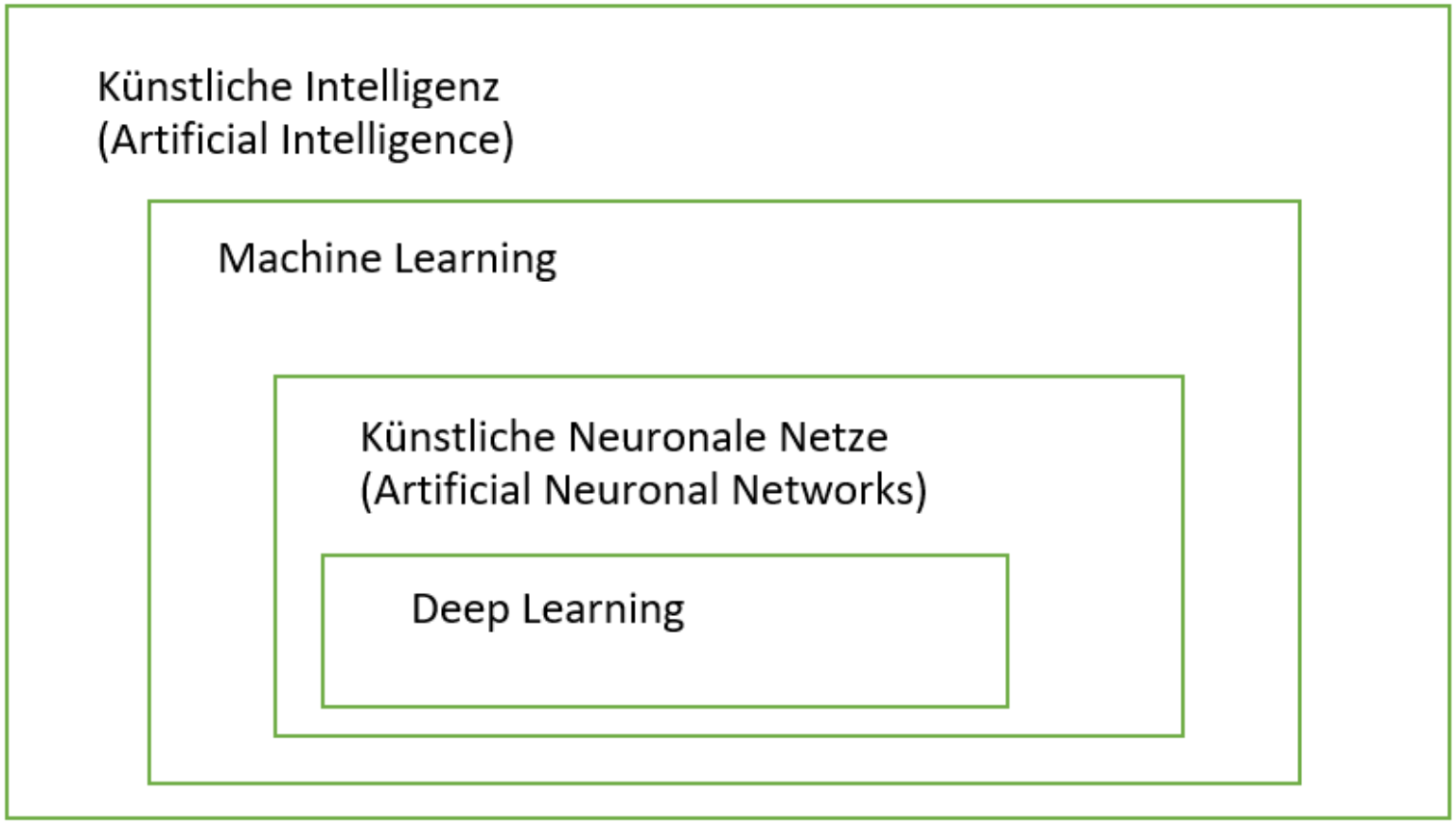
Künstliche Intelligenz

Fachschule für Maschinenbau und Mechatroniktechnik

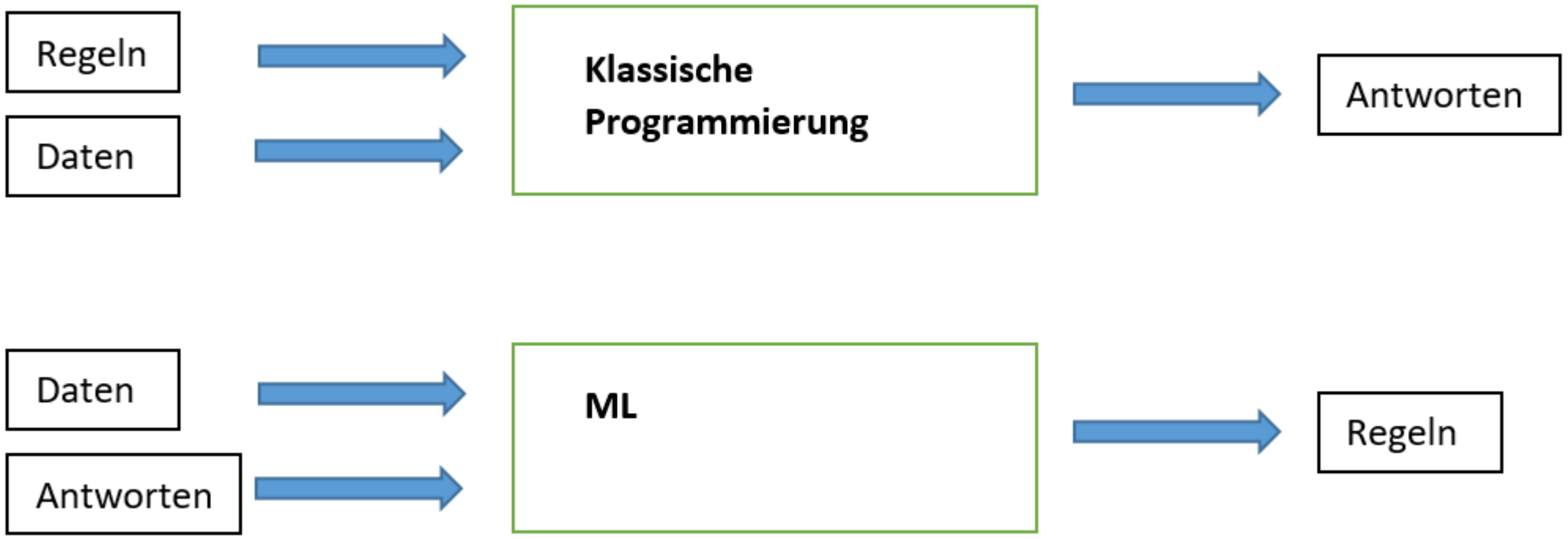
Staatliches Berufliches Schulzentrum | Herzogenaurach Höchstadt a. d. Aisch

