



©UDE

UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

Grundlagen des maschinellen Lernens

Einführung

Prof. Dr. Volker Gruhn

se Lehrstuhl für
Software Engineering

1. Künstliche Intelligenz (KI) und Maschinelles Lernen (ML)
 - 1.1. KI im täglichen Einsatz
 - 1.2. Abgrenzung und Definition
2. Kategorien des ML
 - 2.1. Supervised Learning
 - 2.2. Unsupervised Learning
 - 2.3. Semi-Supervised Learning
 - 2.4. Reinforcement Learning
3. Herausforderungen

- In der Literatur unterscheidet sich häufig
 - die verwendete Notation. Wir versuchen, möglichst einheitlich vorzugehen, können das aber nicht über alle Einheiten garantieren.
 - die verwendete Terminologie. Wir versuchen, möglichst einheitlich vorzugehen und die englische Nomenklatur zu bevorzugen.
- Oft sind die Beispiele zur Veranschaulichung (sehr) einfach gewählt, um eine bessere Vorstellung zu ermöglichen. Reale Probleme sind meist mehrdimensional. Hier liegt auch die wahre Stärke von KI!
- Manche Sachverhalte werden sich in den Vorlesungseinheiten wiederholen, u.a. wenn sich Details ändern.

Künstliche Intelligenz im Alltag

Anwendungen

KI im täglichen Einsatz: Beispiele

► KI und ML
► KI im täglichen Einsatz



- **Spracherkennung und -Synthese:**

Identifizierung von Worten aus gesprochener Sprache und Erkennen der Intention

- Digitale Assistenten: Alexa, Siri, Cortana, ...
- Smart-Home Steuerung
- Interaktion mit dem Bordcomputer im Auto während der Fahrt

- **Chatbots**

- **Betrugserkennung**

- Kreditkartenbetrug
- Betrug in der Arzneimittelabrechnung

- **Medizin**

- Diagnose von Krankheiten
- Analyse von Bildern zur Erkennung von Tumoren
- Entwicklung von Arzneimitteln
- Identifizieren von Biomarkern für die Diagnose einer Krankheit

- **Auswertung von Nutzerverhalten**

- Suchmaschinen: kontinuierliches Lernen aus dem Verhalten von Nutzer*innen zur Verbesserung von Suchergebnissen
- Musik- und Streamingdienste
- Online-Handel

Abgrenzung und Definition

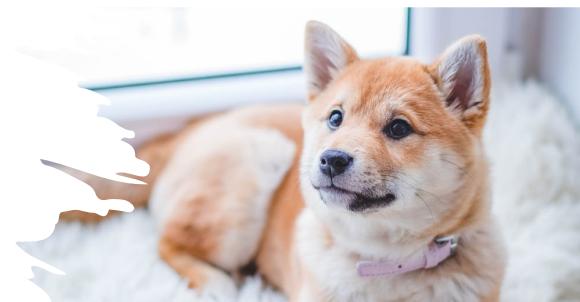
Künstliche Intelligenz (KI) und
Maschinelles Lernen (ML)

- Anforderung: Sortieren einer Liste mit Zahlen
 - $[1,3,6,2,5,8,7,9,4] \rightarrow [1,2,3,4,5,6,7,8,9]$
- Ein Software-Entwickler gibt der Maschine vor, nach welchen Regeln/Schritten ein Problem gelöst werden soll
 - Wir nennen den Regelsatz *Algorithmus*
- Bedingung: Wir kennen die Regeln und können sie (maschinenlesbar) formulieren
 - Das nennen wir *programmieren*
 - Die Programmierung bestimmt das Verhalten der Maschine und wird vom Programmierer vorgegeben
- Aber: Was ist, wenn wir die Regeln nicht kennen?
- Beispiel: BubbleSort
- Regeln:
 - Nimm die erste Zahl und vergleiche sie mit dem Nachbarn
 - Wenn die Zahl größer ist als der Nachbar, tausche die Plätze
 - Wiederhole Schritte 1+2 für jedes Zahlenpärchen
 - Wiederhole Schritte 1-3 für alle Elemente

```
def bubbleSort(list):  
    n = len(list)  
    # Durchlaufe die Liste für alle Elemente  
    for i in range(n-1):  
        # Vergleiche alle Elemente miteinander  
        for j in range(0, n-i-1):  
            # Tausche ggf. die Plätze  
            if list[j] > list[j+1] :  
                list[j], list[j+1] = list[j+1], list[j]
```

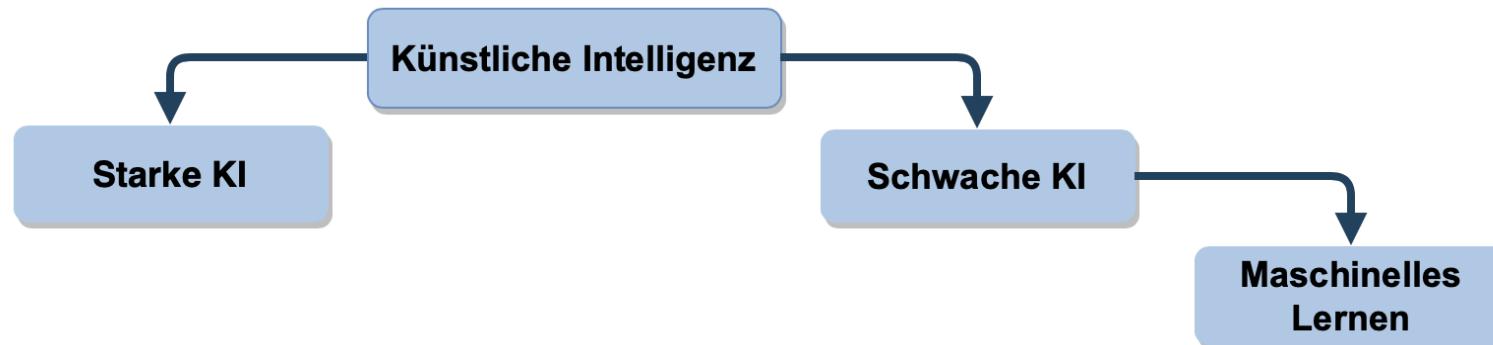
Datengetriebene Systeme

- Anforderung: Hunde von Katzen unterscheiden
 - Wie lauten wohl die Regeln?
- Wir brauchen KI/ML immer dann, wenn wir die Regeln nicht selbst definieren können, aber glauben, dass es welche gibt
- KI kann Regeln/Muster in Daten erkennen, die Menschen nicht beschreiben können
 - Wir sagen auch: Diese Systeme sind *datengetrieben*
 - Aber: Wir brauchen dafür sehr viele Daten
- Folglich wird eine KI nicht *programmiert*, sondern ermittelt selbstständig die Verteilung von Eigenschaften in einer Menge von Daten
 - Das nennen wir *lernen*



Bildquelle: freepik.com

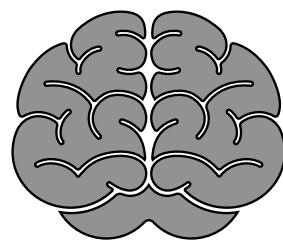
Starke und schwache KI



Starke und schwache KI

Starke KI ...

- versucht, menschliches Schlussfolgern zu imitieren
- ist ein **universeller Problemlöser, der automatisch Strategien zur Lösung neuer Probleme entwickelt**
- existiert bislang nicht



Schwache KI ...

- erkennt Muster in Daten
- erzeugt Struktur in großen Datenmengen
- ermöglicht Vorhersagen für neue Daten
- **löst eine klar definierte Aufgabe**



Einordnung: Machine Learning ist eine Menge an Werkzeugen (Algorithmen), die schwache KI umsetzen.



