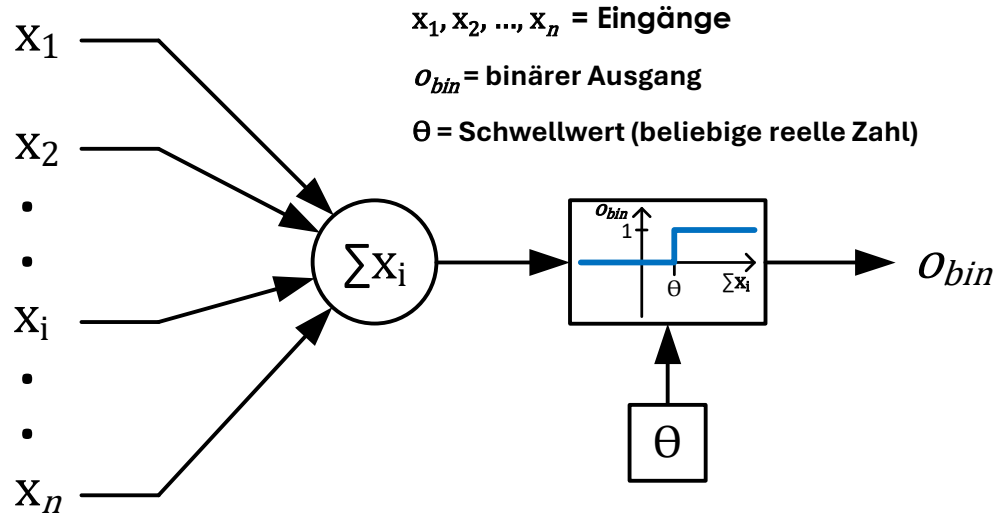




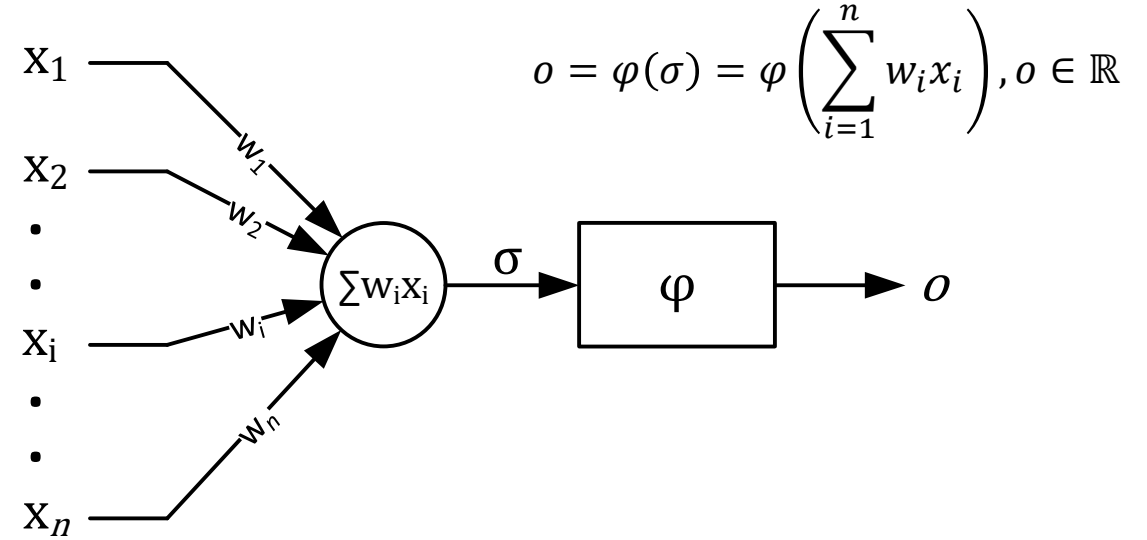




## McCulloch-Pitts-Neuron (1943)



## Perzeptron (1958)

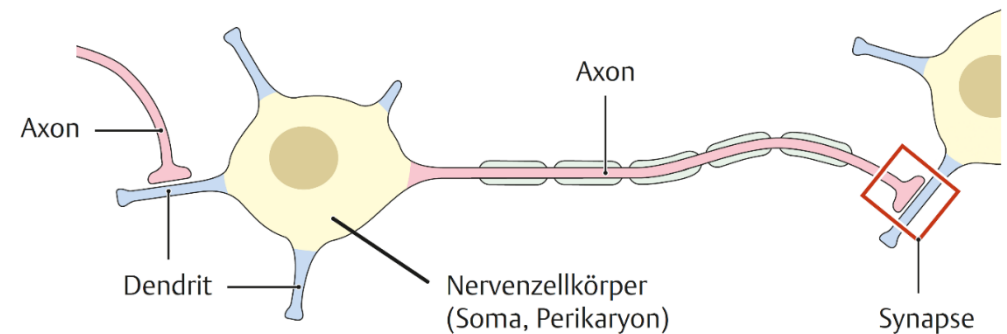


### McCulloch-Pitts-Neuron

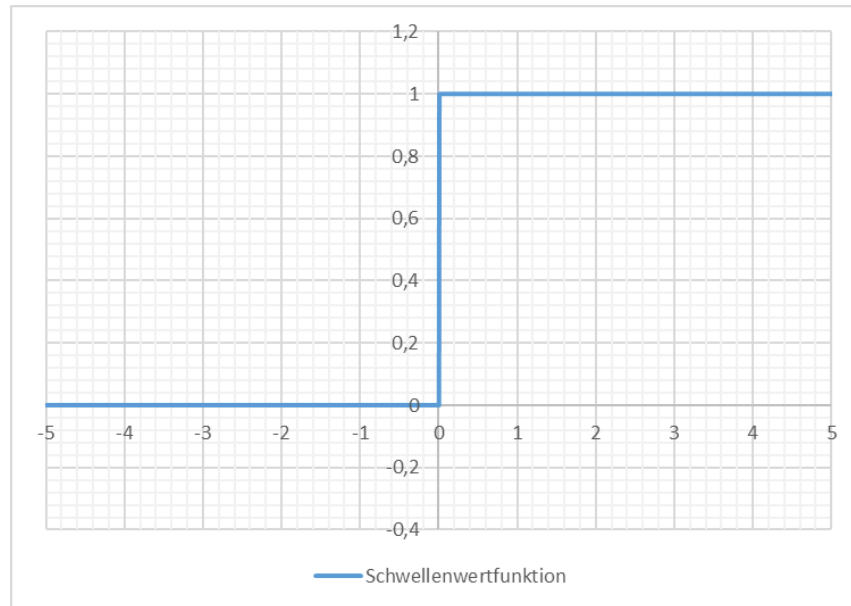
- **Übertragungsfunktion:** Summe
- **Aktivierungsfunktion:** Schwellwertfunktion
- **Ausgang:** binär

### Perzeptron

- **Übertragungsfunktion:** frei wählbar, meist gewichtete Summe
- **Aktivierungsfunktion:** frei wählbar
- **Ausgang:** variabel
- **Stärke der Synapse:** wird über Gewichtung berücksichtigt

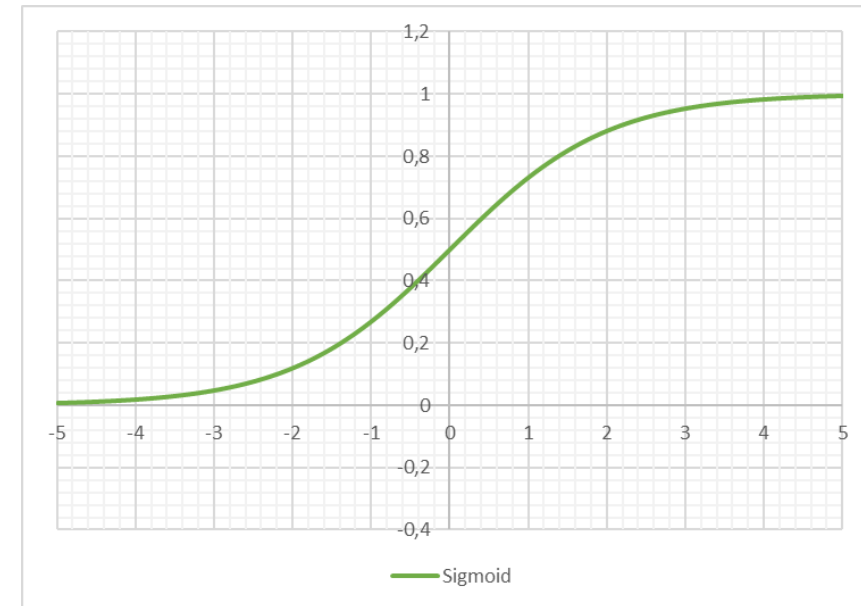


$$\varphi_{\text{Schwellwert}}(\sigma) = \begin{cases} 1, & \text{falls } \sigma \geq \text{Schwellwert} \\ 0, & \text{sonst} \end{cases}$$



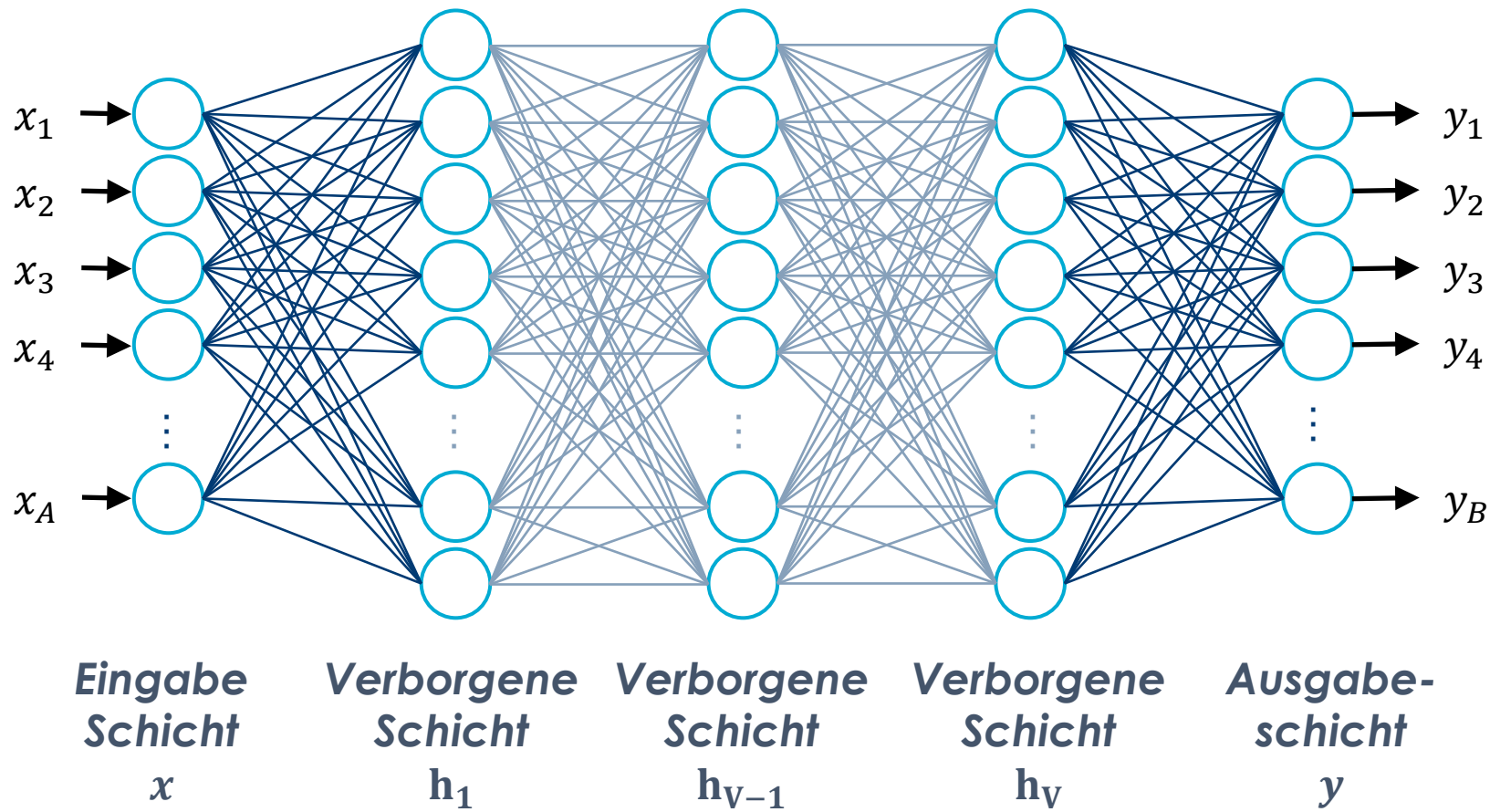
- Sonderfall: Schwellwert = 0  
→ Heaviside-Funktion
- dem menschlichen Körper nachempfunden
- 👎 nicht differenzierbar in allen Punkten