Zusammenhang der Begrifflichkeiten



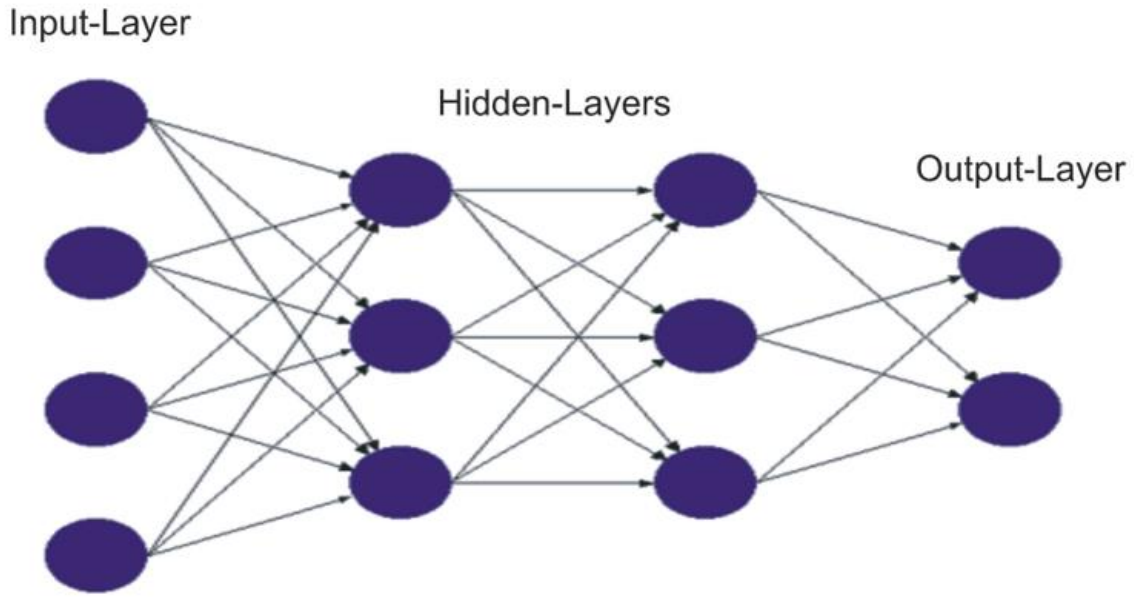
Zusammenhang der Begrifflichkeiten

- **KI (AI):** Der Versuch, normalerweise von Menschen erledigte geistige Aufgaben automatisiert zu lösen.
- **ML:** Die Generierung von Wissen aus Erfahrung.



Zusammenhang der Begrifflichkeiten

- **KNN (ANN):** Das Lernen durch schichtweise angeordnete künstliche Neuronen.

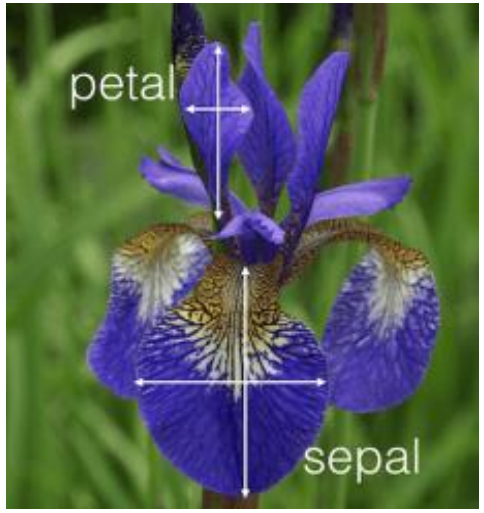


Quelle Bild: http://pille.iwr.uni-heidelberg.de/~ocr01/nnet.html

- **DL:** KNN mit mehreren (tiefen) Schichten.

Das erste Klassifizierungsproblem (Classification)

Iris Flower Dataset



Quelle Bild: http://blog.kaggle.com/2015/04/22/scikit-learn-video-3-machine-learning-first-steps-with-the-iris-dataset/

Iris setosa

Iris versicolor

Iris virginica

Quelle Bilder: https://en.wikipedia.org/wiki/Iris_flower_data_set

„Kochrezept“ zur KI Entwicklung

Auszug aus Iris Flower Dataset

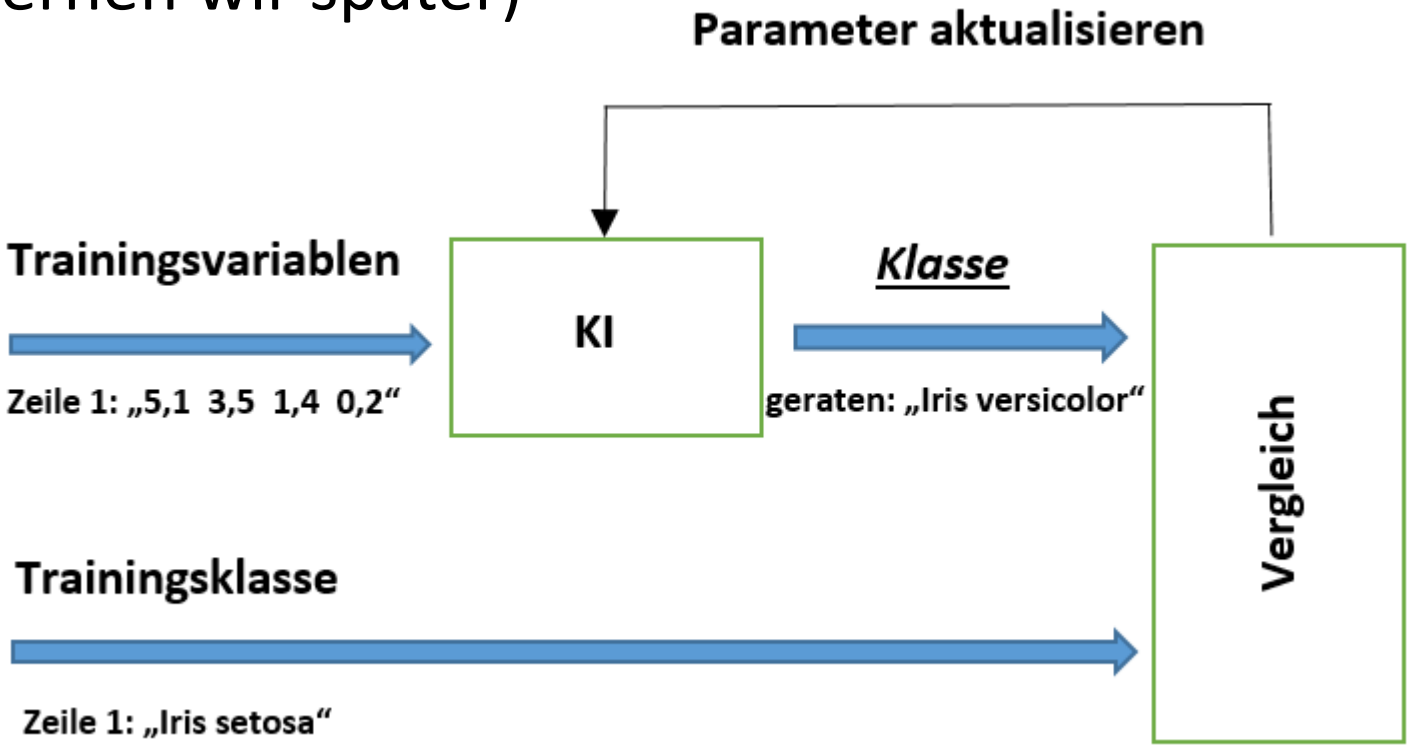
	sepal lenght	sepal width	petal lenght	petal width	class
Zeile 1	5,1	3,5	1,4	0,2	Iris setosa
Zeile 2	7,0	3,2	3,5	1,0	Iris versicolor
Zeile 3	6,3	3,3	6,0	2,5	Iris verginica
	…	…	…	…	…
Zeile 135	5,0	3,4	1,2	0,3	Iris setosa
	…	…	…	…	

Aufgabenstellung: Mit Hilfe von vier Variablen (sl, sw, pl, pw) soll die Klasse automatisch von der KI bestimmt werden.