Machine Learning ist Mustererkennung



Hund

Katze

- Hinweis: Wir nutzen die Begriffe KI und ML synonym, auch wenn, wie wir hier sehen, es einen qualitativen Unterschied gibt.

Eine eher allgemeine Definition:

„[Maschinelles Lernen ist das] Fachgebiet, das Computern die Fähigkeit zu lernen verleiht, ohne explizit programmiert zu werden.“

— Frei übersetzt nach Arthur Samuel, 1959

Eine eher technische Definition:

„Wir sagen ein Computerprogramm lernt aus der Erfahrung E bezüglich einer Aufgabe A und der Performance-Metrik P , wenn sich seine Performance in A , gemessen an P , mit mehr E verbessert.“

— Frei übersetzt nach Tom Mitchell, 1997

- **Aufgabe** (Aufgabe A): das zu lösende Problem

- **Daten** (Erfahrung E bzgl. Aufgabe A):

- Datensatz $X = (x^{(1)}, \dots, x^{(m)})^T = (x_{ij})_{i=1, \dots, m; j=1, \dots, n} \in \mathbb{R}^{m \times n}$ (Tabelle/Matrix), der aus ursprünglich erhobenen Daten durch eine Transformation in einen geeigneten *Feature Space* (Merkmalsraum) entsteht
- Der Vektor $x_j \in \mathbb{R}^n$ bezeichne das j -te *Feature* (Merkmal, Variable) und der Vektor $x^{(i)} \in \mathbb{R}^m$ sei i -te *Sample* (Beobachtung, Datenpunkt, Beispiel), $i \in \{1, \dots, m\}$, $j \in \{1, \dots, n\}$
- Zeilen $(x^{(1)})^T, \dots, (x^{(m)})^T \in \mathbb{R}^{1 \times n}$ von X repräsentieren somit Samples und Spalten $x_1, \dots, x_n \in \mathbb{R}^n$ repräsentieren Features
- $x_{ij} = x_j^{(i)}$ sei die Ausprägung des Feature x_j des Sample $x^{(i)}$, $i \in \{1, \dots, m\}$, $j \in \{1, \dots, n\}$; da jeder Eintrag von $x^{(i)}$ ein anderes Feature beschreibt, wird $x^{(i)}$ auch als *Feature Vector* (Merkmalsvektor) bezeichnet
- X besteht somit aus m Samples, die jeweils durch n Features charakterisiert sind

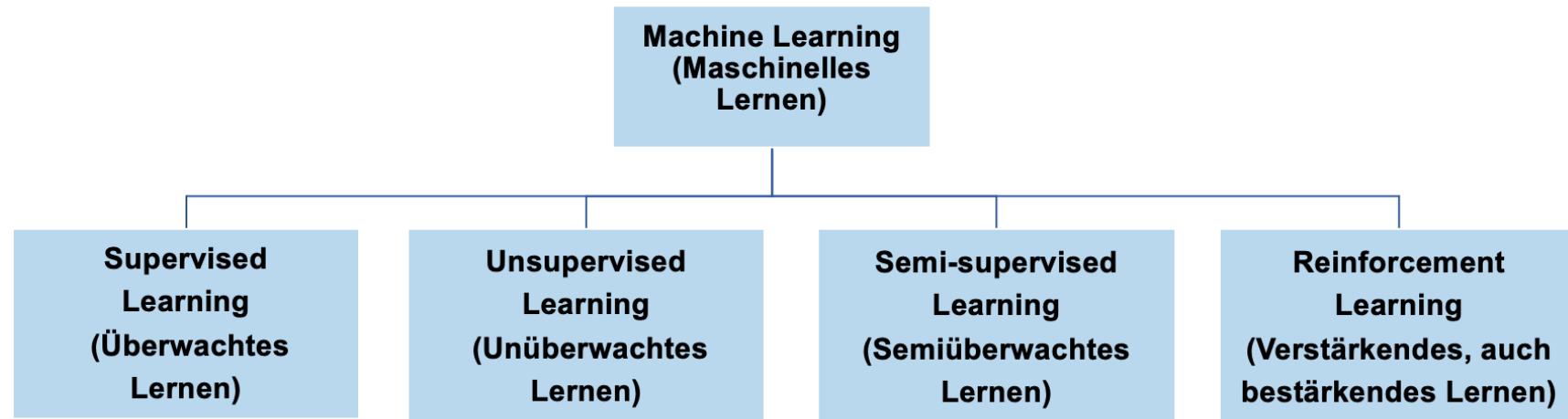
- **Algorithmus** für das Trainieren eines ML-Modells

- Modellierung: Mathematisches Modell entsprechend der gestellten Aufgabe
- *Loss Function* (Kostenfunktion, Bewertungsfunktion, Performance-Metrik P): wie gut bewältigt das aktuelle ML-Modell die Aufgabe A? Die Wahl von P ist abhängig von der Aufgabe A
- Optimierungsvorschrift: Vorschrift, wie die Modellparameter abhängig vom Ergebnis der Kostenfunktion angepasst werden müssen

Kategorien des ML

Kategorien des ML

► Kategorien des ML

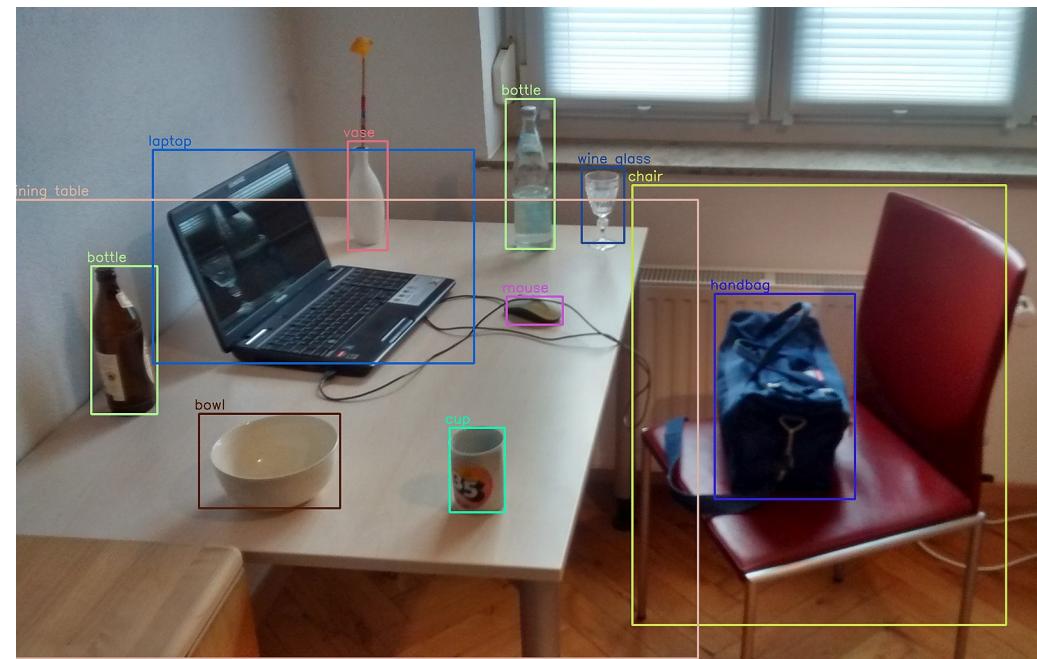


Supervised Learning

Überwachtes Lernen

Supervised Learning

- Beispiel: Objekterkennung
- KI lernt auf einer Menge korrekt markierter Beispiele (Trainingsdaten mit Label).
 - Beispiele werden oft von Menschen (z. B. Experten auf dem Gebiet) separat korrekt markiert (*gelabelt*)
 - KI lernt, die Informationen aus den Trainingsdaten zu imitieren
- Typische Beispiele:
 - Klassifikation
 - Regression
- Viele am Markt erhältliche KI-Lösungen sind dem überwachten Lernen zuzuordnen