$$\varphi_{\text{Sigmoid}}(\sigma) = \frac{1}{1 + e^{-a\sigma}} \in \{0, \dots, 1\}$$



- variable Steigung  $a$
- 👍 differenzierbar → für **Backpropagation**\* geeignet
- 👎 Ableitung geht an den Randbereichen gegen 0  
→ Gradient der Aktivierungsfunktion wird sehr klein  
(**Vanishing Gradient Problem**)
- 👎 nicht null-zentriert

\* wird für Anpassung der Gewichtungen im Trainingsprozess benötigt



# Der Trainingsprozess

**Trainingsprozess:** mathematisch beschreibbarer Prozess, Anpassen der Gewichtungen

## Trainingsdaten

- dienen zur Anpassung der Gewichtung  $w_{1...n}$

## Validierungsdaten

- dienen zur Überwachung des Trainingsprozesses

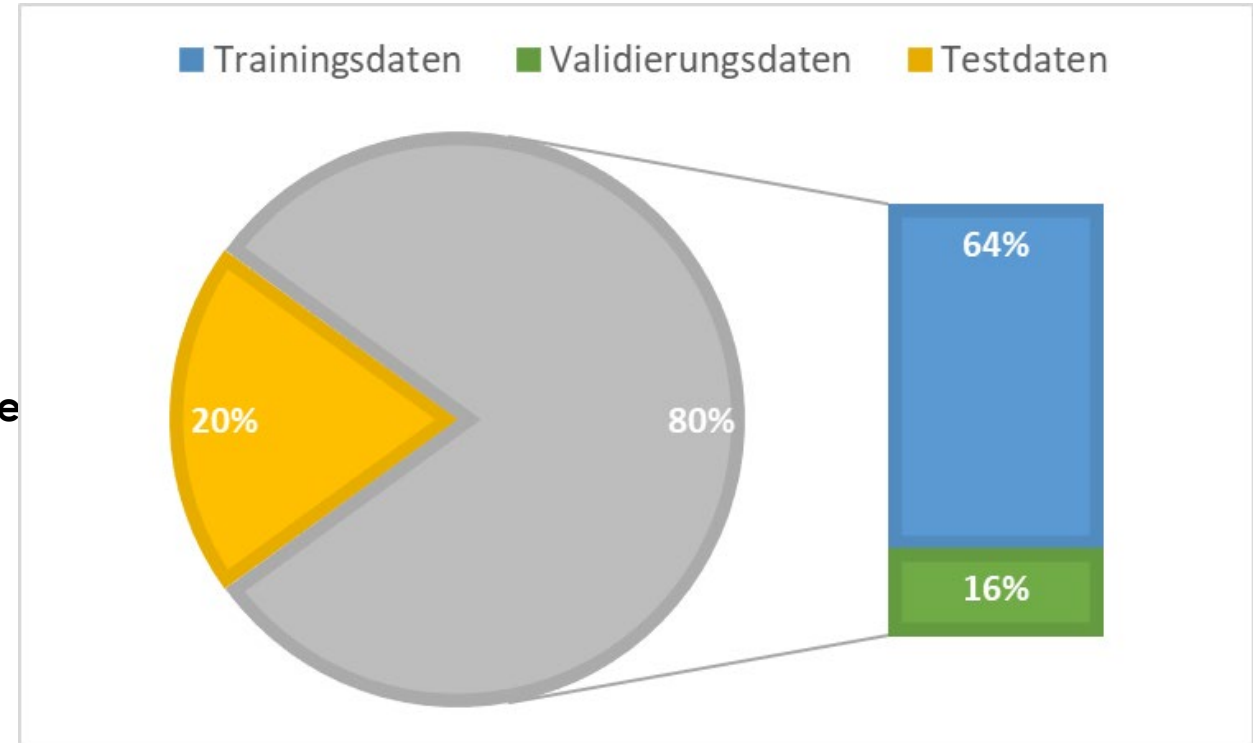
## Testdaten (Evaluierungsdaten)

- werden erst nach dem Training angewendet, um die Qualität des Modells zu berechnen

## ⇒ Performancedaten

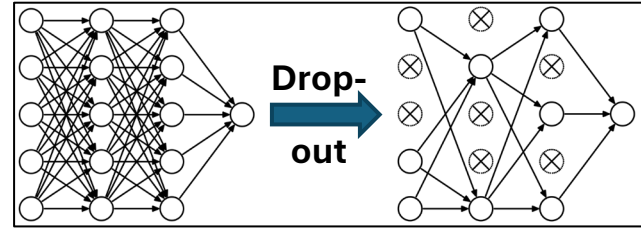
### Anforderungen an Trainingsdaten:

- möglichst viele realitätsnahe Netzzustände
- z.B. Abbildung von Minimallast-, Maximallast- und Überlastzuständen, etc.



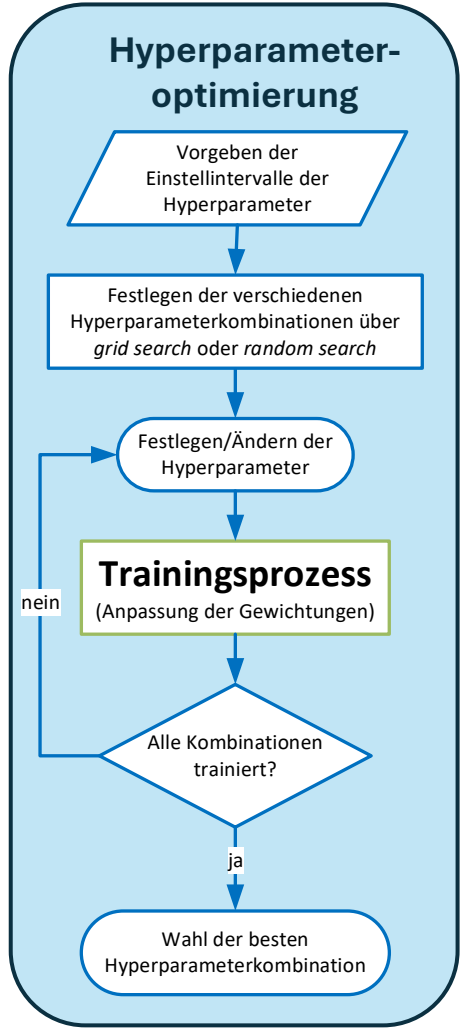
Hyperparameter für die KNN-Struktur	Hyperparameter für den Trainingsprozess
Anzahl an Neuronen	Lernrate $\eta$
Anzahl an verdeckten Schichten	Anzahl an Epochen
Dropout-Rate	Batch size
Anfangsgewichtung	Genauigkeitsmaß
Aktivierungsfunktion $\varphi$	Optimierungsverfahren
Verlustfunktion (z.B. Mean Square Error $E_{MSE}$ )	

$$E_{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

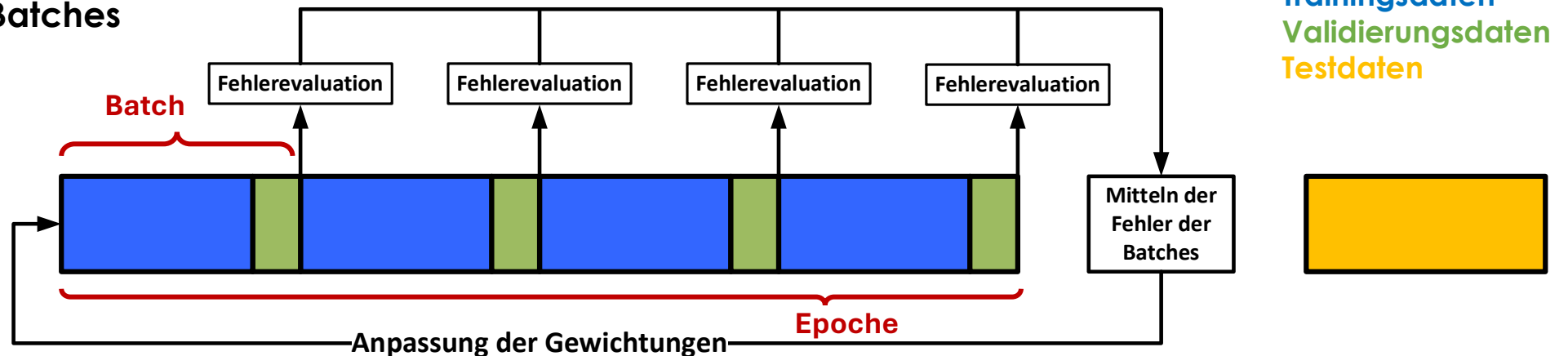


Hyperparameter werden üblicherweise vor jedem einzelnen Trainingsprozess festgelegt!