Annahme: Die KI ist eine Black-Box, deren Inhalt zu einem späteren Zeitpunkt betrachtet wird.

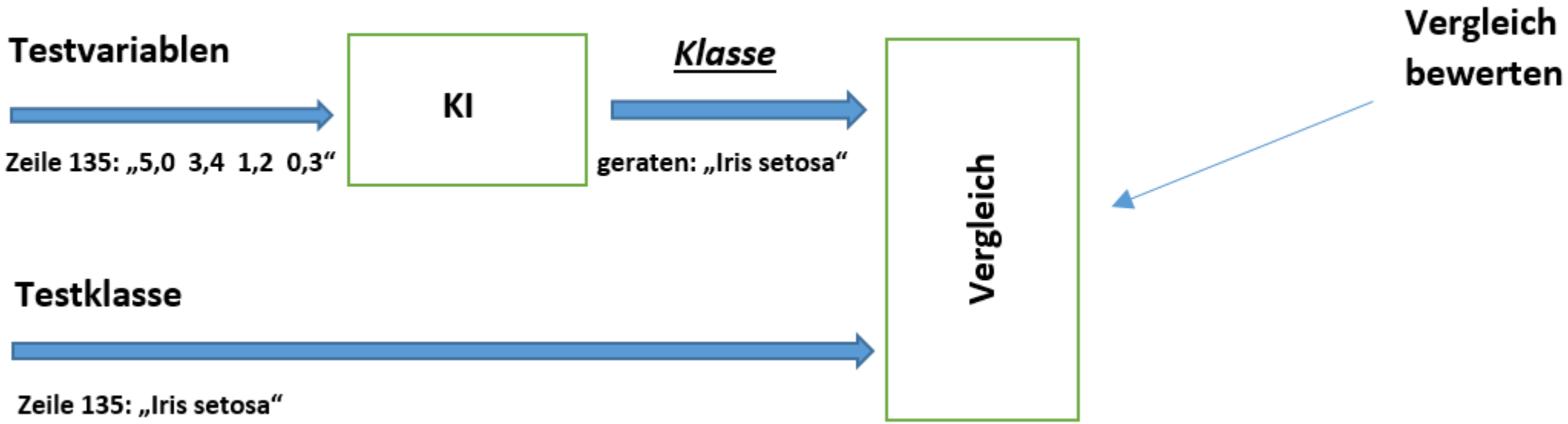
„Kochrezept“ zur KI Entwicklung

- Schritt 1: Datenvorbereitung
- Schritt 2: KI aufbauen (lernen wir später)
- Schritt 3: KI trainieren



„Kochrezept“ zur KI Entwicklung

- Schritt 4: KI testen



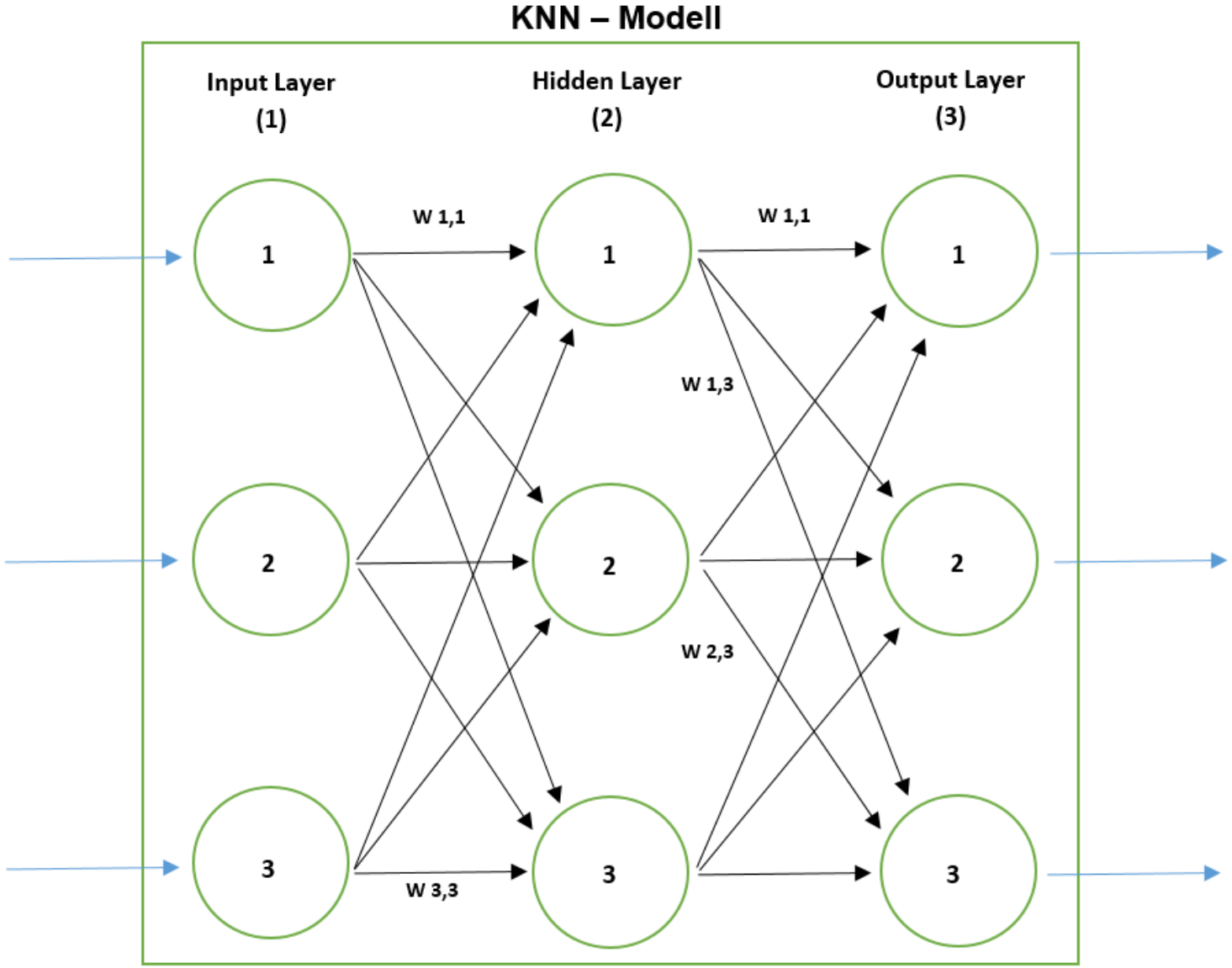
„Kochrezept“ zur KI Entwicklung

- Schritt 5: KI Entwurf optimieren (vorherige Schritte wiederholen)
- Schritt 6: KI Produktiv einsetzen

„Kochrezept“ zur KI Entwicklung

Aufgabe: Erstellen Sie einen Ablaufplan zur Veranschaulichung der Schritte unseres „Kochrezeptes“.