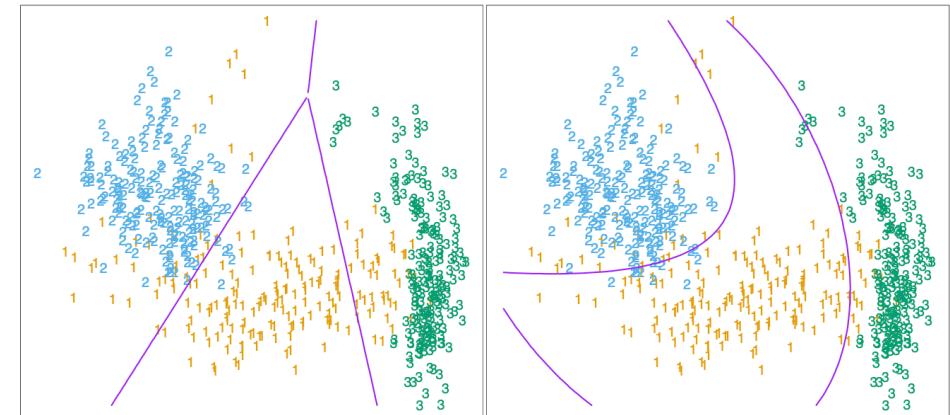


Objects detected with OpenCV's Deep Neural Network module, by M. Theiler, CC BY-SA 4.0.

Supervised Learning: Classification

- ▶ Kategorien des ML
- ▶ Überwachtes Lernen

- Gesucht: $f: \mathbb{R}^n \rightarrow \{1, \dots, k\}, y = f(x; \theta)$
- Die Zielfunktion f ordnet jeder Eingabe eine der k Klassen zu
- Für jede Klasse vergibt das Modell eine Teilwahrscheinlichkeit, mit der die (unbekannte) Eingabe zu der entsprechenden Klasse gehört
- Beispiele: Objekterkennung, Spamfilter, Bewertung gutartiger/bösartiger Hautveränderungen



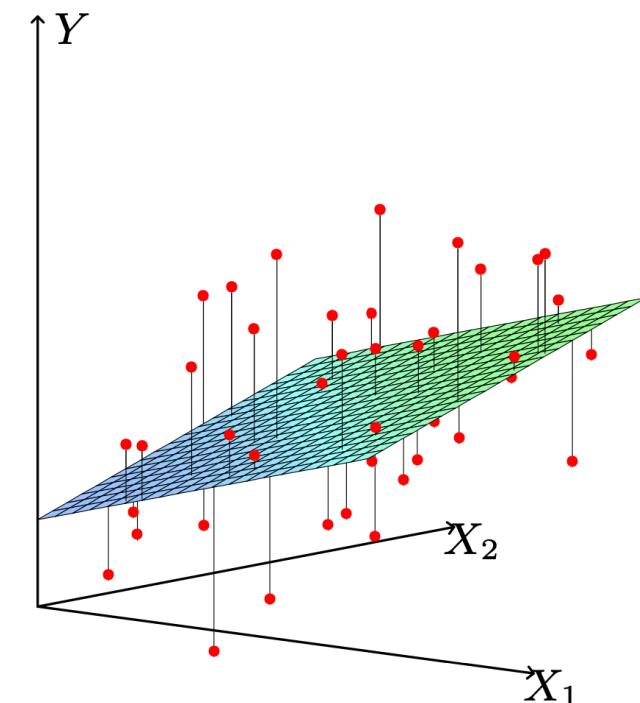
Bildquelle: Hastie T., Tibshirani R., Friedmann J. 2009. The Elements of Statistical Learning. Springer.

Supervised Learning: Regression

- ▶ Kategorien des ML
- ▶ Überwachtes Lernen



- Gesucht: $f: \mathbb{R}^n \rightarrow \mathbb{R}$, $y = f(x; \theta)$
- Prognose eines numerischen Wertes
- Starke Ähnlichkeit zur Klassifikation, jedoch mit Unterschieden im Detail
- Zeitreihen bringen besondere Eigenschaften und Anforderungen an die Modellierung und Evaluierung mit
- Beispiele: Prognose der Schadensumme bei einer Versicherung, Absatzprognose im Einzelhandel



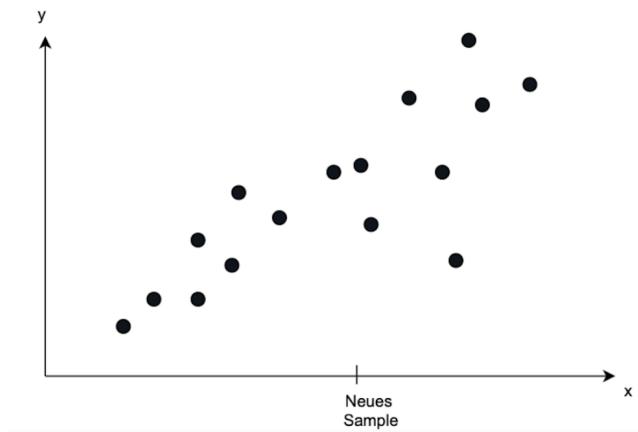
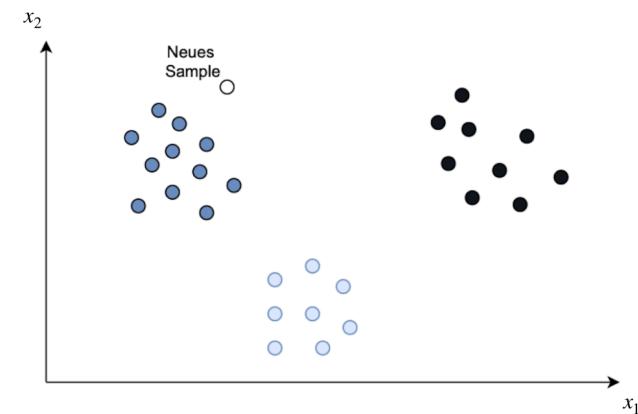
Bildquelle: Hastie T., Tibshirani R., Friedmann J. 2009. The Elements of Statistical Learning. Springer.

Supervised Learning

- ▶ Kategorien des ML
- ▶ Überwachtes Lernen



- Jedes Sample $x^{(i)}$ hat ein Label $y_i \in \mathcal{Y}$, Datenpunkte sind somit Tupel $(x^{(1)}, y_1), \dots, (x^{(m)}, y_m)$
- Überwachte Lernalgorithmen lernen aus Daten durch den Vergleich der Vorhersage mit dem korrekten Label - das Lernen wird *überwacht*, weil in den Trainingsdaten durch das Label die Ausgabe (für die Trainingsdaten) vorgegeben wird
- Ziel: möglichst treffende Vorhersagen von Labels für neue Samples durch das trainierte ML-Modell
- Gesucht: $f: \mathbb{R}^n \rightarrow \mathcal{Y}, y = f(x; \theta)$
 - f ist die unbekannte Zielfunktion, die einem Feature Vector x ein Label y zuordnet
 - $\theta = (\theta_0, \dots, \theta_l)^T$: $\theta_0, \dots, \theta_l$ sind die Parameter der Zielfunktion f , die gelernt werden
 - Beispiel: ein Polynom dritten Grades in 2D
 $f(x; \theta) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3$ mit den Parametern $\theta_0, \theta_1, \theta_2, \theta_3 \in \mathbb{R}$