- ➡ Trainingsprozess wird mit verschiedenen Hyperparametersätzen durchlaufen
- ➡ Wahl der besten Hyperparameterereinstellungen (z.B. Festlegen auf eine Aktivierungsfunktion  $\varphi$ )

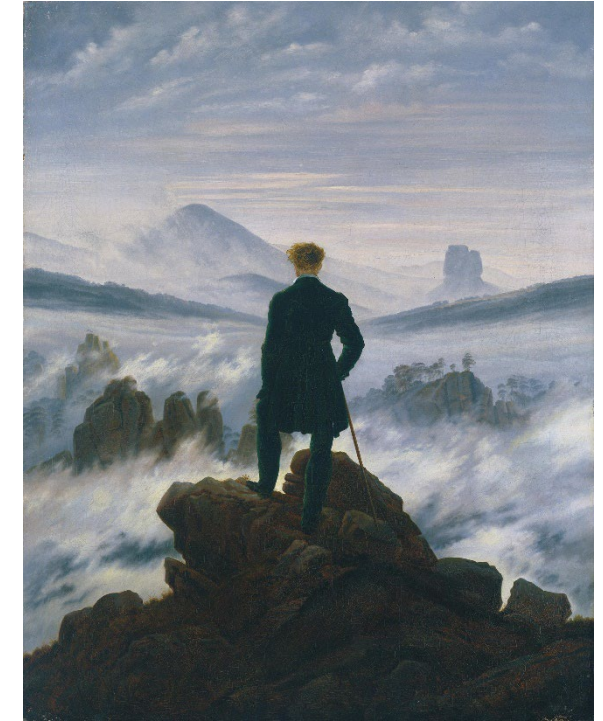
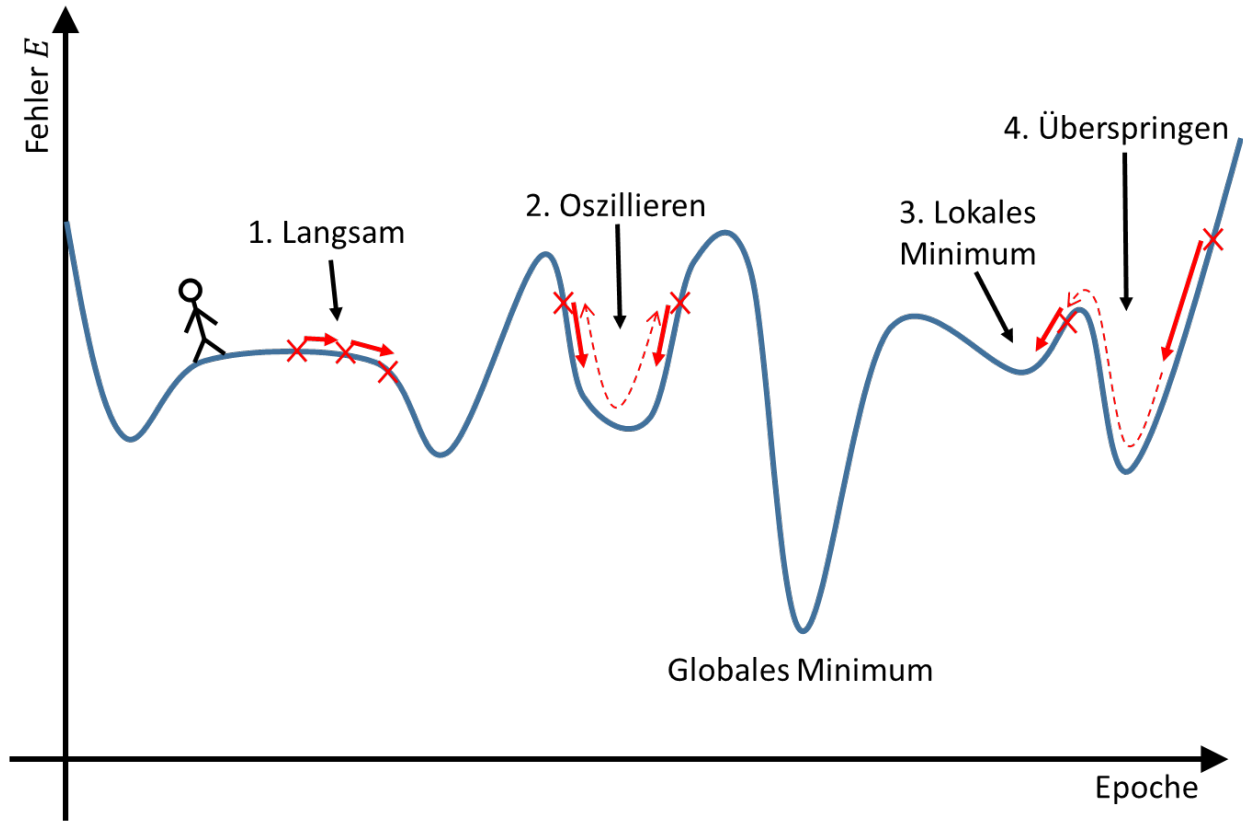


- Trainings- und Validierungsdaten werden in „kleinere Päckchen“ unterteilt → **Batches**
- Aufteilung der Trainingsdaten in die Batches erfolgt zufällig
- Anzahl der Trainingsdaten pro Batch → **Batch size**
- Gängige Batch sizes: 32, 64, 128, ...
- Durchlauf eines kompletten Datensatzes (alle Batches) durch den Trainingsprozess → **Epoche**
- In jeder Epoche werden die Trainingsdaten in neue Batches unterteilt
- Nach jedem Batch: Berechnen des Fehlers zwischen erwarteter und aktuell berechneter Ausgangsgröße → **Fehlerevaluation**
- **ABER:** Anpassung der Gewichtung erfolgt erst am Ende der Epoche anhand des gemittelten Fehlers aus allen Batches



➔ zur Minimierung der Verlustfunktion (Fehlerfunktion *Fehler E*)

**Hyperparameter:** Auswirkungen der Wahl der Lernrate  $\eta$  ( $\triangleq$  Schrittweite):



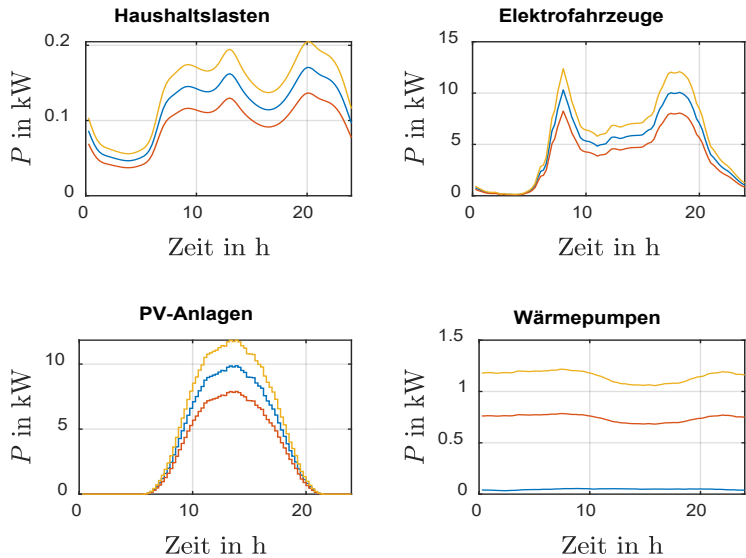
Der Wanderer über dem Nebelmeer (Caspar David Friedrich)

Länge des Pfeils ( $\rightarrow$ )

$$w_{ij,new} = w_{ij,alt} + \Delta w_{ij} = w_{ij,alt} - \eta \cdot \frac{\partial E}{\partial w_{ij}}$$

# Die Trainingsdaten generieren

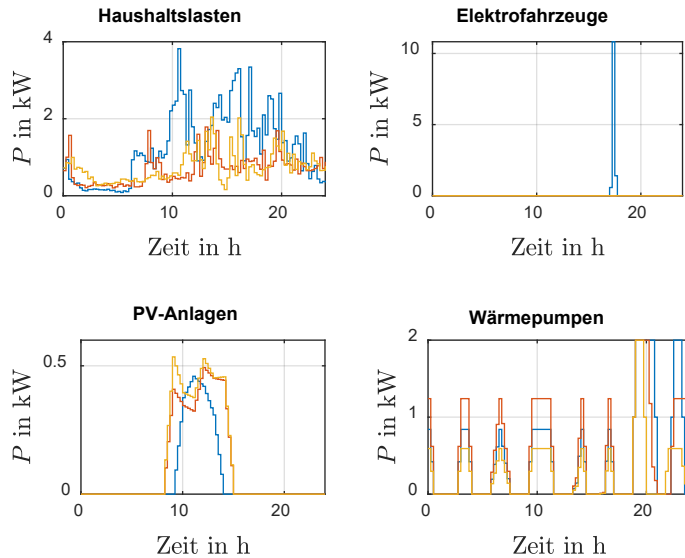
## Standardisierte Leistungsprofile



- Normierung auf die Jahresenergiemenge

➔ Leistungsspitzen werden vernachlässigt

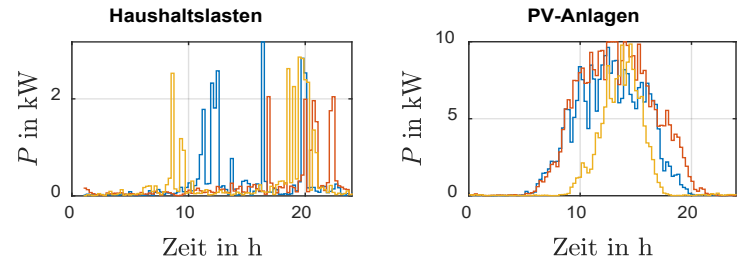
## Messwerte



- Realistische und stochastische Abbildung

➔ Berücksichtigung individueller und kollektiver Schwankungen

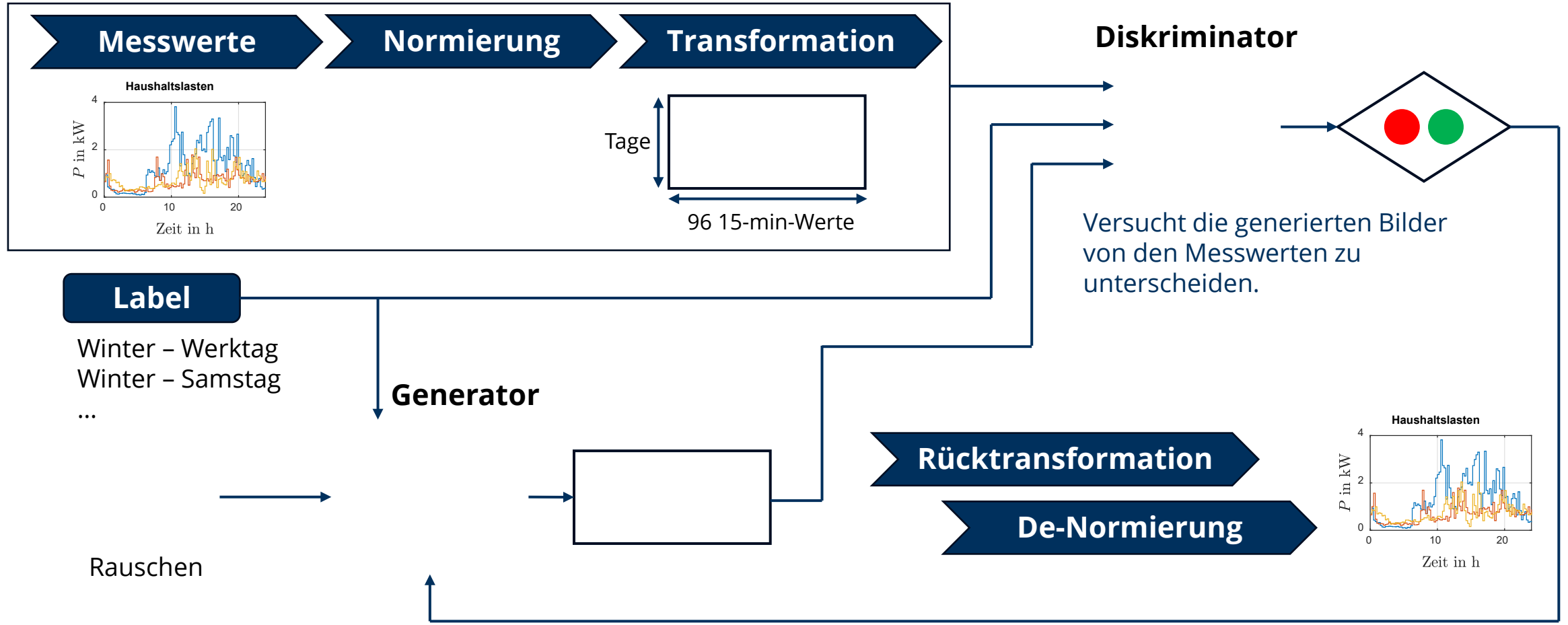
## Generative Adversarial Networks



➔ **Generative Adversarial Networks (GAN) können jede beliebige Verteilung nachahmen.**