Aufbau künstlicher neuronaler Netze

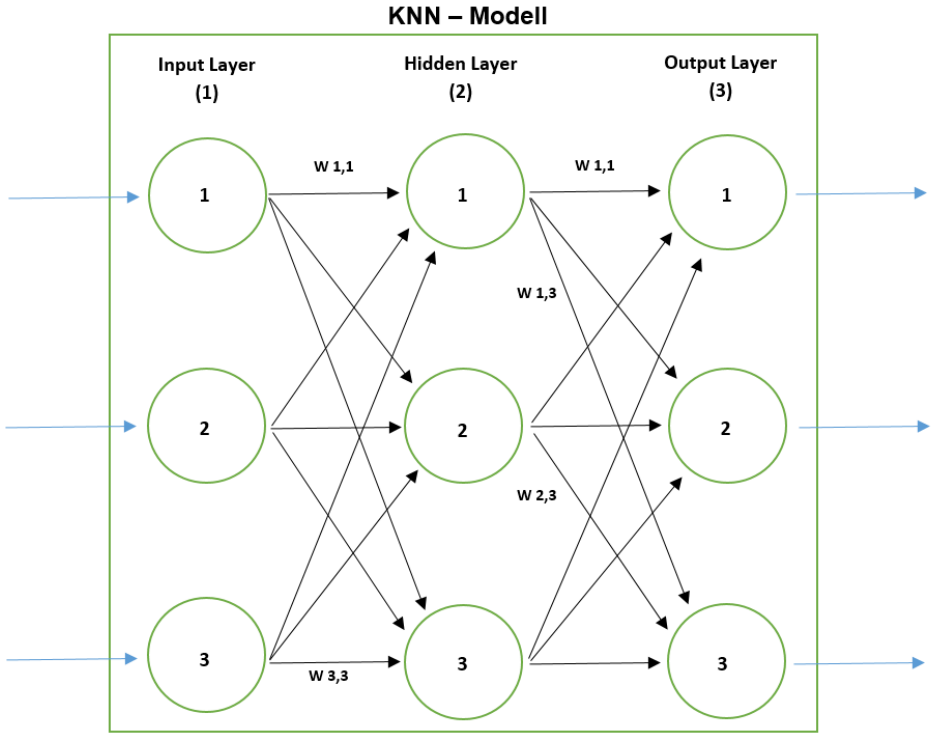


Aufbau künstlicher neuronaler Netze

Aufgabe: Tragen Sie mindestens drei weitere Gewichte in die obere Grafik ein.

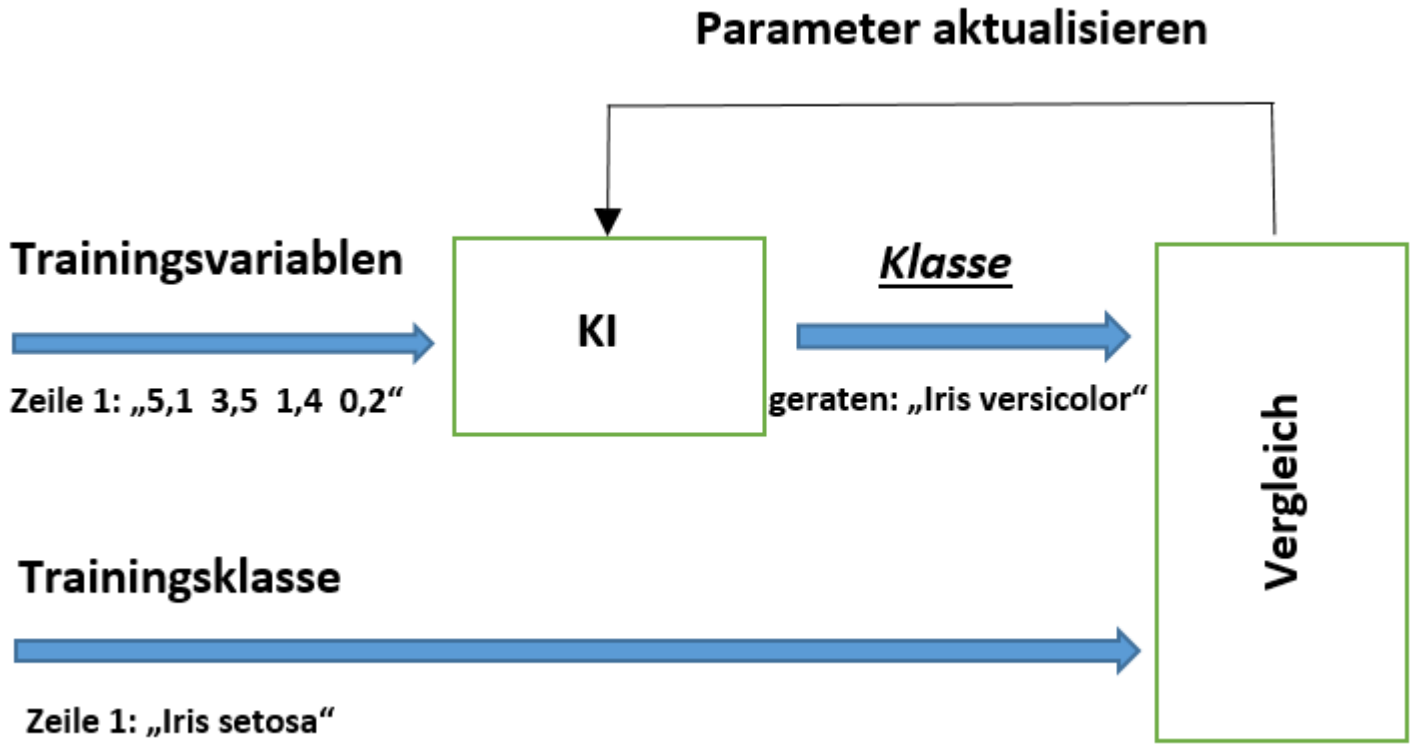
Aufbau künstlicher neuronaler Netze

1. Wie viele Knoten müssen Input- und Output-Layer für die Klassifizierung von Schwertlinien haben?
2. Wie viele Hidden-Layer sind notwendig und wie viele Knoten soll jede dieser Schichten haben?



Aufbau künstlicher neuronaler Netze

Zu Frage 1: Der Input-Layer muss vier Knoten besitzen, da die KNN mit vier Variablen (sl, sw, pl, pw) „gefüttert“ wird. Output-Layer muss drei Konten haben.