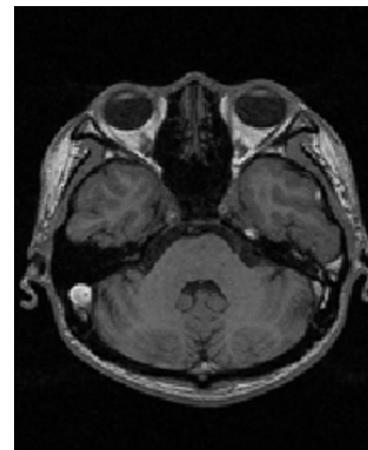


Unsupervised Learning

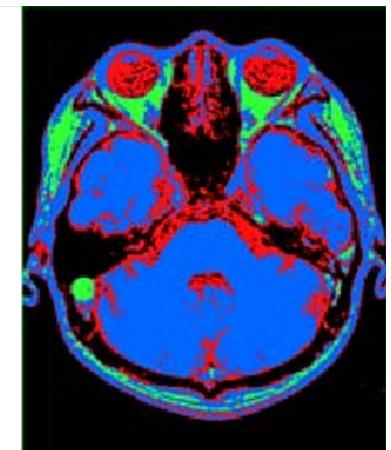
Unüberwachtes Lernen

Unsupervised Learning

- Samples im Trainingsdatensatz haben keine Labels, Datenpunkte sind somit $x^{(1)}, \dots, x^{(m)}$
- Arten der Aufgaben, wobei sich diese überschneiden
 - Clustering (Gruppierung ähnlicher Datenpunkte)
 - Dimensionality Reduction (Dimensionsreduktion)
 - Outlier Detection (Anomalieerkennung)
 - Recommender (Empfehlungssysteme)



(a)



(b)

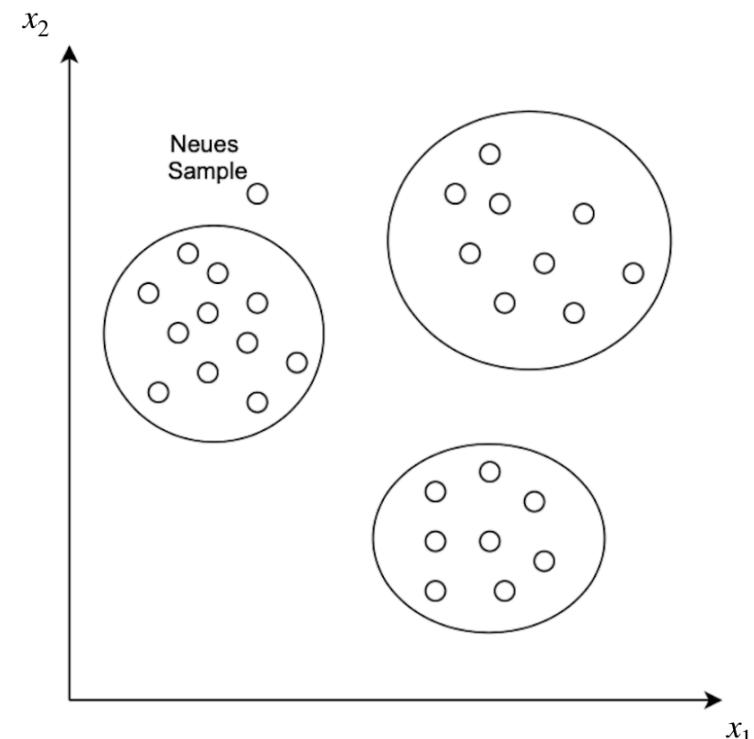
H. P. Ng, S. H. Ong, K. W. C. Foong, P. S. Goh and W. L. Nowinski, "Medical Image Segmentation Using K-Means Clustering and Improved Watershed Algorithm," *2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, Denver, CO, 2006, pp. 61-65.

Unsupervised Learning: Clustering

► Kategorien des ML
► Unüberwachtes Lernen



- Ziel: Struktur in eine Menge von Datenpunkten bringen, ähnliche Punkte zu Clustern zusammenfassen, neue Datenpunkte einem der Cluster zuordnen können
- Unüberwachte Lernalgorithmen verwenden Distanzmetriken und lokale Dichte im Feature Space, um ähnliche Punkte zu entdecken
- Unüberwachte Lernalgorithmen zum Erstellen von Clustern unterscheiden sich in
 - der Definition von Ähnlichkeit
 - der Definition von Clustern
 - dem algorithmischen Vorgehen
- Beispiele: Kunden eines Onlineshops zu Gruppen zusammenfassen, Bildsegmentierung, Identifikation von repräsentativen Samples

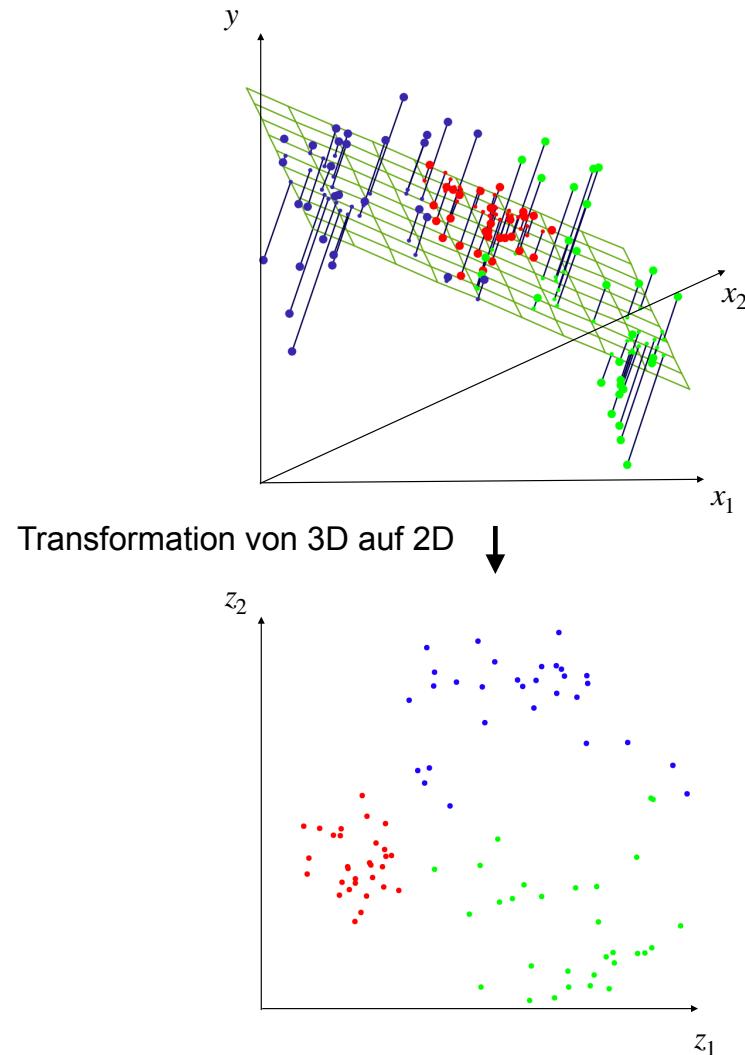


Unsupervised Learning: Dimensionality Reduction

► Kategorien des ML
► Unüberwachtes Lernen



- Ziel: Auffinden von versteckten Repräsentationen von hochdimensionalen Daten in einem weniger dimensionalen Raum, sodass die ursprünglichen Informationen und Struktur möglichst bewahrt werden
- Unterscheidung zwischen
 - Feature Selection: Auswahl aussagekräftiger Features aus allen Features
 - Feature Extraction: Reduzieren der Dimension durch Gewinn neuer aus vorhandenen Features
- Ist ein Feature für jeden Datenpunkt gleich, so kann sich daraus keine zusätzliche Information ergeben, die nicht ohne dieses Feature vorhanden war
- Beispiele: Visualisierung als ein Teilaufgabengebiet, um eine 2D- oder 3D-Repräsentation komplexer Daten ohne Labels zu ermöglichen