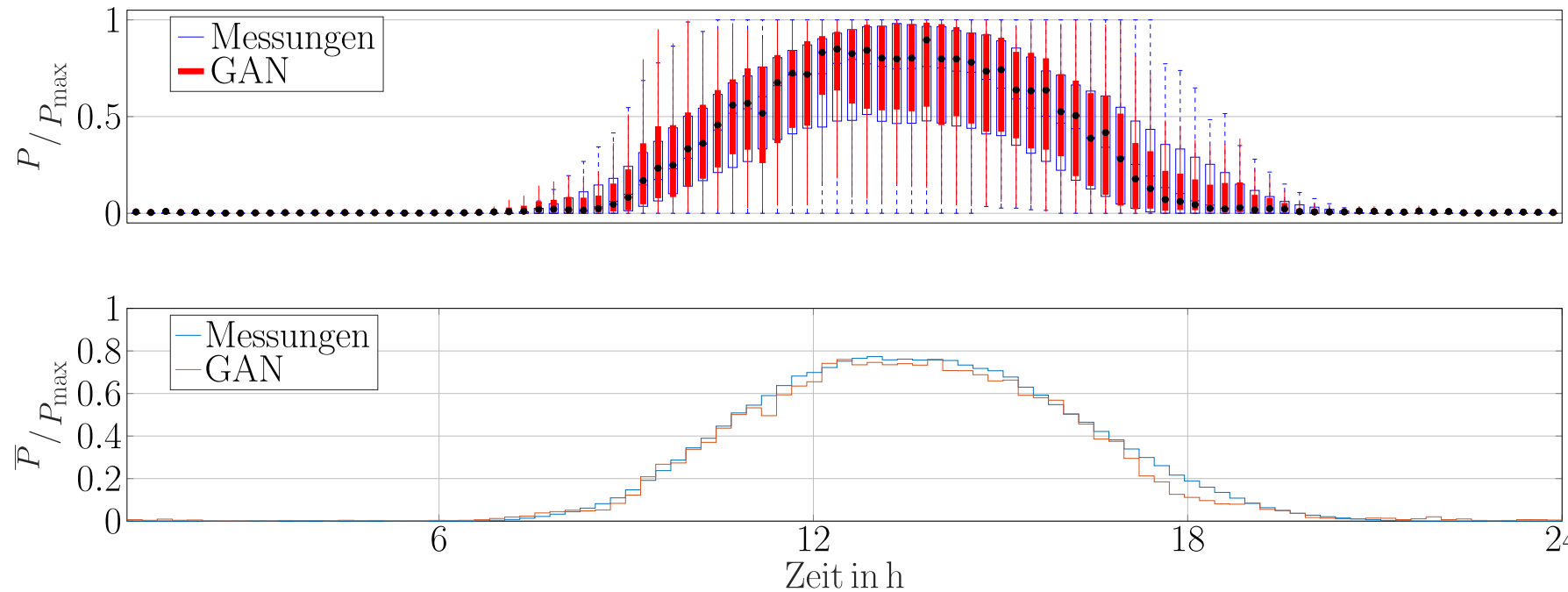
- Realistische und hochauflösende Mustererzeugung
- Erfassung komplexer Zusammenhänge
- Anpassung an neue Trends und Technologien

## GAN-basierte Generierung synthetischer Profile



## ➔ Vergleich – GAN-basierte Profile und Messungen

- PV-Anlagen
- Datengrundlage: eine PV-Anlage, 365 Tage
- Anforderungen an die maximale und mittlere Profileistung sind stets erfüllt



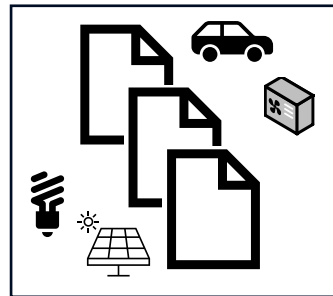
Einspeisespitzen  
sind korrekt  
abgebildet

Mittelwerte sind  
korrekt  
abgebildet

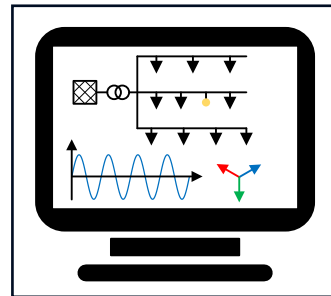
# Stromverteilnetze überwachen - Die KI-basierte Netzsimulation

## ➔ Falldatengenerierung für das Training der KI-Systeme

- Durchführung der Netzberechnung mit einem klassischen **Netzberechnungsprogramm** und einem **Stromiterationsverfahren**
- Durchführung der Netzzustandsoptimierung mit einem **quadratischen Programm**

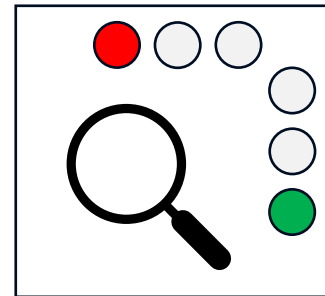
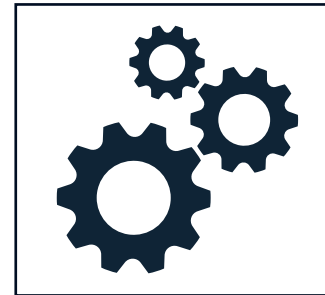


Ergebnisse der Szenariengenerierung über mehrere Jahre



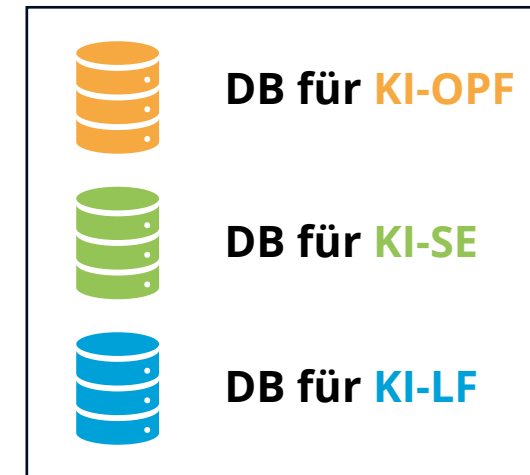
Netzberechnung

### Netzzustandsoptimierung



Netzzustandsdiagnose

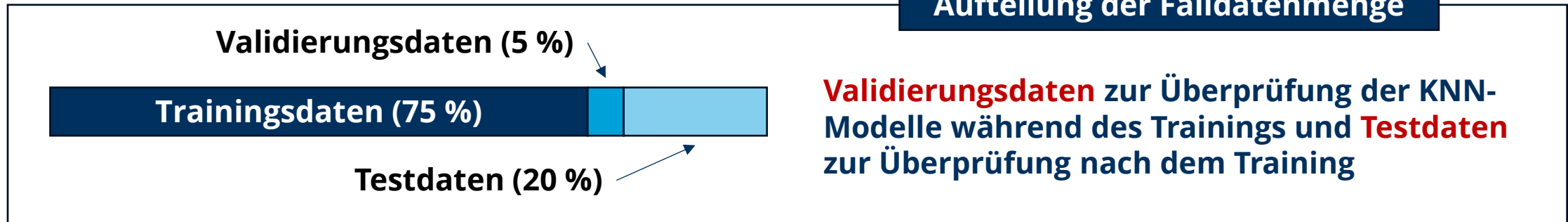
Tarife und Flexibilitäten



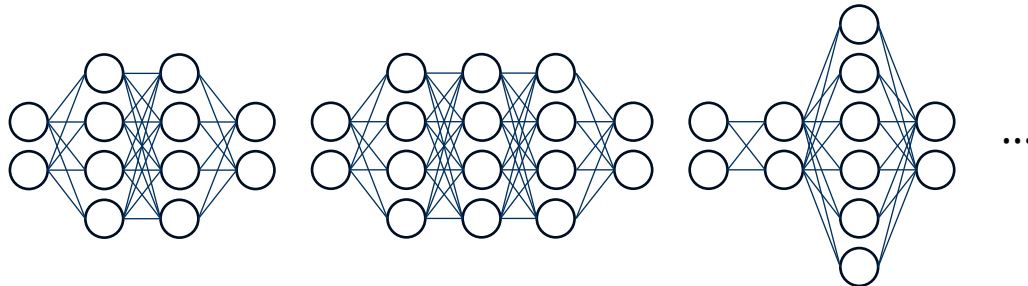
Quellen: [8]

