Maschinelles Lernen

Praktischer Kurs

Vorlesung (jede Woche), und Labor (jede 2. Woche): Victor Șolea

>7 Jahre Erfahrung als Data Scientist im Privatsektor

Kontakt: victor.solea@ubbcluj.ro oder Teams

MS Teams Code: 55566f2

Struktur

- Keine Anwesenheitspflicht
- Beim Labor bitte mit der eigenen Gruppe kommen

- Laboraufgaben: 75% aus der Endnote
- Schriftlicher Test in der letzten Woche: 25% aus der Endnote
  - Im Intervall einer der DB-Laborstunden
- Bonuspunkte

- Bemerkung: Chatgpt macht dumm! link, link

Laboraufgaben

- Verspätung: -2p je Laborstunde
- In Teams von 2 oder 3 Studierenden, jeder wählt das Team selbst
- Man kann für jede Laboraufgabe mit Anderen arbeiten

Inhalt

1. Bearbeiten von tabellarischen Daten: missing data, outliers, kategorische Daten, exploratorische Datenanalyse
2. Klassische supervised learning Modelle (Entscheidungsbäume, k nearest neighbours, Support Vector Machines, u.a.)
3. Ensemble Learning
4. Bewertung der Modellergebnisse. Metriken. Unbalancierte Daten, Augmentation
5. Unsupervised Learning (Clustering). Dimensionalitätsreduktion
6. Neuronale Netze. Einführung in die Bilderverarbeitung

Inhalt (flexibel)

Andere mögliche Themen (abhängig von Zeit und Interesse)**:**

- Zeitreihenanalyse
- Recommender Engines
- Textanalyse
- Large Language Models
- Reinforcement learning
- Monitoring / retraining
- …

Feedback ist immer erwünscht und willkommen

Literatur

Raschka, Sebastian and Mirjalili, Vahid: Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow

Geron, Aurelien: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems

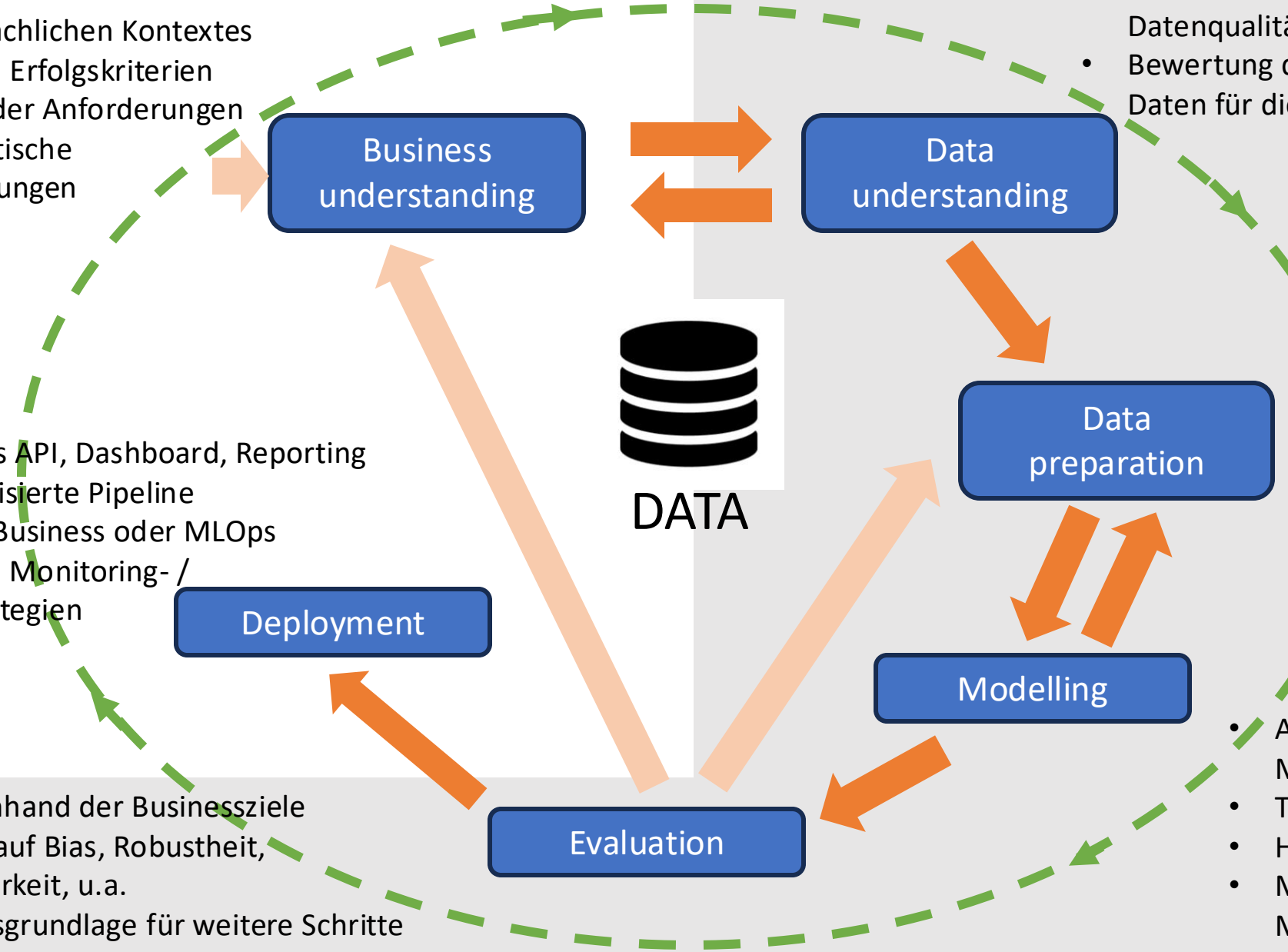
Hastie, Trevor; Tibshirani, Robert; and Friedman, Jerome: The Elements of Statistical Learning