- 0: Iris setosa
- 1: Iris versicolor
- 2: Iris verginica

z.B. (0,2; 0,7; 0,1)

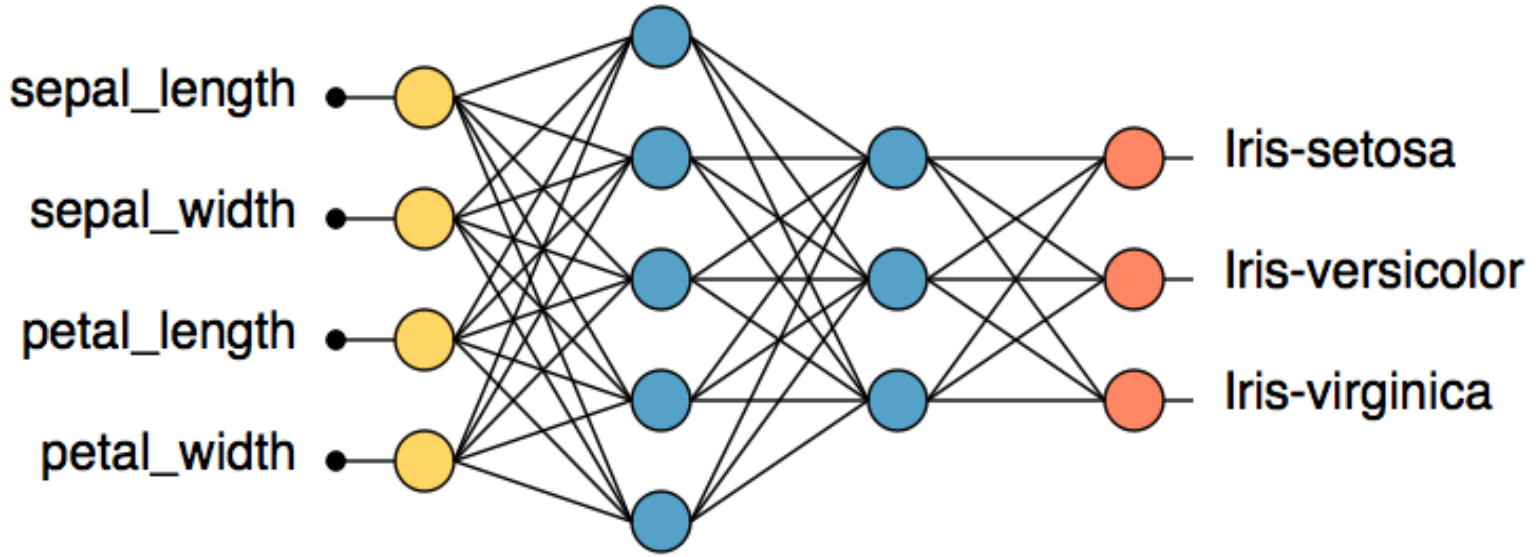
Aufbau künstlicher neuronaler Netze

Zu Frage 2: Zu dieser Frage gibt es keine eindeutig richtige Antwort.

- Von Aufgabenstellung abhängig.
- Wenn die Anzahl zu klein ist, kann es passieren, dass das KNN nicht gut trainiert wird und keine Klassen zuordnen kann.
- Ist die Anzahl zu groß, braucht das KNN sehr lange zum Trainieren und lernt die Trainingsdaten auswendig, kann aber die Testdaten nicht richtig zuordnen.
- Das KNN hat in diesem Fall also nicht das „System“ verstanden, um neue Daten zu klassifizieren. Es hat lediglich die Testdaten auswendig gelernt.

Aufbau künstlicher neuronaler Netze

Unser KNN zur Klassifizierung von Schwertlilien ist somit wie folgt aufgebaut:

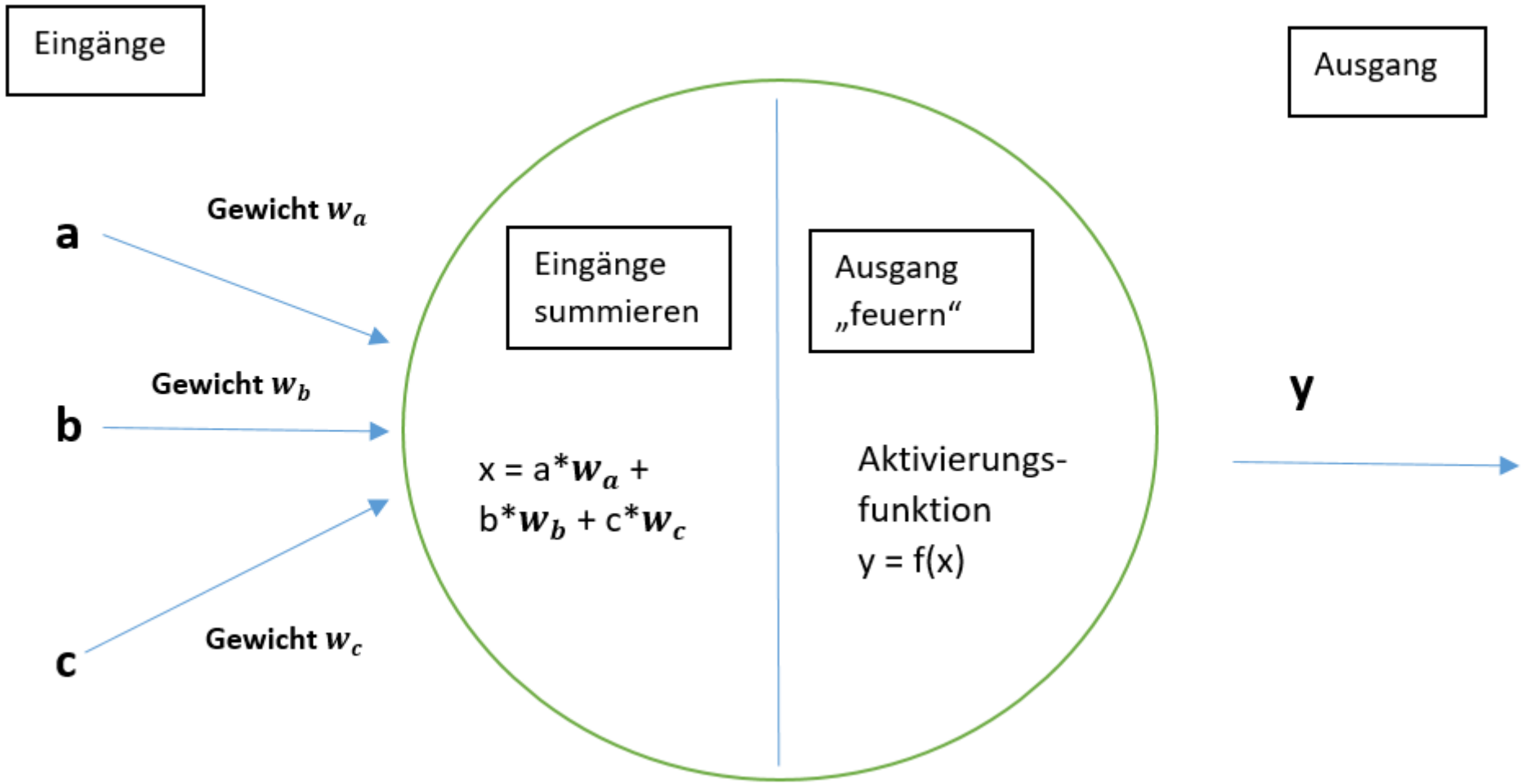


Quelle Bild: https://github.com/tanishq9/Iris-Classifier-Android

Aufbau künstlicher neuronaler Netze

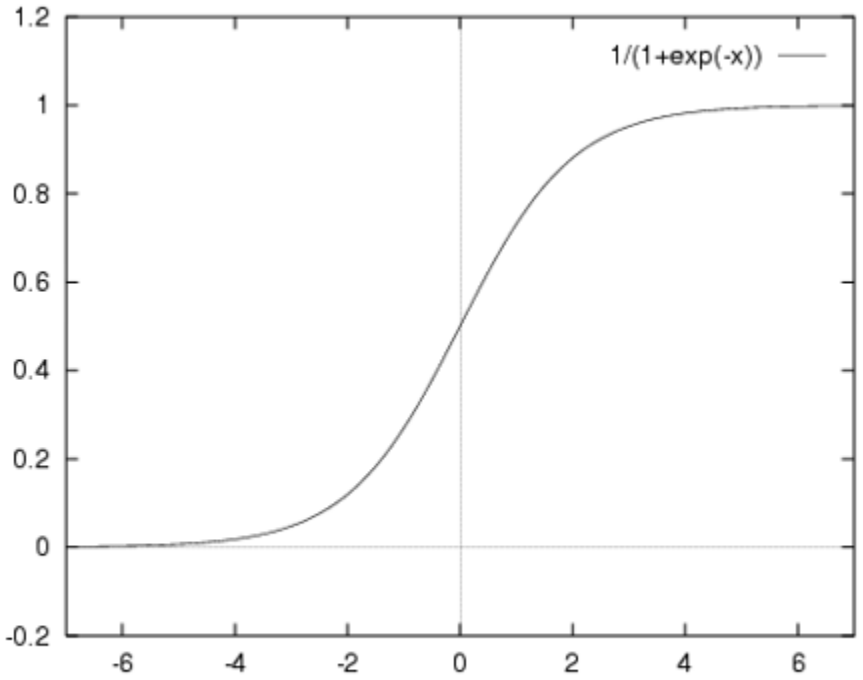
Aufgabe: Schauen Sie sich noch einmal das Kochrezept an und ordnen Sie Ihre neuen Kenntnisse darin ein. Ersetzten Sie gedanklich die Black-Box durch unser künstliches neuronales Netz.