## ➔ Training des KNN

### Aufteilung der Falldatenmenge



### KNN-Architekturen entwickeln

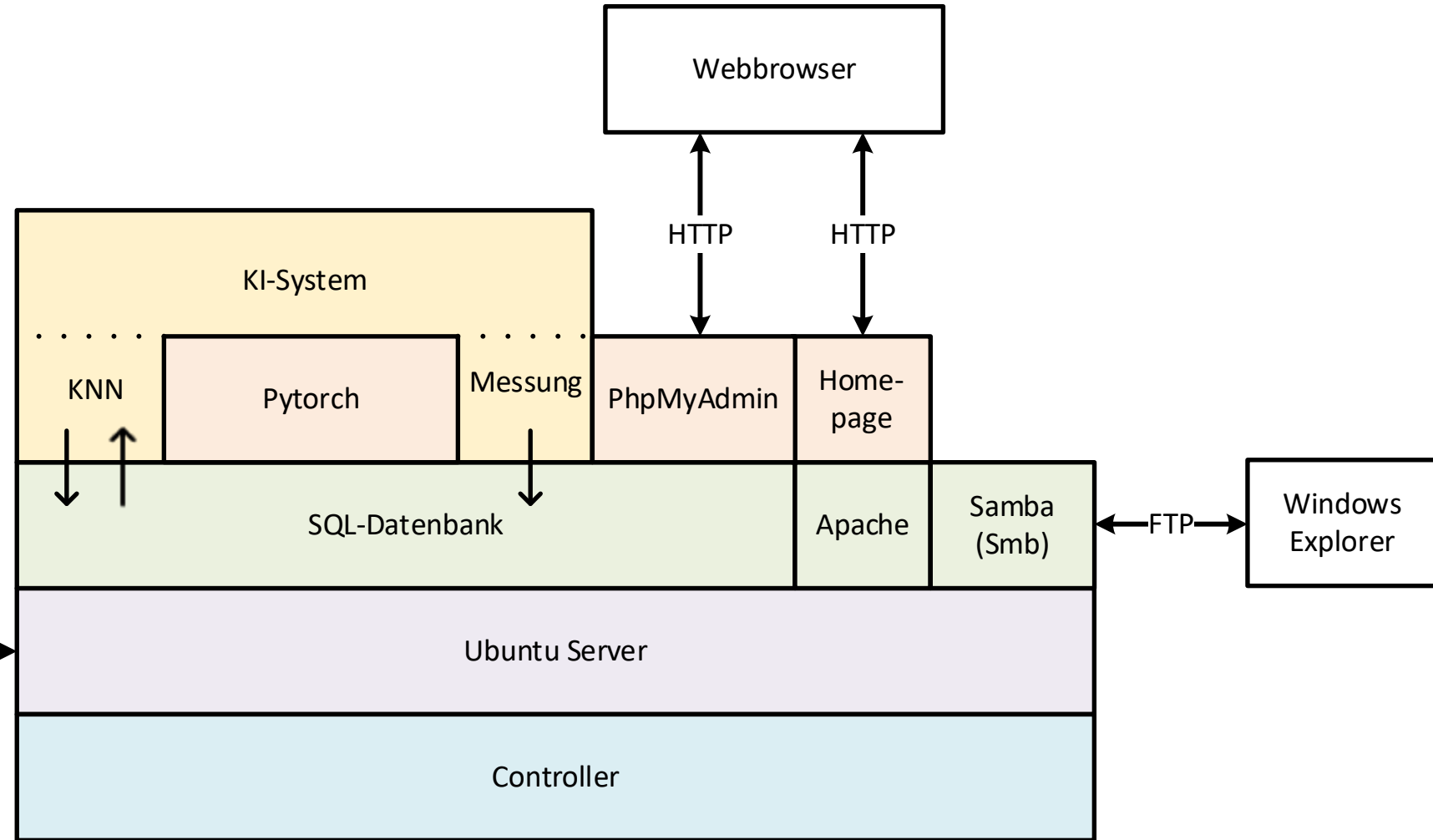


- Erstellung zufälliger KNN-Architekturen
- Training (20 Epochen) der KNN → Auswahl der besten Performance

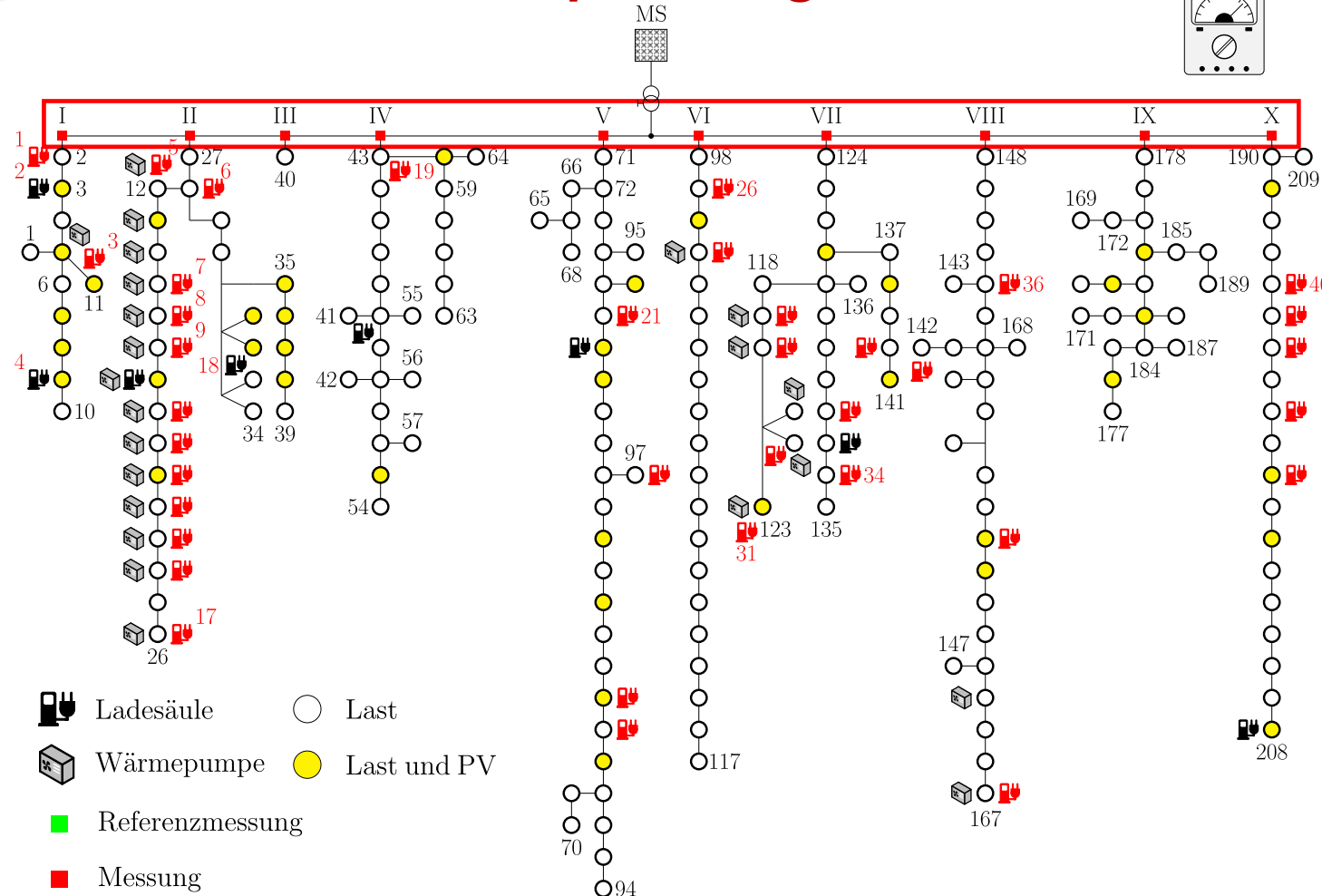
### Training der KNN

- Training der ausgewählten Modelle mit 500 Epochen
- Optimierungsalgorithmus: **ADAM** [7]
  - Individuelle Anpassung der Lernrate für jedes Gewicht → schnelle Konvergenz
- Validierung des Modells mit den Testdaten

**Softwarekonzept für das KI-System**



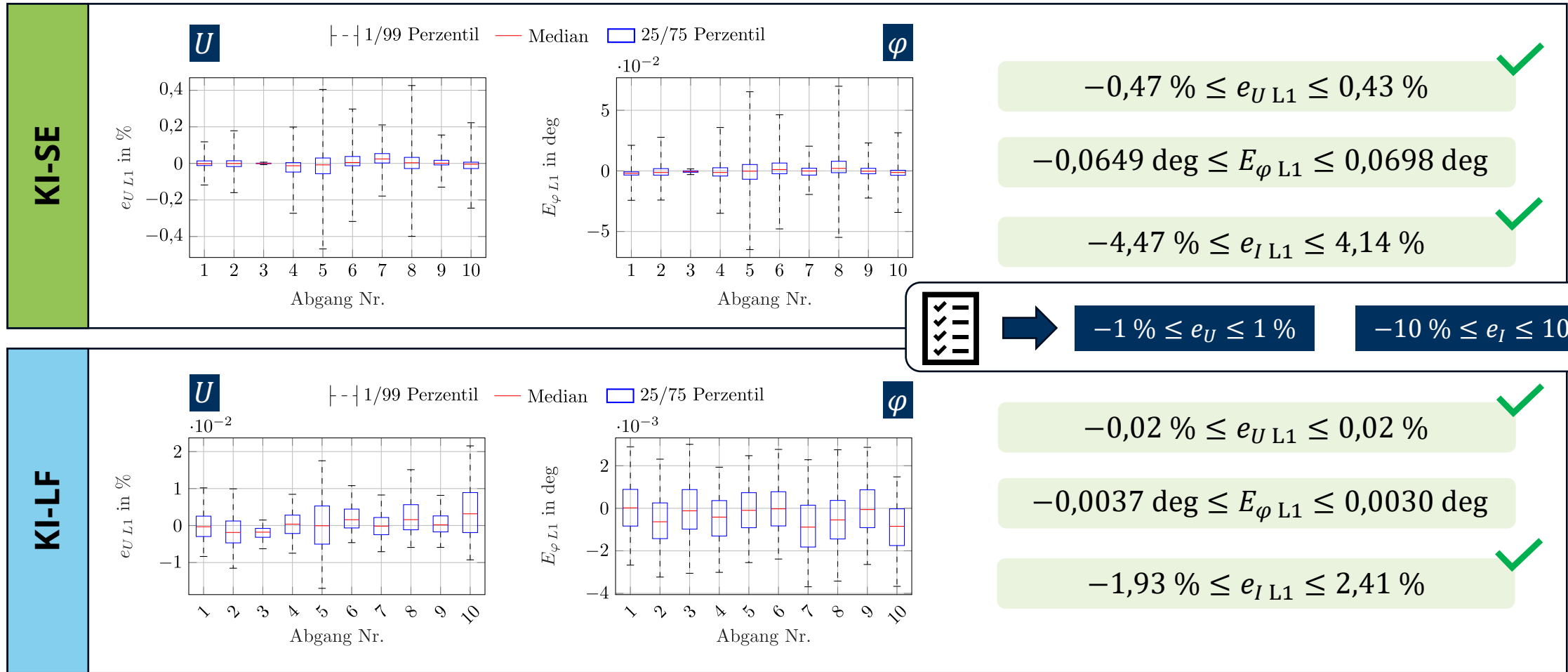
## Ländliches Niederspannungsnetz



- Zehn Abgänge
- 209 Knoten
- Leitungslänge (ges.): 7,3 km
- Leitungslänge (mittel): 24,7 m