Ng, Andrew: Machine Learning Course, Stanford University, youtube link

Die amtliche Dokumentation von den relevanten Bibliotheken (pandas, sklearn, …)

Lebenszyklus einer ML-Anwendung

- Analyse des fachlichen Kontextes
- Definition von Erfolgskriterien
- Übersetzung der Anforderungen in datenanalytische Aufgabenstellungen



- Datenakquise
- Exploratorische Datenanalyse
- Identifikation von Datenqualitätsproblemen
- Bewertung der Relevanz vorhandener Daten für die Fragestellung

- Datenbereinigung
- Feature engineering
- Auswahl relevanter Variablen
- Aufbau von Trainings- / Testdaten
- Aggregation, Transformation, Normalisierung

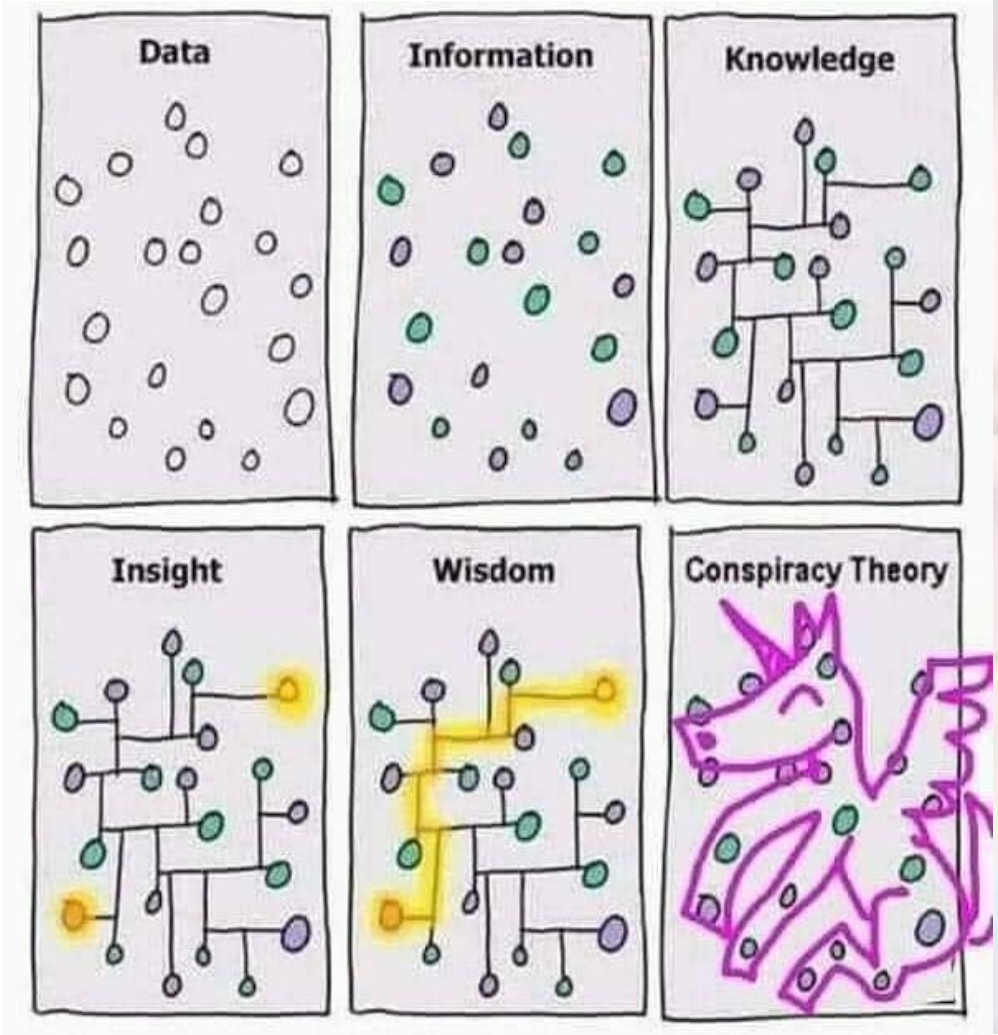
- Auswahl von Modellierungstechniken
- Training und Validieren
- Hyperparameter tuning
- Modellvergleich anhand Metriken