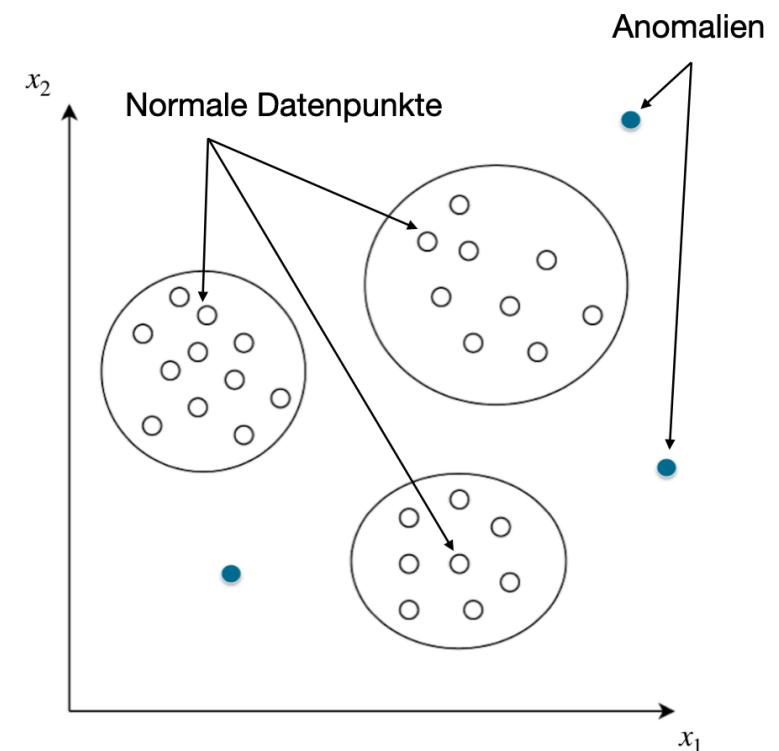
Bildquelle: Hastie T., Tibshirani R., Friedmann J. 2009. The Elements of Statistical Learning. Springer.

Unsupervised Learning: Outlier Detection

► Kategorien des ML
► Unüberwachtes Lernen



- Ziel: Identifikation ungewöhnlicher Datenpunkte
- In der Regel liegen bekannte Anomalien nicht in ausreichender Menge vor
- Hohe Komplexität im Aufbau des Feature Space, da Modelle nicht zwischen relevanten und irrelevanten Features unterscheiden können
- Beispiele: Erkennen von Kreditkarten- oder Versicherungsbetrug, Erkennen von Einbrüchen in Computer-Netzwerke



Unsupervised Learning: Recommender

► Kategorien des ML
► Unüberwachtes Lernen



- Ziel: Auffinden von Beziehungen zwischen Features in einer großen Datenmenge
- Anwendung häufig bei binären Features: alle Werte der Features (und damit der Matrix X) sind 0 oder 1,
 $x_{ij} \in \{0,1\} \quad \forall i \in 1, \dots, m, \forall j \in 1, \dots, n$
- Beispiel: Filmempfehlung, mit $x_{ij} = 1$ (Film geschaut) und $x_{ij} = 0$ (Film nicht geschaut)
- Ziele: feststellen, wer welche Filme schaut und daraus Regeln für Empfehlung unbekannter Filme ableiten
- Vereinfacht: „Wenn Film A und Film B gesehen wurden, dann wird auch Film C geschaut: $\{A, B\} \rightarrow \{C\}$ “ ohne dass die umgekehrte Richtung gilt

	Film A	Film B	Film C	Film D
Person 1	✓	✗	✓	✓
Person 2		✓	✗	✗
Person 3	✓	✓	✗	
Person 4	✗		✓	
Person 5	✓	✓	?	✗

Semi-Supervised Learning

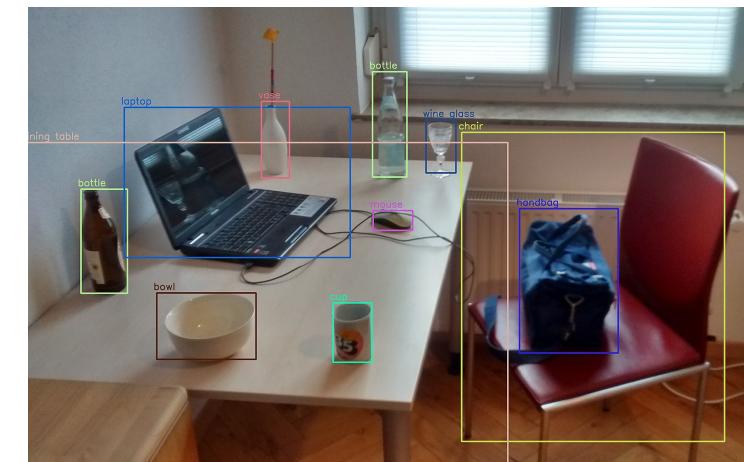
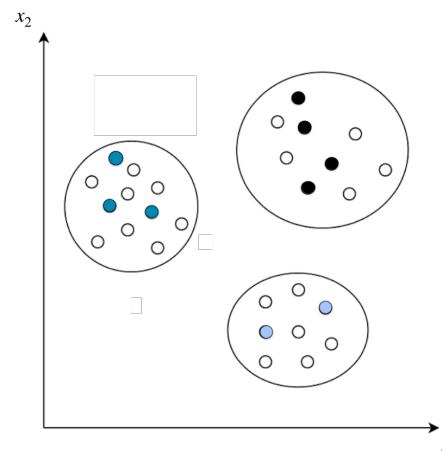
Semiüberwachte Verfahren

Semi-Supervised Learning

► Kategorien des ML
► Semiüberwachtes Lernen



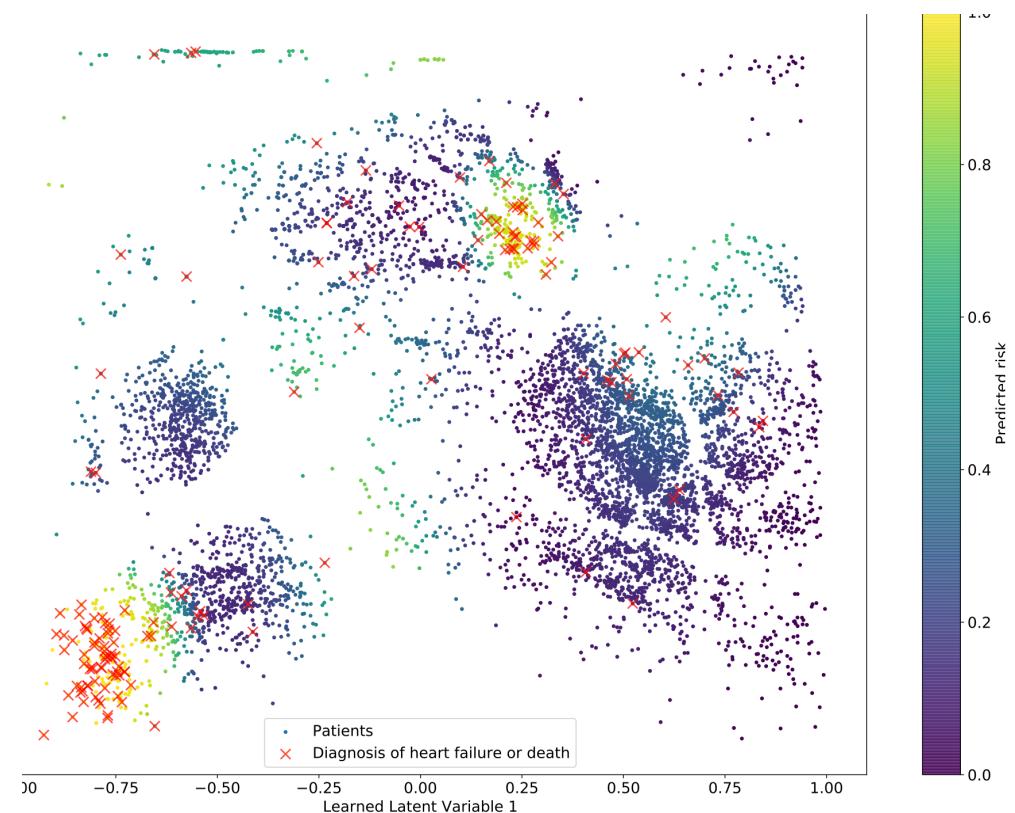
- Nur ein kleiner Teil der Datenpunkte ist gelabelt
- Ziel: den Datenpunkten ohne Label ein Label zuweisen können
- Die meisten Algorithmen sind Kombinationen von überwachten und unüberwachten Verfahren
- Eine verbreitete Vorgehensweise ist, die Datenpunkte zu clustern, und innerhalb der Cluster erhalten Datenpunkte ohne Label dasselbe Label, das die gelabelten Datenpunkte im Cluster aufweisen
- Beispiel: Optimierung der Objekterkennung



