



1.4 Machine Learning

Zusatz: Normalisierung

- Normalisierung bzw. auch “**Min-Max-Skalierung**” genannt wird verwendet, um **Features so zu verändern**, dass sie **ähnlich zueinander** werden → Daten sollen ähnliche und somit **vergleichbare Maßstäbe** annehmen
- Dabei werden sie oft auf Bereiche zwischen **[0,1]** oder **[-1,1]** skaliert
- Normalisierung kommt zum Einsatz, wenn Merkmale der Features unterschiedliche Maßstäbe haben und sie so nicht vergleichbar sind
- **Achtung!** Es gibt unterschiedliche Methoden, um Daten einander anzugleichen und sie zu “normalisieren”!

Zusatz: Normalisierung

- **Standardisierung** kommt dann zum Einsatz, wenn Daten sich so verteilen, dass man eine **Gauß'sche Glockenkurve** zeichnen kann → **Daten sind normalverteilt**
- Standardisierung wird verwendet, wenn der **Mittelwert 0 und** die **Standardabweichung eine Einheit** sein sollen
- Wird oft als **Z-Score-Normalisierung** bezeichnet

Ressourcen zu Normalverteilung:

- <https://artemoppermann.com/de/batch-normalisierung-in-deep-learning/>
- <https://www.youtube.com/watch?v=ecjN6Xpv6SE>
- <https://www.ffe.de/veroeffentlichungen/kuenstliche-intelligenz-data-preprocessing-und-paradigmenwechsel-der-programmierung/>
- <http://patternrecognition.tech/modellentwicklung/preprocessing/>

Zusatz: Cross Validation

- Um sicherzustellen, dass ein trainiertes Modell tatsächlich korrekt ist und nicht zu viel an **Rauschen (aka “Noise”)** aufgenommen hat, verwendet man **Cross-Validation**
- Man nimmt einen **Teil des Datensatzes**, gibt ihn beiseite und verwendet ihn später, um das Modell damit zu testen
- Vgl. Kochen: Für ein bestimmtes Rezept benötigt man insgesamt 500ml Wasser; 150ml werden davon beiseite gegeben, um diese zu einem späteren Zeitpunkt zu verwenden

Zusatz: Cross Validation

- Es gibt **verschiedene Cross-Validation-Verfahren**, die angewendet werden können:
 - LOOCV (Leave One Out Cross Validation)
 - Leave-P-Out Cross Validation
 - K-Fold Cross Validation
 - Stratified K-Fold Cross Validation