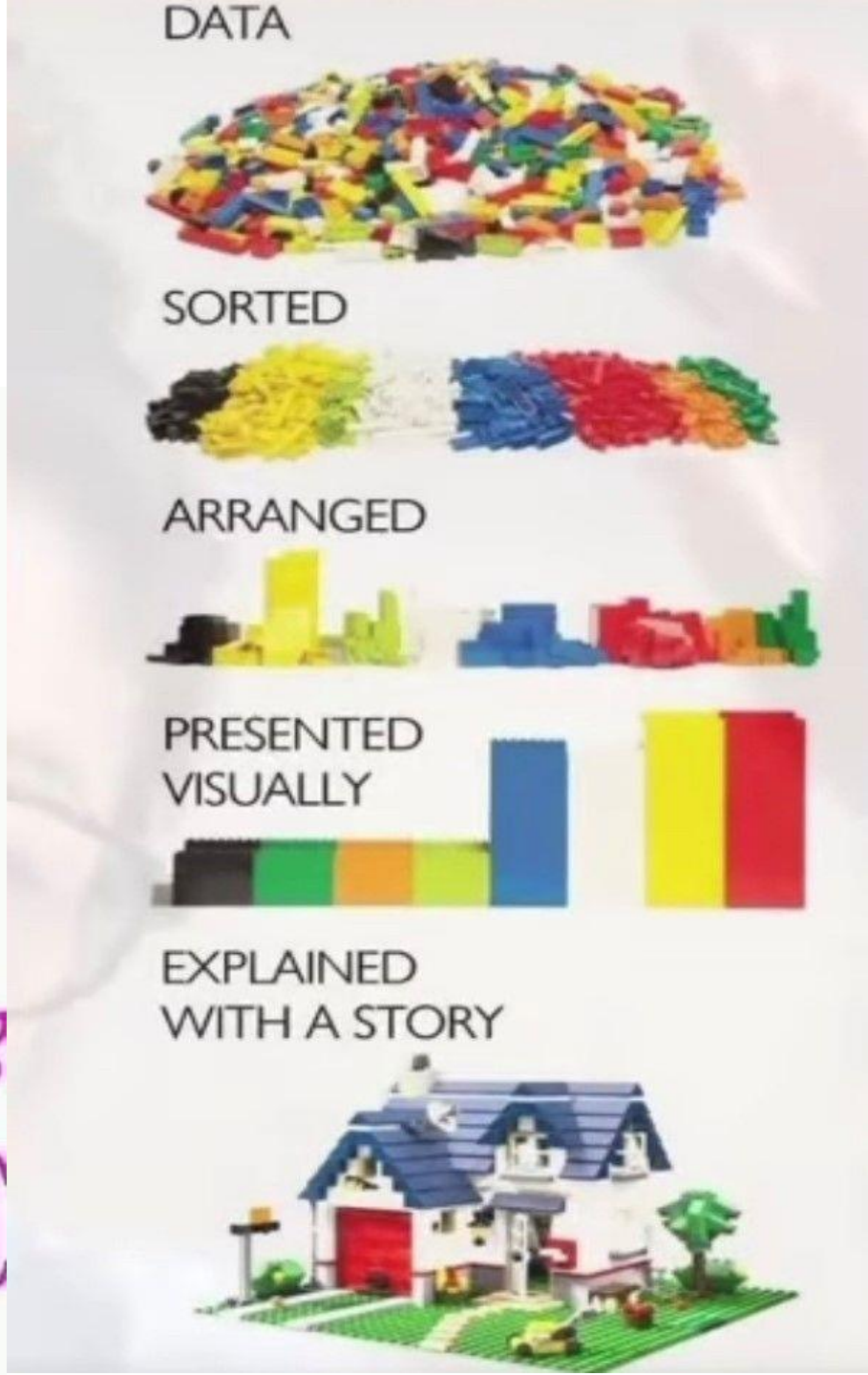
- Validierung anhand der Businessziele
- Überprüfung auf Bias, Robustheit, Interpretierbarkeit, u.a.
- Entscheidungsgrundlage für weitere Schritte

- Umsetzung als API, Dashboard, Reporting oder automatisierte Pipeline
- Übergabe an Business oder MLOps
- Definition von Monitoring- / Retrainingstrategien

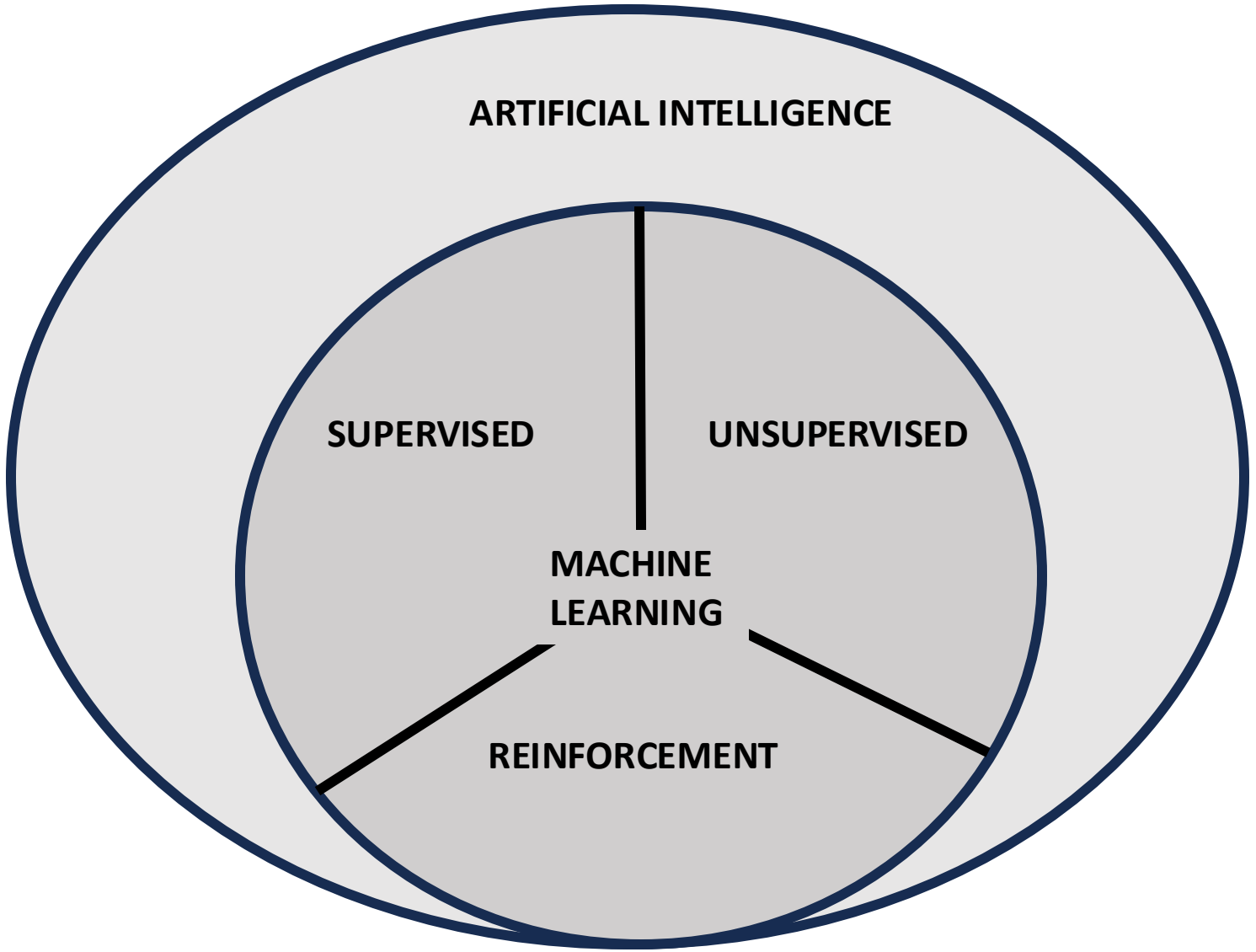
At the end of the day, a data scientist is a storyteller