Objects detected with OpenCV's Deep Neural Network module. by M. Theiler, CC BY-SA 4.0.

Semi-Supervised Learning

- Beispiel: Automatische Bewertung des Risikos für kardiovaskuläre Ereignisse
- Daten sind gelabelt, aber es gibt verschiedene Probleme:
 - Viele Informationen über einzelne Patienten (>300 Attribute)
 - Wenige Patienten
 - Sehr seltes Auftreten des Ereignisses
- Lösung: Kombination aus unüberwachtem und überwachtem Verfahren
 - 1. unüberwachte Reduktion der Dimensionen
 - 2. überwachte Klassifikation der Daten



Reinforcement Learning

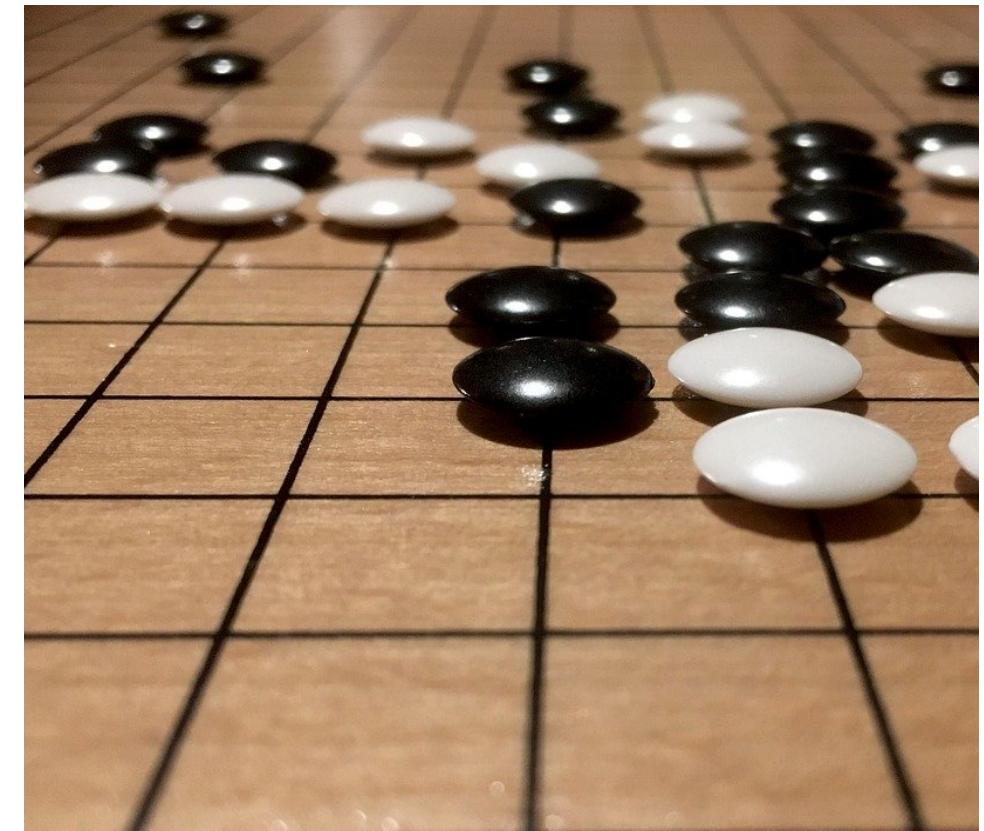
Verstärkendes Lernen

Reinforcement Learning

- ▶ Kategorien des ML
- ▶ Bestärkendes Lernen



- Ziel: Erlernen optimaler Strategien
- Beispiel: Weltmeister im Go werden
- KI lernt in einer Simulationsumgebung durch *Try and Error* und ein Belohnungssystem eine Strategie zur Lösung eines Problems
 - KI soll Strategien finden, die Menschen verborgen bleiben
- Typische Beispiele
 - Spiele aller Art (Schach, StarCraft, ...)
 - Individualisierung von Empfehlungen
 - Platzierung von Werbung

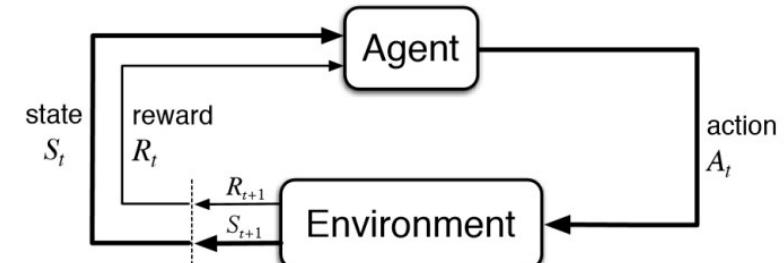


Reinforcement Learning

- ▶ Kategorien des ML
- ▶ Bestärkendes Lernen

- KI, hier als *Agent* bezeichnet, das in einem *Environment* (Umgebung) interagiert und *Actions* (Aktionen) ausführt
- Ziel: Lernen einer mittel-/langfristigen *Policy* (Strategie) zur Maximierung eines zukünftig zu erwartenden kumulativen *Rewards* (Belohnung)
 - Positiver Reward bei guten Actions
 - Negativer Reward (Bestrafung) bei schlechten
- Eine Policy bestimmt, welche Aktion als nächstes ausgeführt wird
- Agent kann zeitliche Aspekte berücksichtigen
 - Kurzfristig schlechte Aktionen können langfristig zu einem höheren Reward führen und umgekehrt

- Setting: Zu jedem Zeitpunkt t
 - Agent beobachtet Environment S_t
 - Agent führt Action A_t aus
 - Environment ändert sich zu S_{t+1}
 - Agent erhält einen Reward R_{t+1} für den neuen Zustand S_{t+1} zur Aktion A_t



- Hinweis: Reinforcement Learning wird in dieser Veranstaltung nicht detailliert behandelt