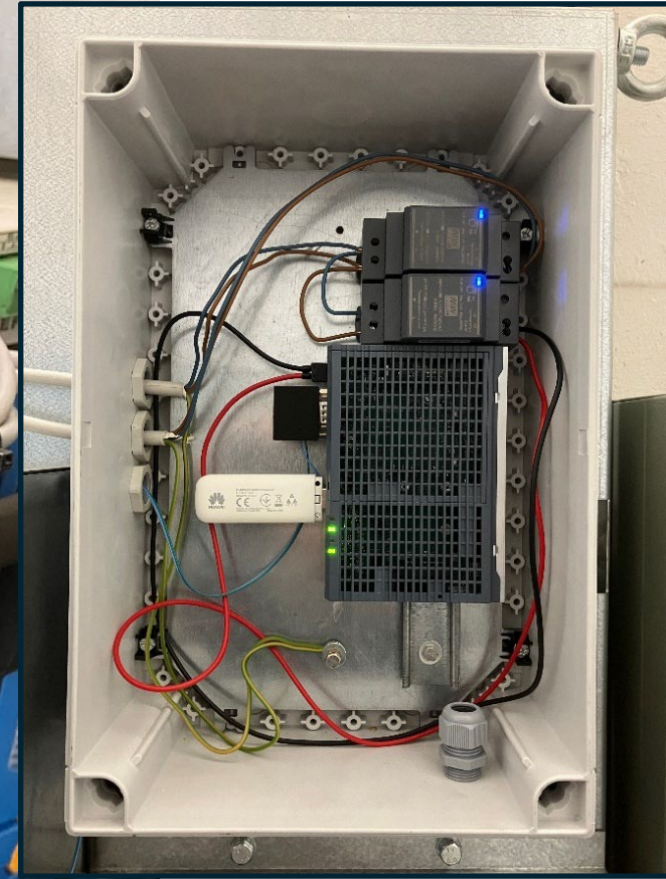
	2023	2030
$\sum P_{n PV}$	345 kW	580 kW
$\sum P_{n EM}$	84 kW	525 kW
$\sum P_{n WP}$	169 kW	280 kW

## Ergebnisse der KI-basierten Netzzustandsidentifikation



➔ **Ein 1. Versuch: Raspberry-PI mit KI-System in einer Netzstation**

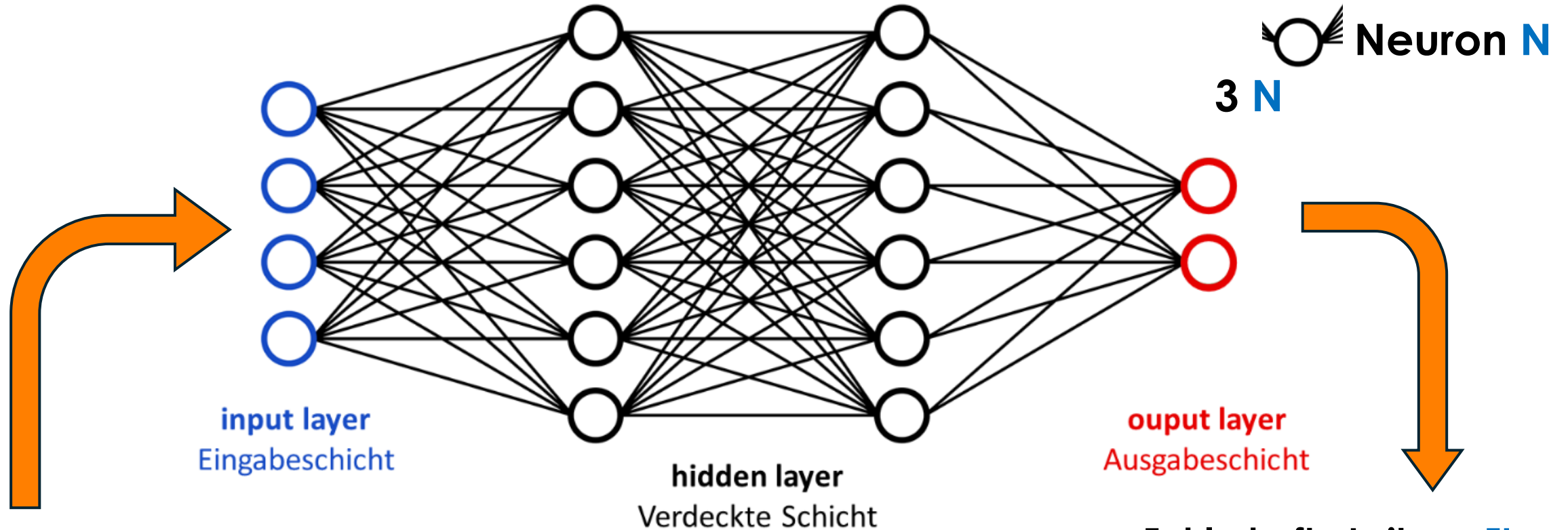




# Fehler in Stromverteilnetzen erkennen

-

## Die KI-basierte Kurzschlussortung



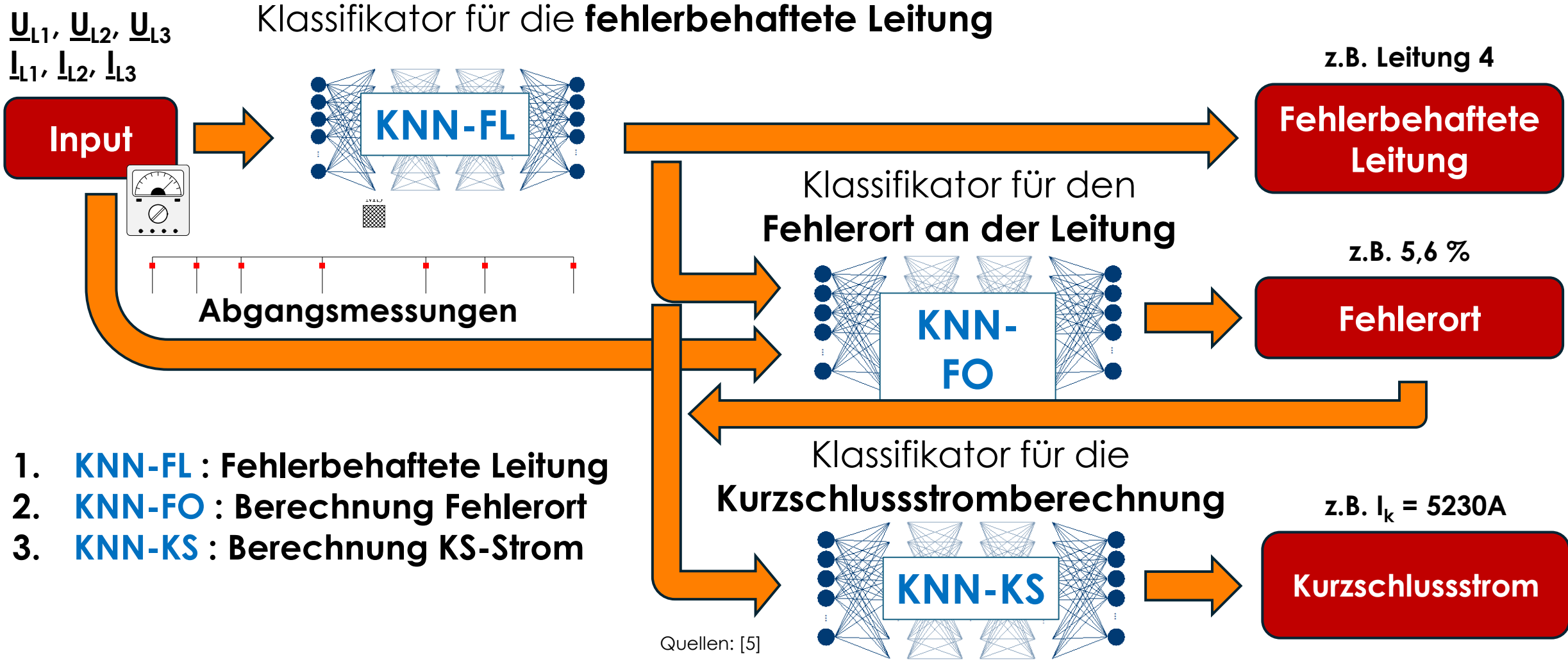
## Messgrößen Netzschutz

- $\underline{U}_{L1}, \underline{U}_{L2}, \underline{U}_{L3}$
- $\underline{I}_{L1}, \underline{I}_{L2}, \underline{I}_{L3}$

12 N

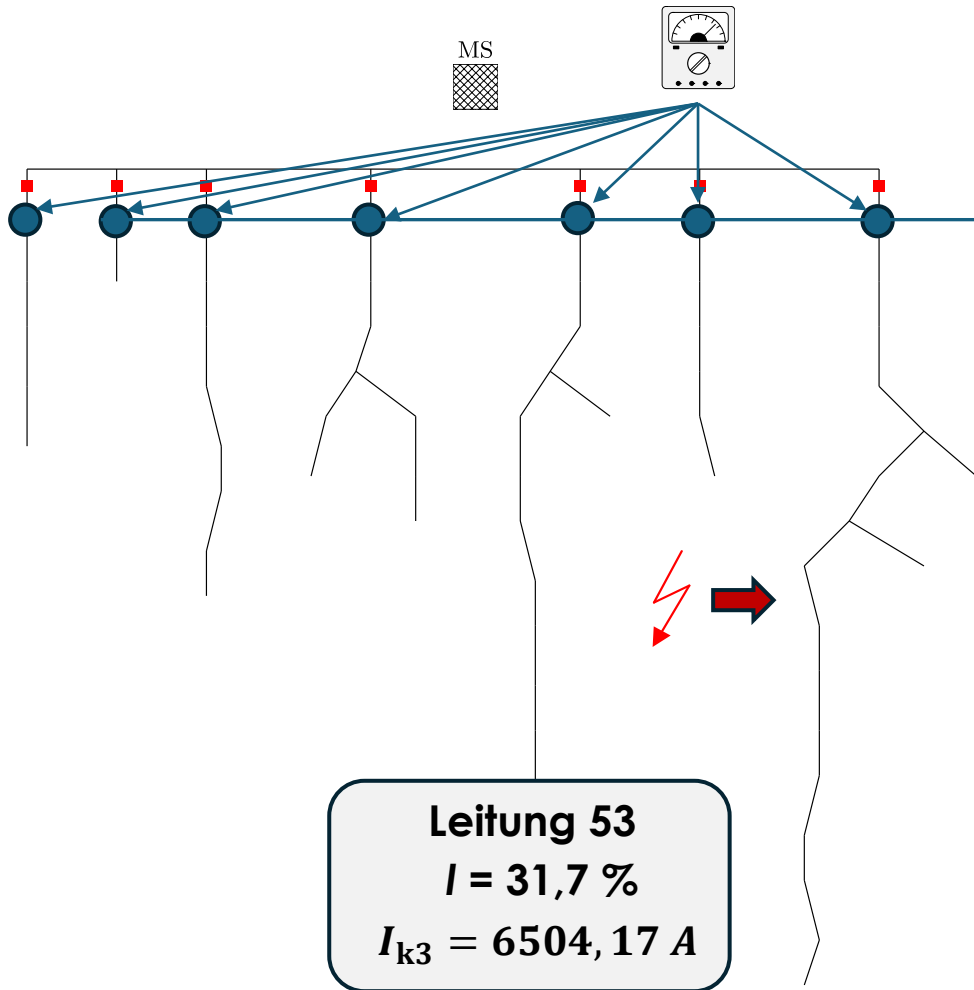
$$m_{\text{hidden}} = \frac{2}{3} m_{\text{in}} + m_{\text{out}} \quad 11 \text{ N}$$