Wie trifft man richtige, sinnvolle und einflussreiche Business-entscheidungen?





Definitionen, Terminologie



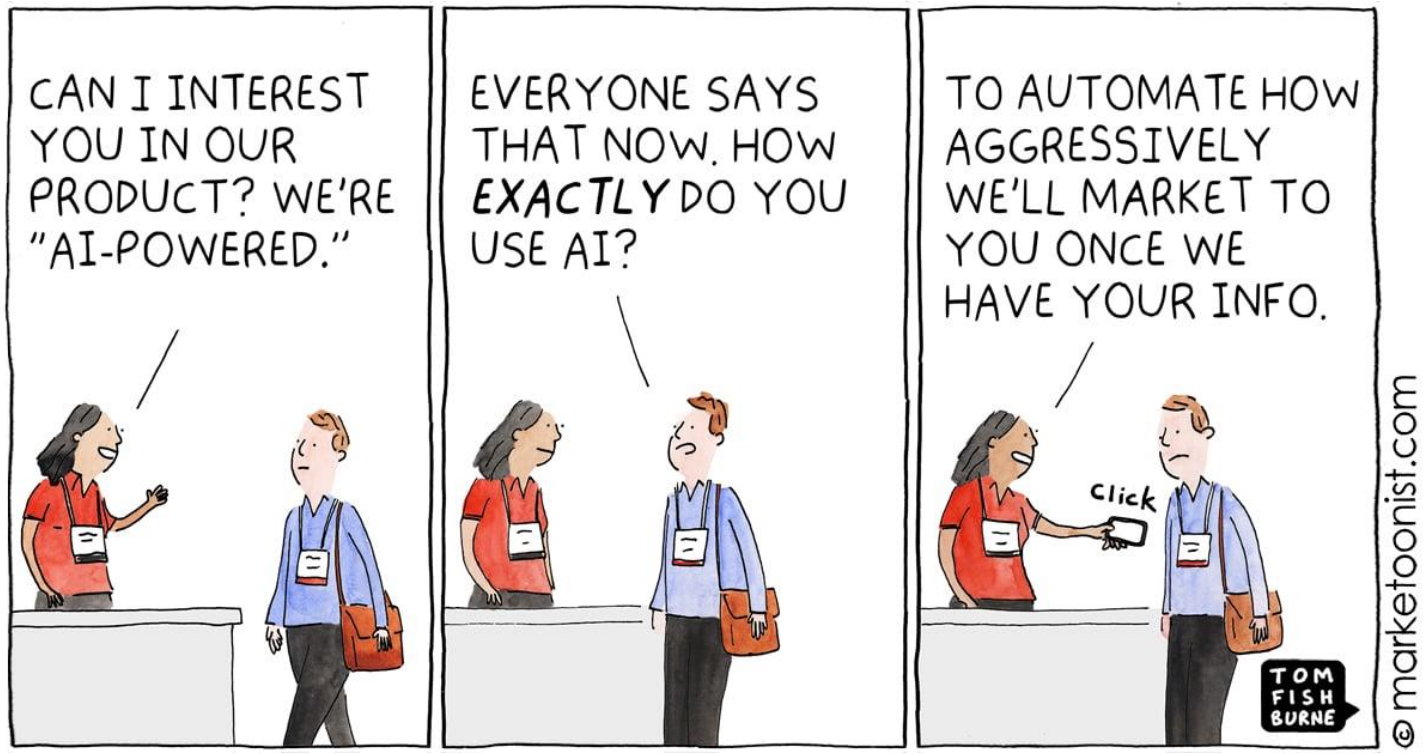
AI: die "Kunst", Maschinen zu bauen, die ähnlich dem Menshen handeln – eher ein philosophischer Konzept, eine Art *deus ex machina*

ML: eine Klasse von Algorithmen, mit Grundlagen in Statistik und linearer Algebra, welche auf Grund von bekannten Daten auf neue, ungesehene Daten verallgemeinern können

Definitionen, Terminologie

In der Praxis bedeutet Machine Learning das Lösen von **DREI** Arten von Businessaufgaben:

- Detektion
- Prädiktion
- Klassifikation



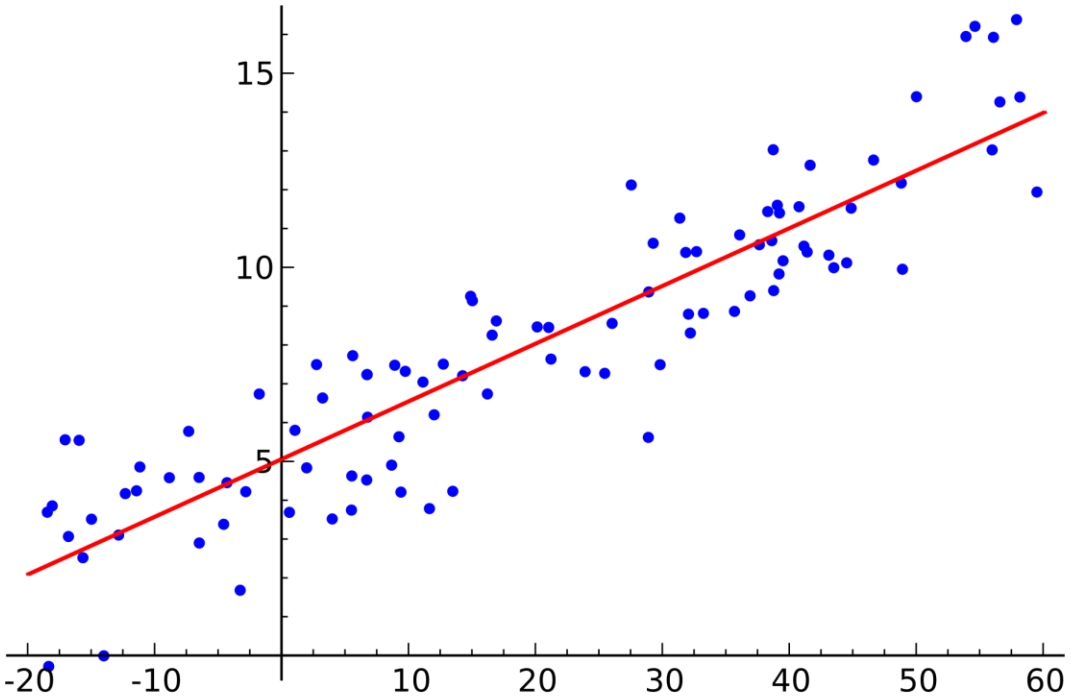
Definitionen, Terminologie

- **Algorithmus:** eine Menge von Regeln (meistens mathematische Funktionen), die es erlauben, Inferenzen aus gegebenen Daten zu ziehen
- **Feature:** Eigenschaft des Sachverhalts mit dem man arbeitet (z.B. Oberfläche einer Wohnung)
- **Hyperparameter:** Koeffizienten des Modells, welche das Trainingsprozess beeinflussen, werden vor dem Training festgelegt (z.B. random seed)
- **Klassifikation:** Aufgabe im supervised learning, wobei ein Datenpunkt einer Kategorie aus einer endlichen Menge zugeordnet wird
- **Modell:** ein trainierter Algorithmus
- **Parameter:** Variable innerhalb des Modells, deren Wert vom Training gefunden wird
- **Regression:** Prädiktion des Wertes einer kontinuierlichen numerischen Variablen auf Grund von mehreren Inputparametern
- **Supervised learning:** die Trainingsdaten sind im Voraus beschriftet, man weiß was man identifizieren möchte
- **Training:** das eigentliche Lernen, der Prozess durch welches die Parameter des Algorithmus am gegebenen Datensatz angepasst werden (in der Praxis: Minimieren / Maximieren einer mathematischen Funktion)
- **Unsupervised learning:** man hat im Voraus keine formellen Ziele und keine Belohnung

Voraussetzungen

- Programmieren in Python
- Verständnis der bisher studierten Mathematik (vor Allem lineare Algebra und Statistik)

Lineare Regression.
Praktische Erläuterung der Begriffe



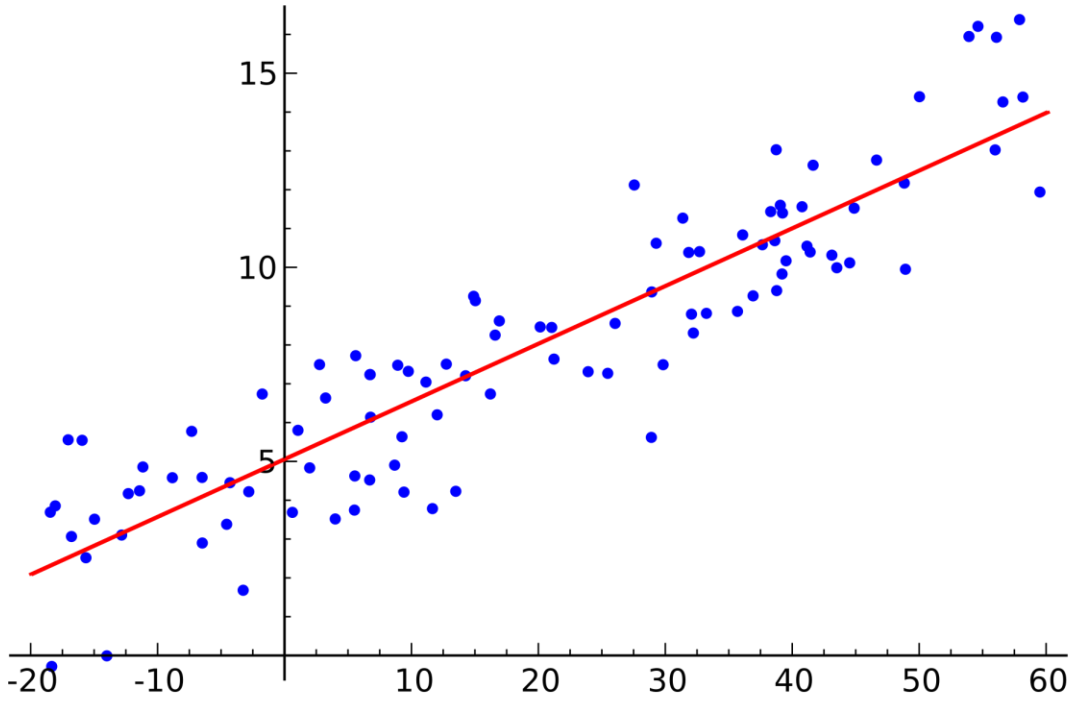
Die einfachste Form:

Gleichung der Geraden: $y = mx+b$

Für die gegebenen Punkte, finde *m* und *b* so dass die gegebene Gerade die beste Aussagekraft hat

Beim Starten des Trainings haben *m* und *b* beliebige Werte

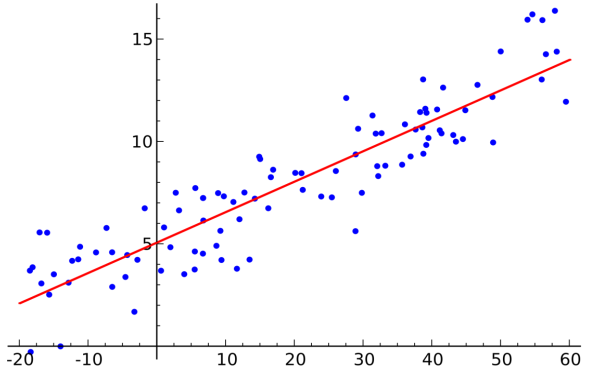
Lineare Regression.
Praktische Erläuterung der Begriffe



In diesem Kontext:

- Welches ist das Modell?
- Was bedeutet Training?
- Welche sind die Parameter?
- Ist es supervised oder unsupervised learning?
- Welche könnten Hyperparameter sein?

Lineare Regression



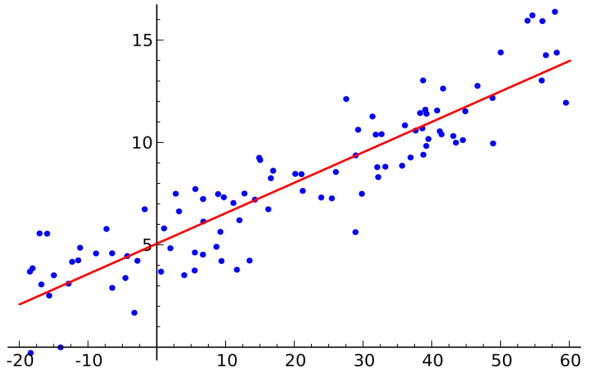
Man definiert eine sog. **Verlustfunktion (loss function)** — intuitiv, ein Maß wie weit weg der wahre Wert von der Schätzung ist

Manchmal auch "objective function" oder "cost function" genannt

Denke nach!

- Wie sollte so eine Verlustfunktion in unserem Fall aussegen?
- Welches ist der wahre, bzw. der geschätzte Wert (true value vs predicted value)?

Lineare Regression



Typischerweise: zwei mögliche Verlustfunktionen

- Mean average error (MAE): $\sum |y_{true} - y_{pred}|$

- Mean square error (MSE): $\sum (y_{true} - y_{pred})^2$

Meistens wird MSE (oder äquivalent) benutzt

Unterschied: wie empfindlich ist man gegenüber Outliers

Stoppbedingung: der Wertunterschied der Verlustfunktionen zwischen zwei aufeinanderfolgenden Iterationen ist kleiner als ein vorgegebener ε

Lineare Regression

Training bedeutet, in der Praxis, die Parameter finden, welche die Verlustfunktion $\sum (y_{true} - y_{pred})^2$ minimieren

Was für Eigenschaften hat die Verlustfunktion?

Was bedeutet, das Minimum einer Funktion finden?

Welche sind die Argumente der Verlustfunktion?

(Hinweis: denke an die mathematische Formulierung der Prädiktion)