Aufbau eines künstlichen Neurons



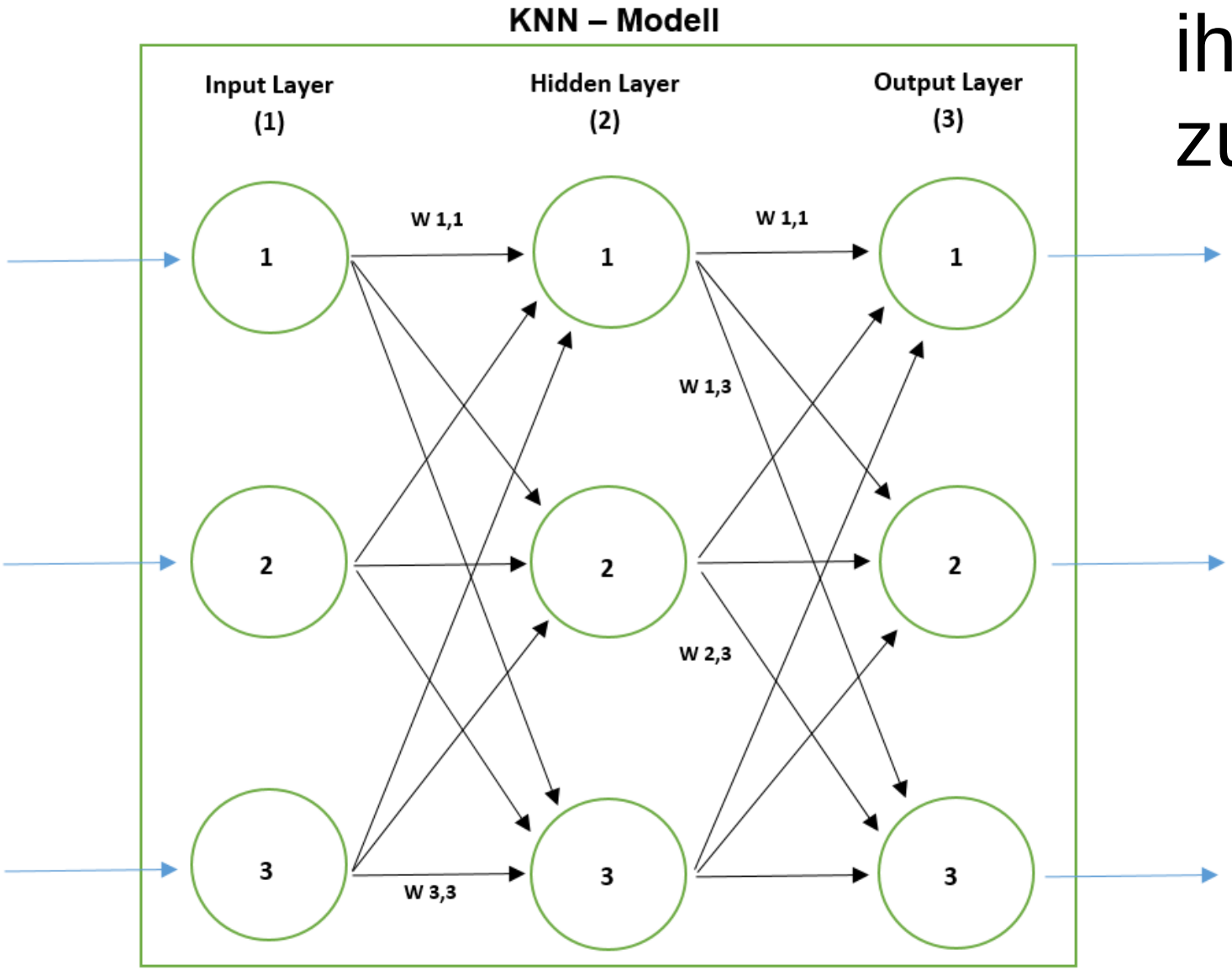
Aufbau eines künstlichen Neurons

Sigmoidfunktion $y = \frac{1}{1 + e^{-x}}$



Quelle Bild: https://computing.dcu.ie/~humphrys/Notes/Neural/sigmoid.html

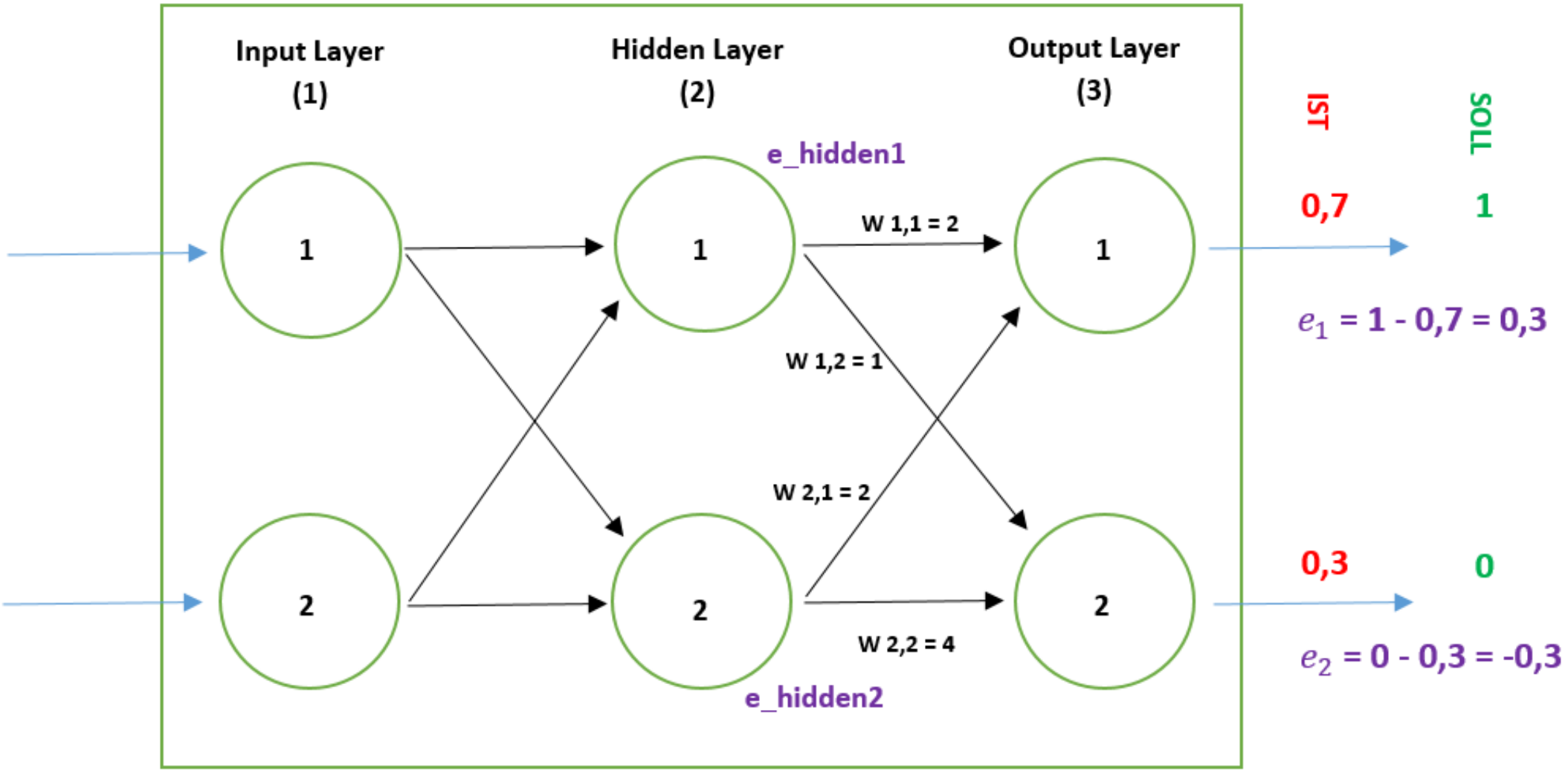
Feed Forward



Aufgabe: Wie werden die Eingabedaten auf ihrem Weg vom Input zum Output verarbeitet?

Backpropagation

$e\_hidden1 = w_{1,1} \times e_1 + w_{1,2} \times e_2$

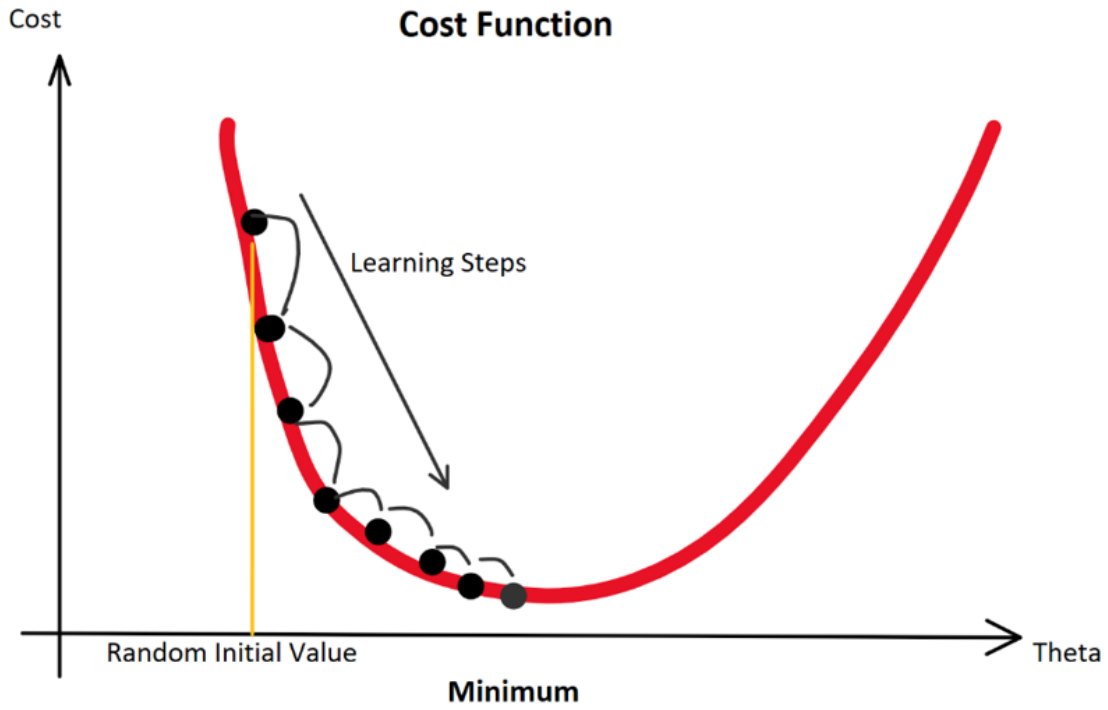


Backpropagation

Aufgabe:

- Stellen Sie die Gleichung für *e_hidden2* auf.
- Berechnen Sie die konkreten Werte für *e_hidden1* und *e_hidden2*.