- Fehlerhafte Leitung **FL**
- Fehlerort **FO**
- Kurzschlussstrom **KS**



1. **KNN-FL** : Fehlerbehaftete Leitung
2. **KNN-FO** : Berechnung Fehlerort
3. **KNN-KS** : Berechnung KS-Strom

Quellen: [5]



**KNN-FL : Leitung**



1,0000 ; 0,9998 ; -0,0022 ; 0,9223 ; 0,0784 ; 1,0006  
 1 1 0 1 0 1 → **Leitung 53**



**KNN-FO : Fehlerort**



**31,582 %**



**KNN-KS : Kurzschlussstrom  $I_k$**



$I_{k3} = 6505,53 A$



Quellen: [5]

# KI-Systeme prüfen

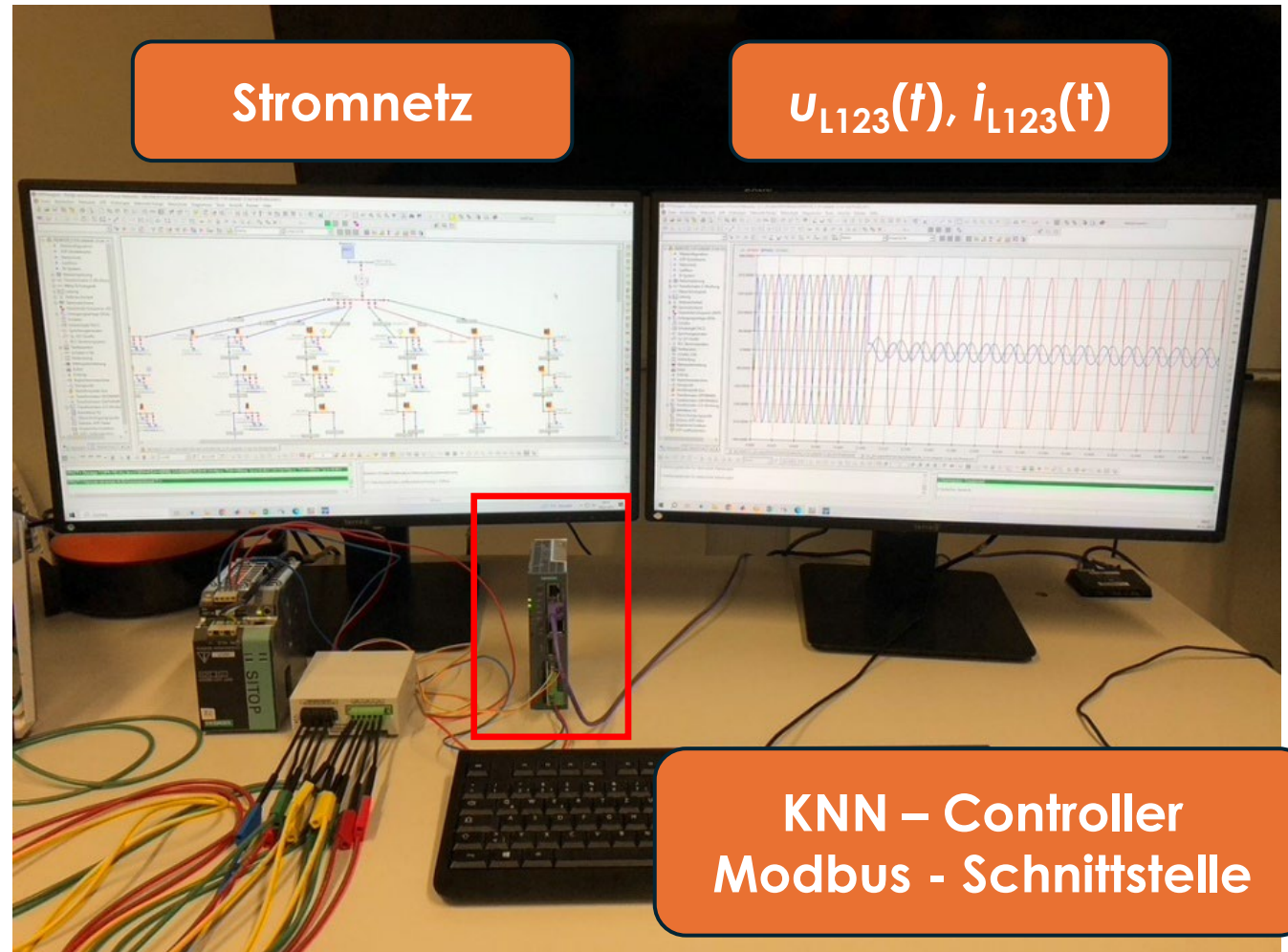
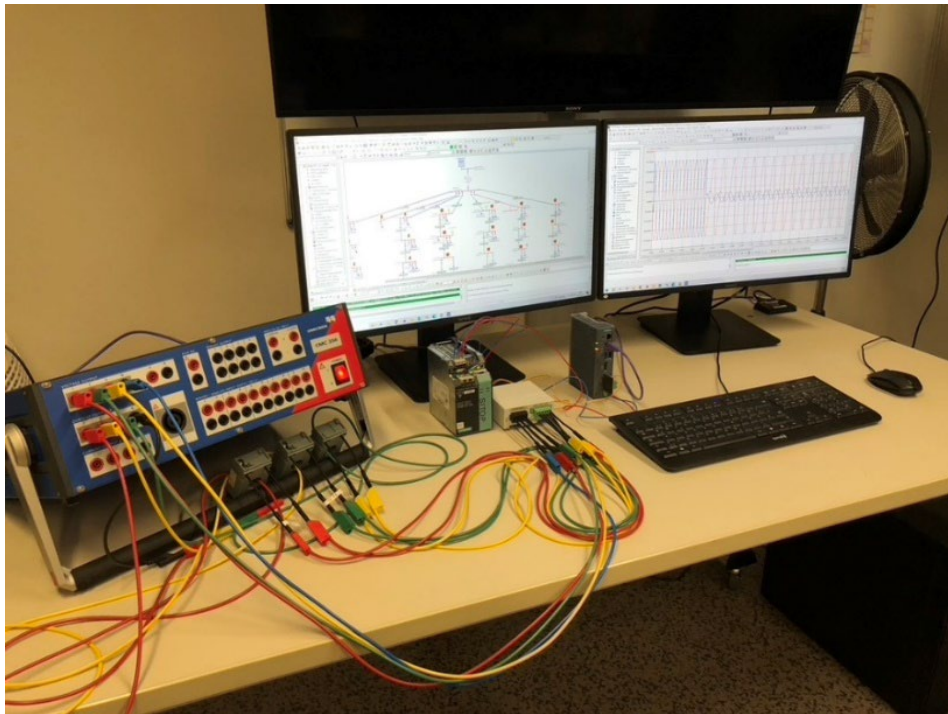
-

## Die Prüfumgebung im Labor

-

# VERTRAUEN SCHAFFEN

Berechnung dynamischer  
Netzvorgänge  $u_{L123}(t)$  und  $i_{L123}(t)$   
erforderlich

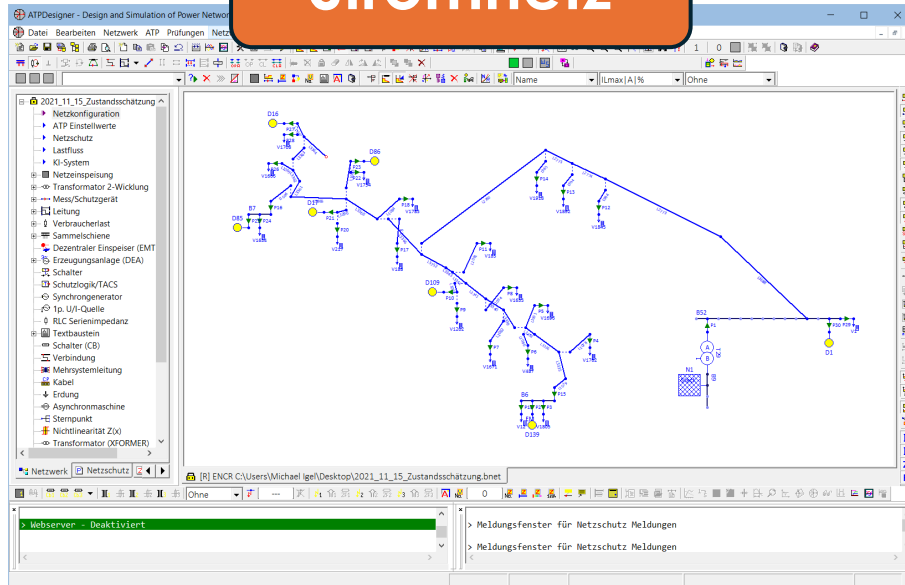


Berechnung eines stationären Netzzustandes mit Lastflussberechnung

$\underline{U}_{L123}$ ,  $\underline{I}_{L123}$ ,  $P$ ,  $Q$ , ... als Referenz (= "Wahrer Wert")

- Typprüfung mit bewährten mathematisch-algorithmischen Verfahren
- geringer Aufwand, beliebig viele Messorte möglich, kein Prüfverstärker erforderlich