Gradient Descent

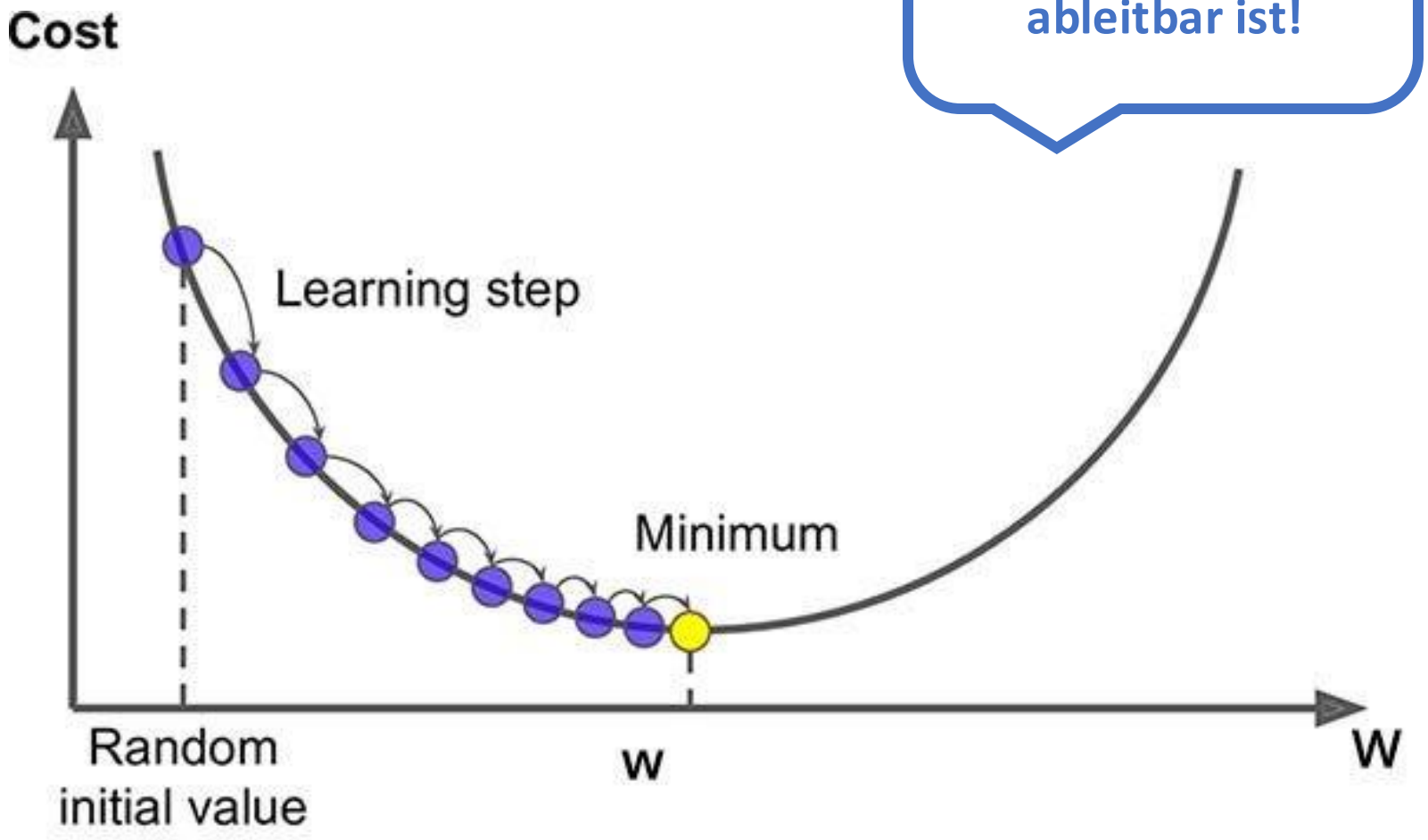
- Mit Optimierungen und Heuristiken: die Standardmethode zum Finden des Minimum- / Maximumwertes einer gegebenen Funktion
- Bei linearer Regression, neuronalen Netzen, und anderen Algorithmen benutzt
- Wir besprechen hier die "Vanilla"-Version

Gradient Descent

Funktioniert, weil die Verlustfunktion ableitbar ist!

Bemerkungen:

- Der Schritt ist meistens proportional zur Steigung der Funktion.
- In den meisten Fällen ist es eine Funktion $\mathbb{R}^n \rightarrow \mathbb{R}$
- *n* ist die Anzahl der Parameter / Features des Modells, kann sehr groß werden für komplexe Modelle



Quelle: Geron (2019)

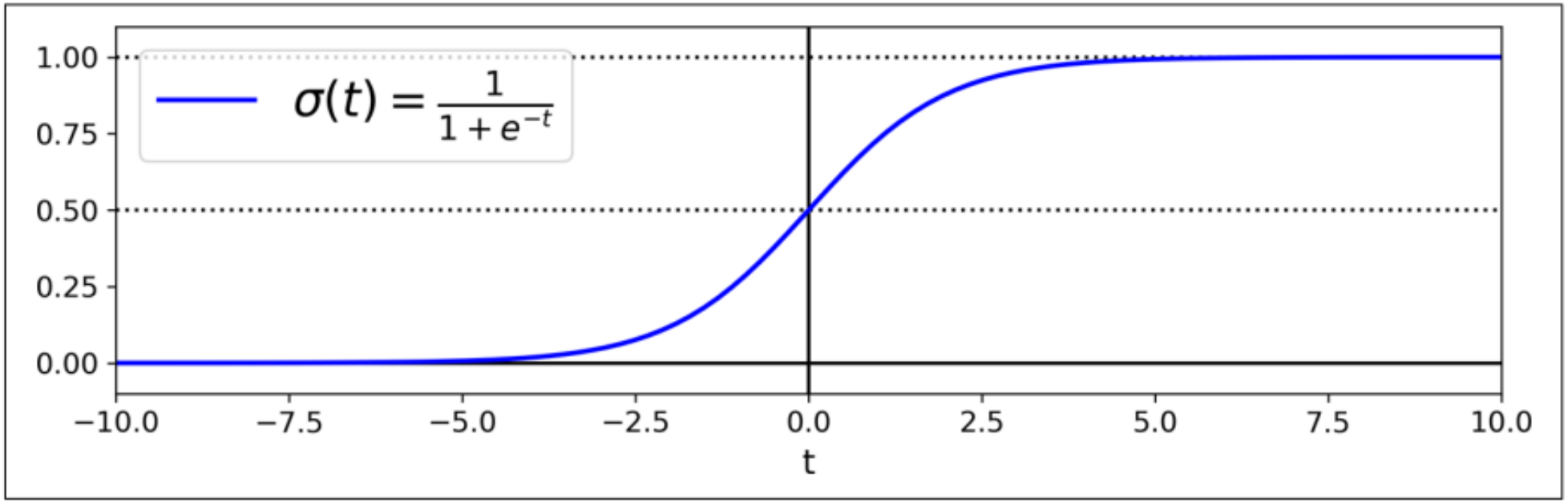
Lineare Regression

Verallgemeinerung auf höherdimensionale Daten: *n* Features

Möglichkeit: Einführen einer Polynomialfunktion

Logistische Regression

- Benutzt die logistische / sigmoide Funktion
- Meistens für binäre Klassifikatoren benutzt