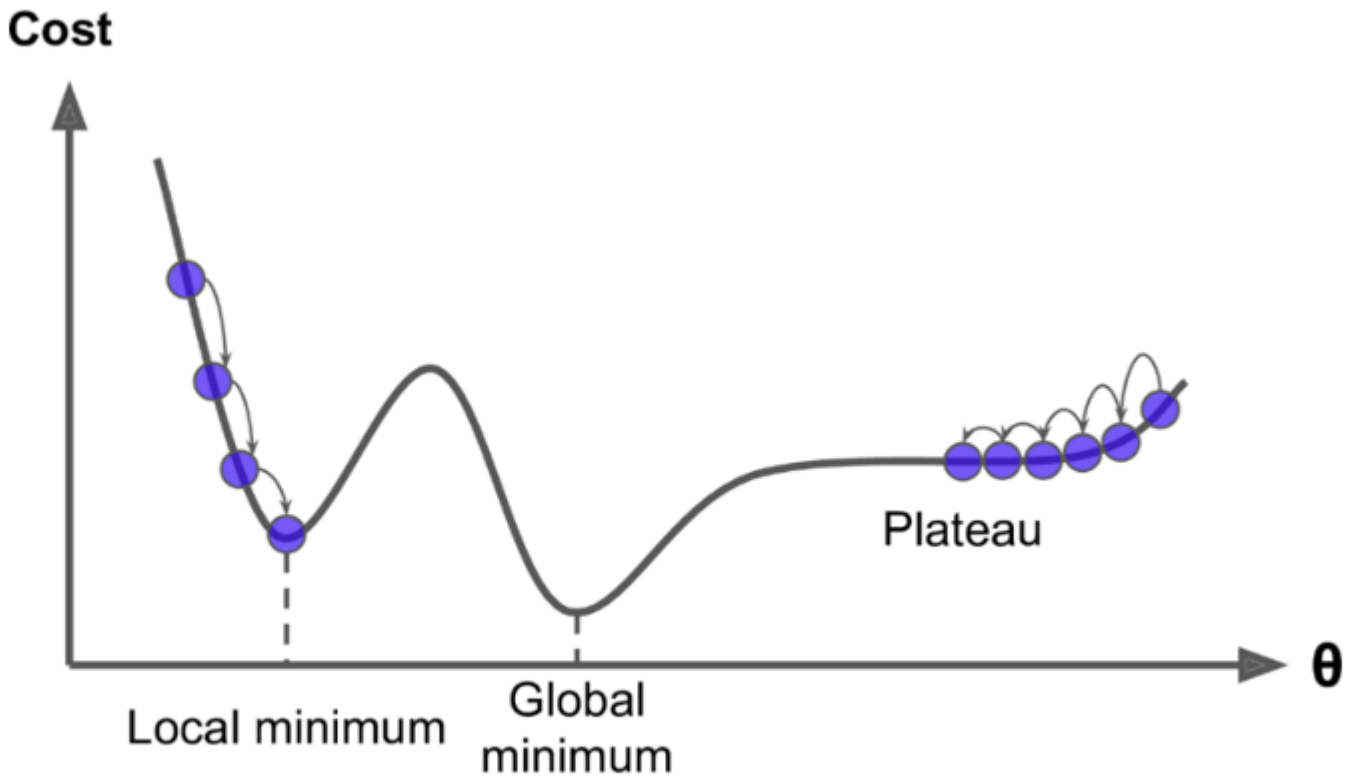
Aktualisierung der Gewichte



Quelle Bild:https://data-science-blog.com/blog/2019/01/13/training-eines-neurons-mit-dem-gradientenverfahren/

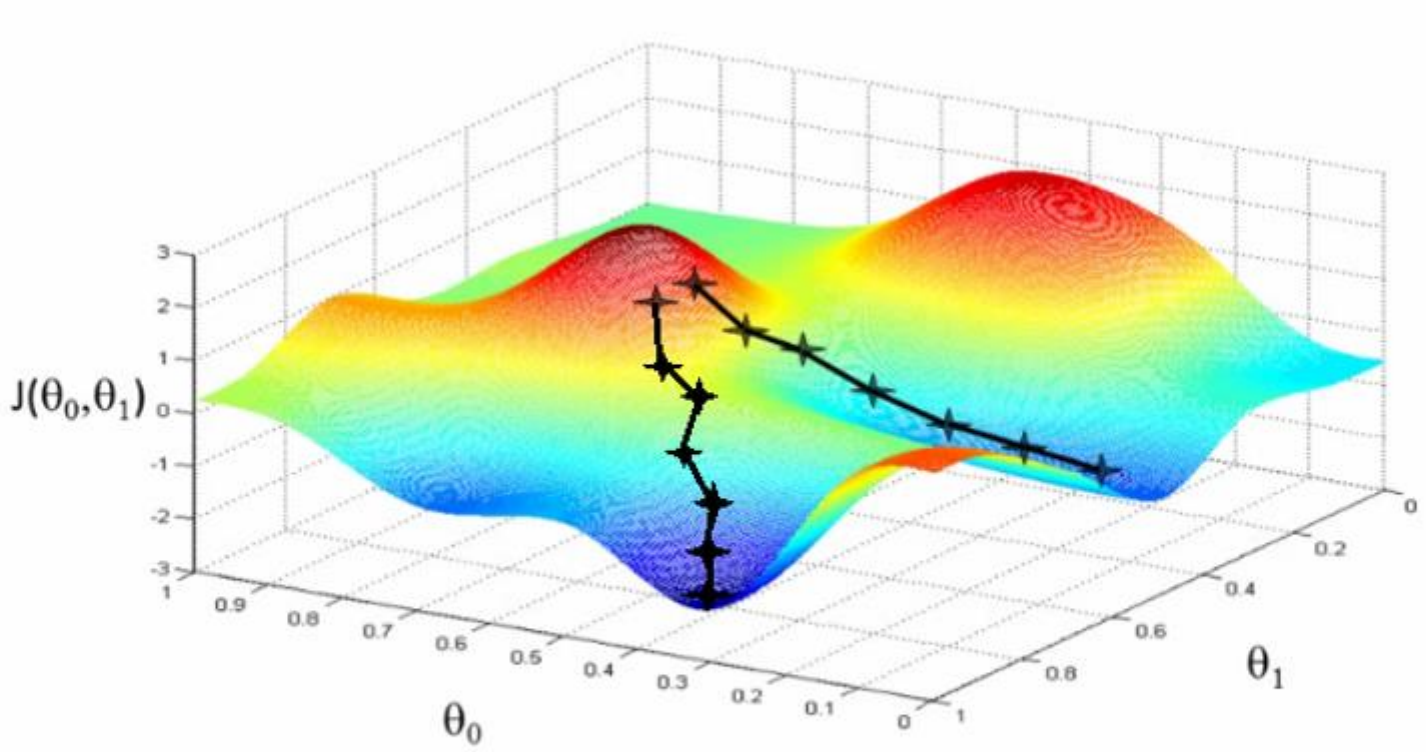
neu $W_{j,k}$ = alt $W_{j,k}$ - α x $\Delta W_{j,k}$ (negativ, da wir uns gegen die Steigung bewegen)

Aktualisierung der Gewichte



Quelle Bild: https://www.oreilly.com/library/view/hands-on-machine-learning/9781491962282/ch04.html

Aktualisierung der Gewichte



Quelle Bild: https://www.freecodecamp.org/news/building-a-3-layer-neural-network-from-scratch-99239c4af5d3/

Praktische Beispiele