**Stromnetz**



IP-basierte Kommunikation

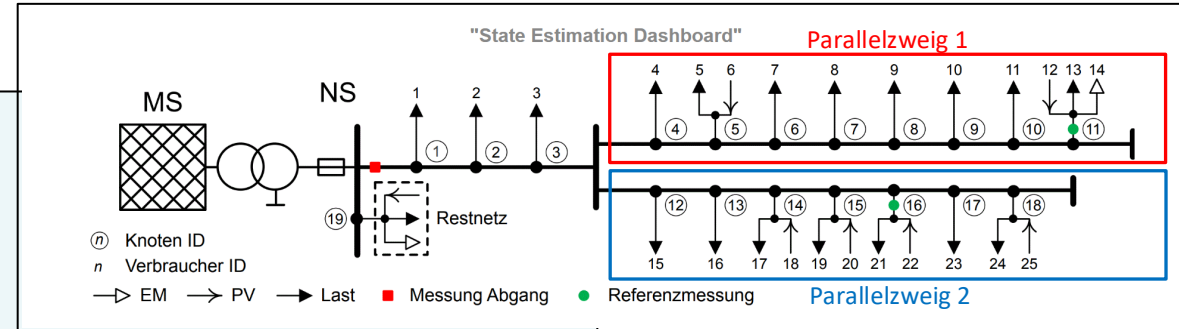
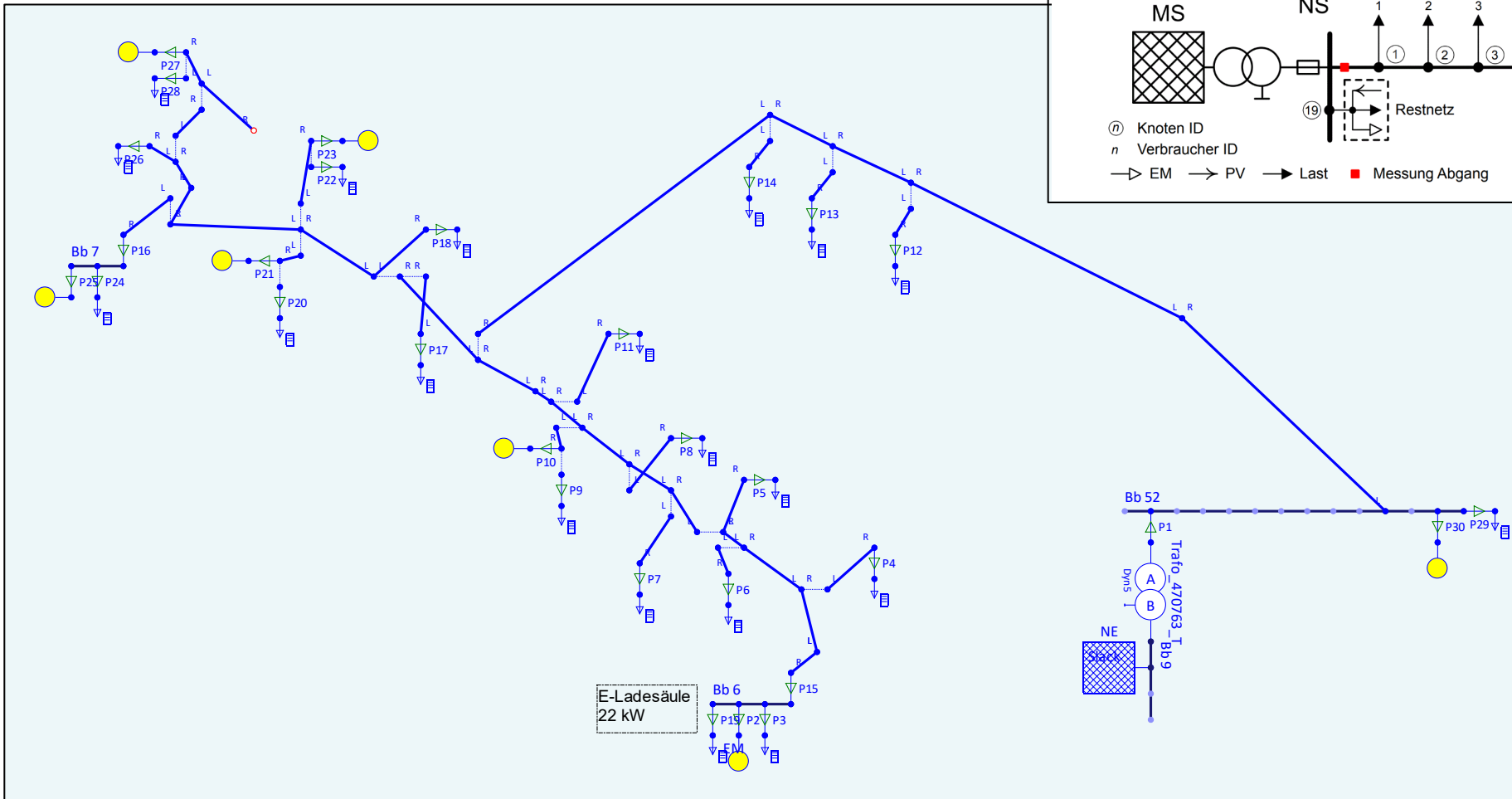


Virtuelles Netzwerk

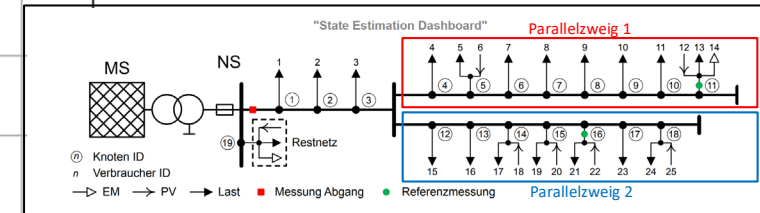
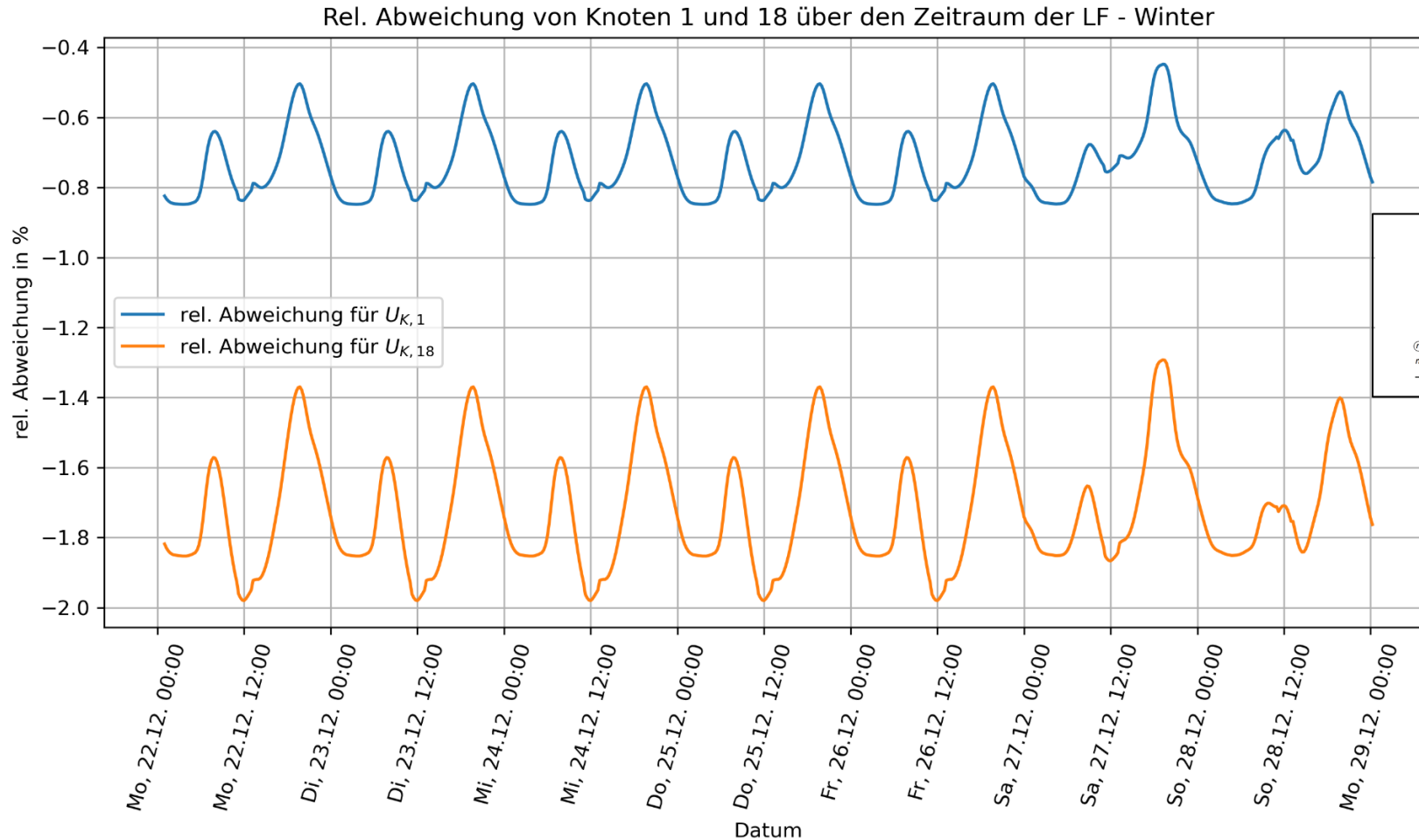
**Virtuelle Prüfumgebung**



## ➔ Niederspannungsnetz $U_n = 400V$



## Relative Abweichung der berechneten (LF) und geschätzten (KNN) Leiter-Erd-Spannungen



# Zusammenfassung



Die **KI-SE** kann die Netzzustände in den ausgewählten Referenzstromnetzen und Feldtestgebieten bei **minimaler Messinstrumentierung** mit einer **hohen Genauigkeit** identifizieren.



Die **KI-LF** kann die Lastflüsse in den ausgewählten Referenzstromnetzen und Feldtestgebieten mit einer **hohen Genauigkeit** berechnen.



Die **KI-OPF** kann **normativ unzulässige Netzzustände** mit einer **hohen Genauigkeit** unter **Berücksichtigung von regulatorischen und marktlichen Rahmenbedingungen** beheben.



KI-basierte Verfahren können Erdschlüsse detektieren und Kurzschlussorte berechnen.



Die Feldtests zeigen, dass die KI-basierte Netzsimulation **geringe Anforderungen an die Hardware** stellt und den Netzzustand mit **hoher Geschwindigkeit** schätzen, berechnen und optimieren kann.



**Typprüfung** kann **mit bekannten und bewährten Prüfmethoden und Prüfumgebungen** erfolgen



**Dr.-Ing. Andreas Winter**

Teamleiter Anschlusswesen / Verbandsarbeit  
+49 (0) 681 4030 2341  
andreas.winter@energis-netzgesellschaft.de



**Prof. Dr.-Ing. Michael Igel**

Institutsleiter  
+49 (0) 1756452194  
michael.igel@htwsaar.de

**energienetz**<sup>7</sup>  
saar  
Teil der **VSE**

- [1] Winter, A.; Igel, M.; Schegner, P. (2021): Supervised Learning Approach for State Estimation in Distribution Systems with missing Input Data. Hg. v. 2021 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe). Espoo, Finland.
- [2] Bundesregierung (2021): Klimaschutzpaket der EU-Kommission, [online] [www.bundesregierung.de](http://www.bundesregierung.de) [abgerufen am 11.02.2022].
- [3] Wikipedia: Complete neuron cell diagramm de.svg, [online] [www.wikipedia.de](http://www.wikipedia.de) [abgerufen am 12.02.2022].
- [4] A. Winter, B. Brandherm, M. Igel, P. Schegner, „Klassische Stromnetzrechnung in Kombination mit künstlicher Intelligenz zur Analyse und Diagnose von Stromverteilnetzen“, in Tutorial Schutz- und Leittechnik: Online Preview 2022, Berlin, 2022.
- [5] Igel, M.; Winter, A. (2024): KI-basierte Kurzschlussstromberechnung und KI-basierte Zentrale Kurzschlussortung. VDE/FNN. Tutorial Schutz- und Leittechnik 2024.
- [6] Winter, A.; Igel, M. (2022): Innovative maschinelle Lernverfahren für KI-basierte Lösungen zur Erdschlussrichtungserkennung und –ortung. VDE/FNN. Sternpunktbehandlung in Netzen bis 110 kV (D-A-CH).
- [7] Winter, A.; Raß, P.; Igel, M.; Schegner, P. (2023): Anwendung Künstlicher Intelligenz für die Zustandsschätzung in Stromnetzen – Validierung des Verfahrens mit einem Weighted Least Squares Algorithmus, VDE/ETG, ETG Kongress 2023.
- [8] A. Winter, „Entwicklung einer hybriden Künstlichen Intelligenz für die Identifikation und Optimierung von Netzzuständen in elektrischen Verteilungsnetzen,“ Dissertation, TU Dresden, 2025.