Quelle: Geron (2019)

Datenreinigung und -vorbereitung

In der reellen Welt sind Daten oft unordentlich, falsch oder einfach nicht vorhanden

- Menschliche Fehler
- Computerfehler
- Daten nicht verfügbar

Annahme: Tabellarische Daten

Zum Nachdenken: was für problematische Fälle können vorkommen?

Datenreinigung und -vorbereitung
Fehlende Daten

Name	Education	Income	Date of Birth
Smith	Postgraduate	550000	1977-07-23
Petersen	NULL	72000	1999-11-02
Andrews	High School	140000	NULL

…

Möglichkeiten:

- Zeile komplett entfernen
- Mit Placeholder-Wert ersetzen (z.B. -1)
- Mit dem Durchschnitt / Median / am Öftesten erscheinenden Wert derselben Spalte ersetzen - nur für numerische Werte
- Falls ein Feature viele schlechte Werte hat, sollte man es vollständig aus dem Datensatz entfernen

Datenreinigung und -vorbereitung
Kategorische Daten

Kategorische Daten sind nicht Zahlenwerte, sondern gehören zu einer endlichen Menge an bestimmten Werten

z.B: Farbe, Geschlecht, Land

Datenreinigung und -vorbereitung
One-hot encoding

County
Cluj
Arad
Vaslui
…

Cluj_County	Arad_County	Vaslui_County	…
1	0	0	
0	1	0	
0	0	1	
…	…	…	…

WICHTIG! Die Kategorien haben keine intrinsische Ordnung, also würde eine Assoziation (Kreis, Zahlenwert) sinnlos sein. Welcher kann Kreis nr. 1 sein?

Pandas – die get_dummies Funktion

Es gibt auch andere Arten zum Encoding von kategorischen Variablen, diese ist die am meisten benutzte

Datenreinigung und -vorbereitung
Outliers

Definition: Daten die sehr weit außerhalb des erwarteten Intervalls liegen

- Manchmal falsche Daten, manchmal realistische aber tatsächlich seltene Fälle

- Beispiel: Startkoordinaten (Länge / Breite) von Taxifahrten in Cluj
  - Fahrt 1: 0.00000, 0.00000 - offensichtlich falsch: Fahrt vollständig entfernen
  - Fahrt 2: 46.77155, 23.62587 - Iulius Mall, offensichtlich ok: Fahrt behalten
  - Fahrt 3: 46.87952, 23.52569 - "la mama naibii" außerhalb von Chinteni: Fahrt mit Vorbehalt behalten – unwahrscheinlich, aber schon möglich irl

Datenreinigung und -vorbereitung
Skalieren

Betrachte die Größenordnungen der Features im folgenden Datensatz:

	Class label	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity	Hue	OD280/OD315 of diluted wines	Proline
0	1	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
2	1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
3	1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
4	1	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735

(the wine dataset, Kaggle, Auszug)

Datenreinigung und -vorbereitung
Skalieren

Verschiedene Features haben unterschiedliche Skalen:

- ist Feature A wichtiger als Feature B weil es größer ist?
- ist die Variation in Feature X (Werte 0.001 - 0.04) bedeutender als die in Feature Y (Werte 41000 - 72000)?

Um den Einfluss jedes individuellen Wertes richtig bewerten zu können, bringt mal alle Features auf demselben Maßstab

Datenreinigung und -vorbereitung
Skalieren

- **min-max scaling**: alle Werte werden auf das Intervall [0, 1] gebracht

$$x_{norm}^{(i)} = \frac{x^{(i)} - x_{min}}{x_{max} - x_{min}}$$

- **Normalisierung**: die Menge aller Werte wird auf Mittel 0 und Standardabweichung 1 gebracht

$$x_{std}^{(i)} = \frac{x^{(i)} - \mu_x}{\sigma_x}$$

Optimale Strategie hängt von Daten und Modell ab, z.B. Normalisierung konserviert Outliers besser, min-max scaling konserviert Distanz zwischen Punkten besser

Datenreinigung und -vorbereitung
Skalieren

Bei den meisten Modellen zählt die Skala **sehr viel!**

z.B. k-means clustering benutzt implizit euklidische Distanz (man kann aber auch mit anderen Distanzen clustern)

Das Benutzen von nicht skalierten Datensätze erzeugt sinnlose Ergebnisse

Ausnahme: Random forest & co.

Overfitting

- Wenn das Training sehr gute Ergebnisse auf die Testdaten gibt, aber schlecht auf neue Daten verallgemeinert
- Intuitiv: das Modell ist zu komplex, es hat zu viele Parameter
- Underfitting, wenn das Modell zu wenige Parameter hat, ist selten ein Problem in der Praxis
- Wenn das Ergebnis "too good to be true" ist, dann ist es meistens so!