Herausforderungen

1. Training und Testen
2. Transformation in einen geeigneten Feature Space
3. Daten
4. Hyperparameter

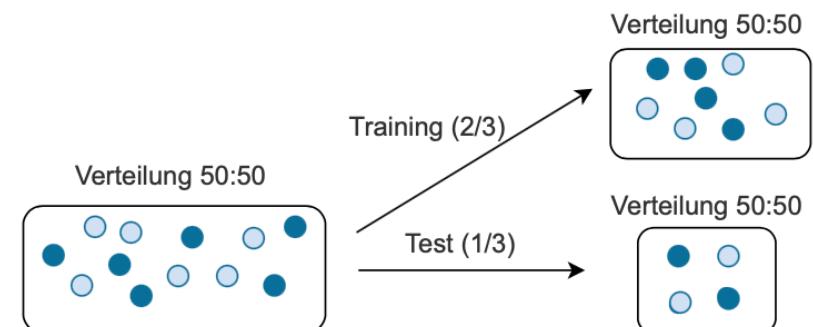
Training und Testen

► Training und Testen



- Das Ziel der Trainingsphase ist es, Modelle zu trainieren, die möglichst gut generalisieren, d.h. Modelle, die richtige Vorhersagen auf neuen Samples zu treffen
- Um Modelle zu trainieren und die Generalisierbarkeit zu überprüfen, werden die vorhandenen Samples $x^{(1)}, \dots, x^{(m)}$ in einen Trainings- und einen Testdatensatz geteilt
 - Training des Modells mit dem Trainingsdatensatz, Anwendung des Modells auf den Testdatensatz zur Prüfung der Güte des Modells
 - Werden Anpassungen am Modell vorgenommen, wird immer mit demselben Trainingsdatensatz trainiert
 - Übliche Aufteilung in einen Trainings- und einen Testdatensatz: 70%:30% oder 80%:20%

- Möglichkeiten der Aufteilung
 - Zufällig
 - Manuell
 - Stratifizierte Stichprobe (repräsentative Stichprobe): sowohl der Trainings- als auch der Testdatensatz weisen dieselbe Verteilung bestimmter Features auf



- Die erhobenen Daten können beliebiger Art sein: Tabelle mit verschiedenen skalierten (nominal, ordinal, kardinal, zum Skalenniveau siehe Anhang) Features, Bilder, Messwerte, Text, natürliche Sprache, Signalamplitude, ...
- Algorithmen können nur mit numerischen Daten umgehen
 - Die ursprünglich erhobenen Daten müssen in einen geeigneten Feature Space transformiert werden, um diese analysieren und die Aufgabe lösen zu können
 - Nach Transformation entsteht ein Datensatz X , sodass die ursprünglich erhobenen Daten für die ML Algorithmen irrelevant sind
- Fragen und Probleme, die bei der Transformation auftauchen (können)
 - Welche Features können/sollen aus den erhobenen Daten extrahiert werden?
 - Abhängig von ursprünglichen Daten und der Art der Transformation kann die Anzahl der Features im transformierten Datensatz X sehr hoch werden → Dimensionsreduktion kann notwendig sein

Beispiel einer möglichen Transformation: One-Hot Encoding für kategoriale Features

► Herausforderungen



Extraktion
von
Features

Durchmesser	Farbe	Gewicht	Label
18 cm	rot	190 gr	Apfel
16 cm	Grün	250 gr	Birne
...

$$\text{Obststück 1} = \begin{pmatrix} 18\text{cm} \\ \text{rot} \\ 190\text{gr} \\ \text{Apfel} \end{pmatrix}, \text{ Obststück 2} = \begin{pmatrix} 16\text{cm} \\ \text{grün} \\ 250\text{gr} \\ \text{Birne} \end{pmatrix}$$

$X =$

Durchmesser	Rot?	Grün?	Gewicht	Apfel?
18	1	0	190	1
16	0	1	250	0
...

$$x^{(1)} = \begin{pmatrix} 18 \\ 1 \\ 0 \\ 190 \\ 1 \end{pmatrix}, x^{(2)} = \begin{pmatrix} 16 \\ 0 \\ 1 \\ 250 \\ 0 \end{pmatrix}$$

- Der Einsatz von KI ist in der Vergangenheit auch negativ aufgefallen. Häufiges Problem: Die Datengrundlage passt nicht zum Anwendungszweck!
 - Daten müssen auf ihre fachliche Eignung hin geprüft werden!
 - KI-Entwicklung ist (meistens) interdisziplinär
- Unzureichende Menge an Daten
 - Auch der „mächtigste“ Algorithmus kann versagen
 - Bei einer ausreichend großen Menge an Daten liefern verschiedene Algorithmen etwa gleich gute Ergebnisse
- Nicht repräsentative Daten: keine Verallgemeinerung möglich, unzutreffende Vorhersagen
 - Minderwertige Daten
 - Starkes Rauschen
 - Hoher Anteil an Ausreißern/Anomalien
 - Fehlende Werte
 - Unpassende Verteilung
 - (Zu) Viele irrelevante Features
 - PROBLEM: Müll rein, Müll raus!
 - Wichtig: Zeit investieren und explizit einplanen, um brauchbare Daten zu sammeln und diese gründlich aufzubereiten!

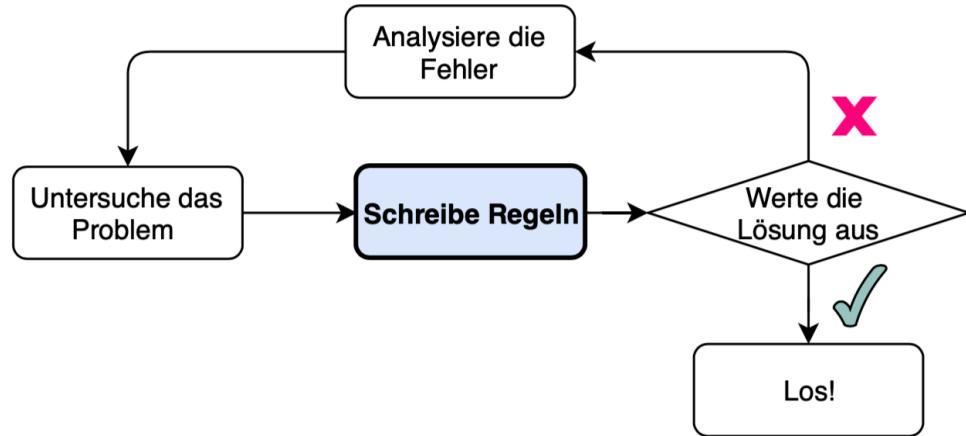
- Parameter des Algorithmus, nicht des Modells (*Modellparameter*)
- Diese werden nicht gelernt, sondern müssen **vor dem Training** gesetzt werden und bleiben während des Trainings konstant
- Die meisten Algorithmen besitzen mindestens einen Hyperparameter
- Hyperparameter sind für das resultierende Modell ausschlaggebend
- Wahl von Hyperparametern ist oft nicht trivial, meist ist Tuning notwendig
- Beispiele: Modellkapazität wie Grad bei polynomialer Regression, Regularisierungsparameter, Modellarchitektur (bei Entscheidungsbäumen u.a. die Tiefe, bei neuronalen Netzen u.a. Anzahl von Neuronenschichten und Anzahl von Neuronen in jeder Schicht)