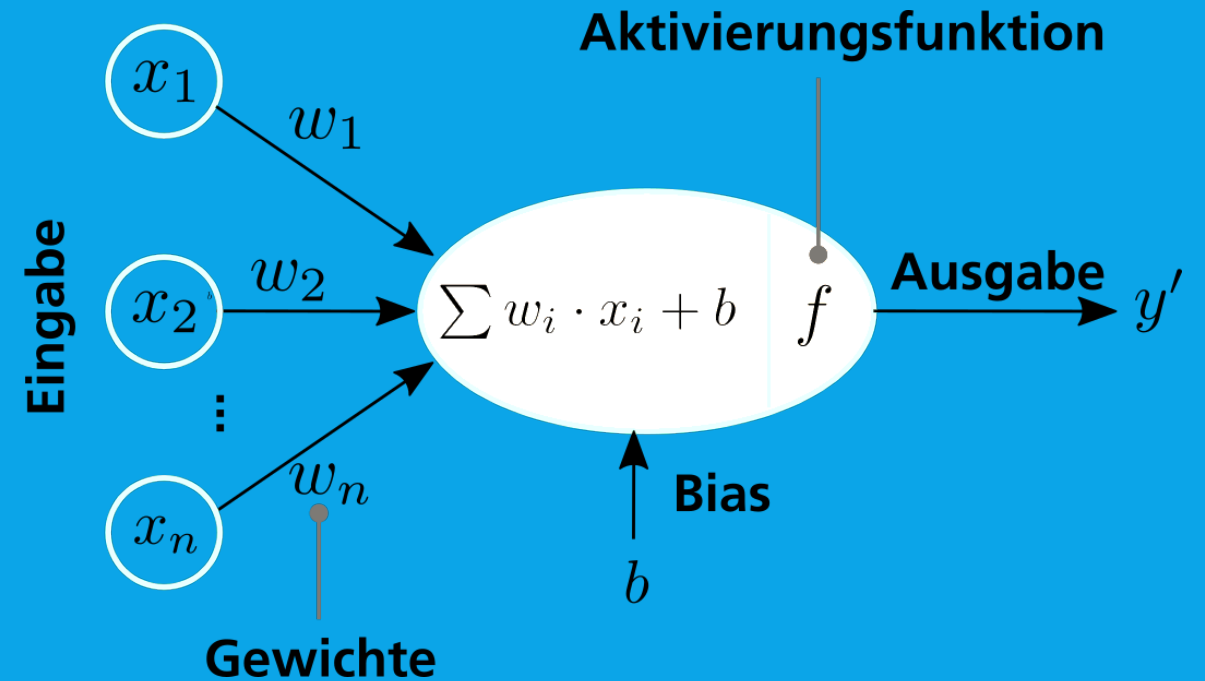
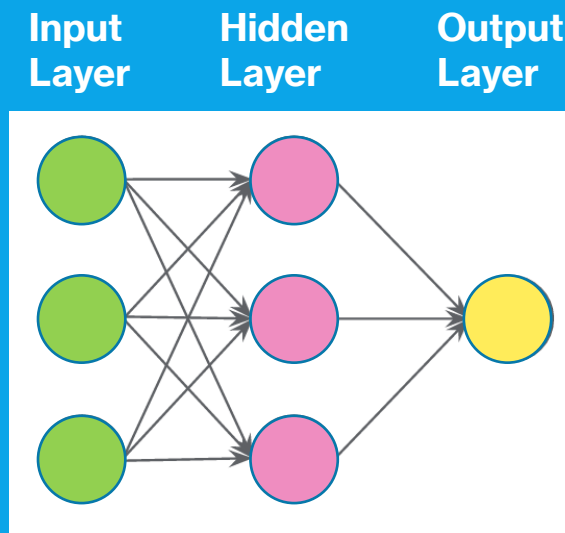
Ressourcen zu Cross Validation

- <https://towardsdatascience.com/cross-validation-in-machine-learning-72924a69872f>
- <https://www.geeksforgeeks.org/cross-validation-machine-learning/>
- <https://www.youtube.com/watch?v=fSytzGwwBVw>



1.5 Deep Learning & Neural Networks

Berechnung von KNNs



Ergänzung: Bias in Neuronalen Netzwerken

- Ein Bias in KNNs ist eine **Konstante**
- Sie sind ein **zusätzlicher Input für den nächsten Layer** – der Bias wird in KNNs jedoch **nicht vom vorhergehenden Layer** beeinflusst!
- Der Bias stellt sicher, dass auch dann eine „Aktivierung“ in einem Neuron stattfindet, selbst wenn die Inputs den Wert 0 haben
- Der Bias kann das **Ergebnis der Aktivierungsfunktion** in eine **positive** oder **negative Richtung verschieben**

Ergänzung: Bias in Neuronalen Netzwerken

- Würde man den Bias in einem KNN weglassen, kann es sehr leicht passieren, dass das Modell „overfitted“
- Durch die „Zugabe“ des Bias¹ wird die **Varianz reduziert**, d.h. das Modell kann **besser generalisieren¹** und ist **flexibler im Umgang mit neuen Daten**

1: Generalisierung in ML bedeutet, dass ein Modell besser mit neuen Daten umgehen kann, besser adaptiert und akkurate Vorhersagen treffen kann

Ressourcen Bias in KNNs:

- <https://afteracademy.com/blog/what-is-bias-in-artificial-neural-network>
- <https://stackoverflow.com/questions/2480650/what-is-the-role-of-the-bias-in-neural-networks>
- <https://www.geeksforgeeks.org/effect-of-bias-in-neural-network/>
- <https://medium.com/fintechexplained/neural-networks-bias-and-weights-10b53e6285da>