Anhang

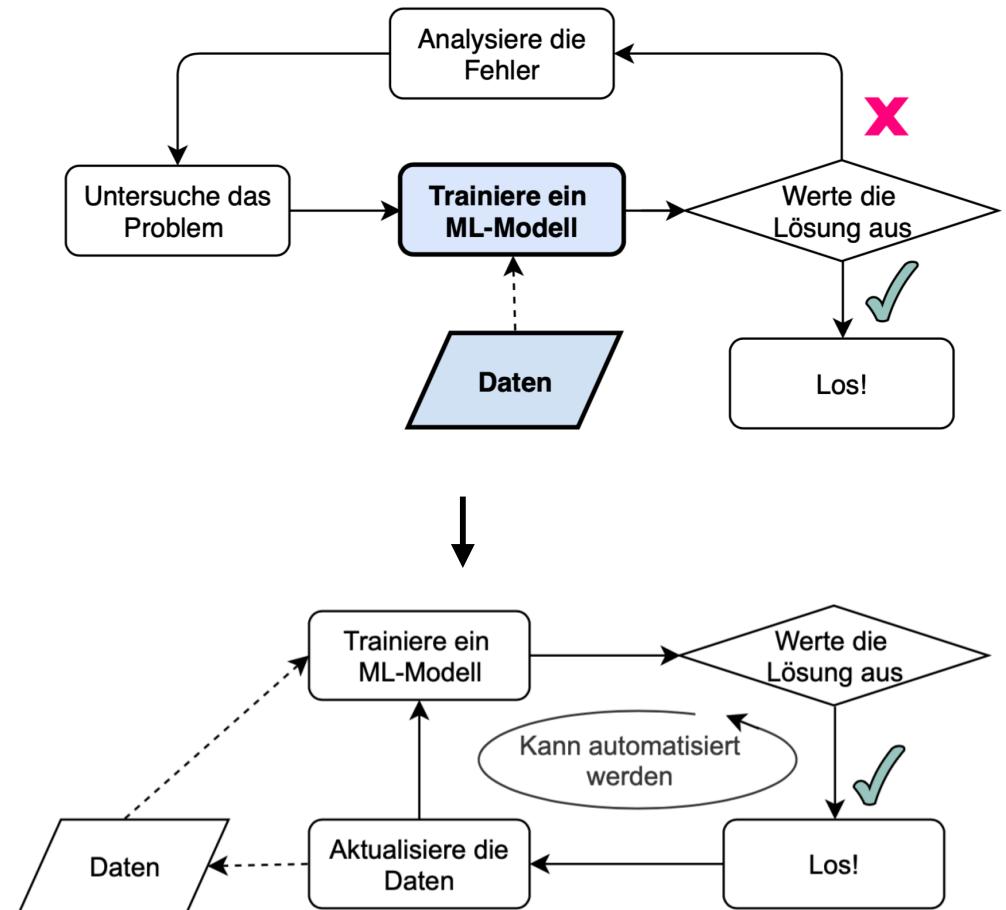
Traditioneller vs. ML Ansatz

- ▶ KI und ML
- ▶ Definition, Einordnung, Setting

Traditioneller Ansatz



ML Ansatz

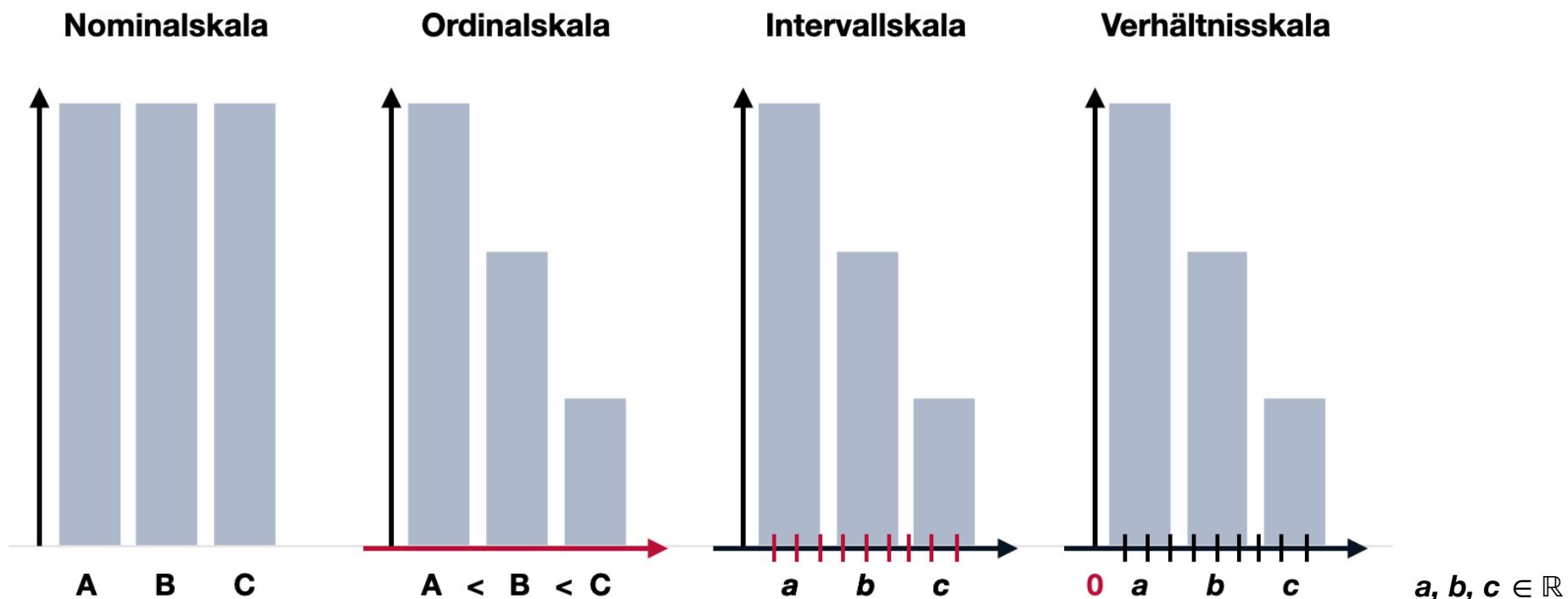


Quelle: In Anlehnung an Gerón, A. 2017. Hands-On Machine Learning with Scikit-Learn and TensorFlow. O'Reilly.

- Nominalskala (hat das niedrigste Skalenniveau)
 - Ausprägungen der Variable (des Feature) haben keine Rangordnung, es handelt sich um qualitative Variablen
 - *Beispiel Haarfarbe* (dies ist die Variable, deren Werte=Ausprägungen „blond“, „braun“, „schwarz“ etc. sind): Man kann lediglich unterscheiden, ob die Haarfarbe zweier Personen gleich oder unterschiedlich ist
- Ordinalskala:
 - Ausprägungen der Variable haben eine Rangordnung, aber Abstände können nicht interpretiert werden
 - *Beispiel Sporttabelle*: Team auf Platz 1 ist besser als das auf Platz 3, aber nicht drei mal so gut
- Kardinalskala mit Unterscheidung zwischen:
 - Intervallskala: Ausprägungen der Variable haben eine Reihenfolge, Abstände können interpretiert werden und eine Differenz- und Summenbildung ist möglich, es besteht Gleichheit der Abständen, jedoch ist keine Multiplikation möglich, da kein natürlicher Nullpunkt existiert (d.h. der Nullpunkt wurde willkürlich gesetzt)
 - *Beispiel Intelligenzquotient*: der Abstand zwischen den IQ-Werten 70 und 75 ist genauso groß wie zwischen 91 und 96, die Aussage aber, dass eine Person mit einem IQ von 130 doppelt so intelligent ist wie eine mit einem IQ von 65 macht keinen Sinn
 - Verhältnisskala (auch Rationsskala, hat das höchste Skalenniveau): zusätzlich zu den Eigenschaften der Intervallskala gibt es einen natürlichen Nullpunkt, sodass Multiplikation und Division möglich sind und das Verhältnis zweier Ausprägungen zueinander definiert und berechenbar ist
 - *Beispiel Preise*: Artikel A kostet 10 EUR und Artikel B 5 EUR, daher ist Artikel A doppelt so teuer
- Nominal- und ordinalskalierte Variablen werden auch als kategoriale Variablen und kardinalskalierte Variablen auch als numerische Variablen bezeichnet

Zur Erinnerung: Skalenniveau

► Anhang



Quelle: In Anlehnung an Wikipedia