Berner Fachhochschule
Haute école spécialisée bernoise
Bern University of Applied Sciences

Machine Learning Algorithmen: Grundlagen

CAS Practical Machine Learning

▸ Violeta Vogel, TI BFH

Agenda

- Distanzmasse
- Overfiting
- Training versus Anwenden

Distanzmass

- Die Wahl des Distanzmasses ist entscheidend, da sie definiert, was "Ähnlichkeit" in deinem Datensatz überhaupt bedeutet.
- Es hängt primär vom Skalenniveau deiner Daten ab:
  - Metrische Daten (Einkommen, Alter): Euklidische, Manhattan – Distanz

Euklidische Distanz

- Euklidische Distanz ist das am häufigsten verwendete Distanzmass in der Clusteranalyse.
- Man kann es wie „Luftlinie“ oder den direkten geometrischen Abstand zwischen zwei Punkten in einem Koordinatensystem vorstellen.

Euklidische Distanz

- Euklidische Distanz ist das am häufigsten verwendete Distanzmass in der Clusteranalyse.
- Man kann es wie „Luftlinie“ oder den direkten geometrischen Abstand zwischen zwei Punkten in einem Koordinatensystem vorstellen.
- Für zwei Punkte P und Q in einem mehrdimensionalen Raum berechnet sie sich nach dem Satz des Pythagoras:

$$d(P,Q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

- n: Anzahl der Merkmale (Variablen).
- q_i, p_i: Die Werte der beiden Objekte beim Merkmal i.

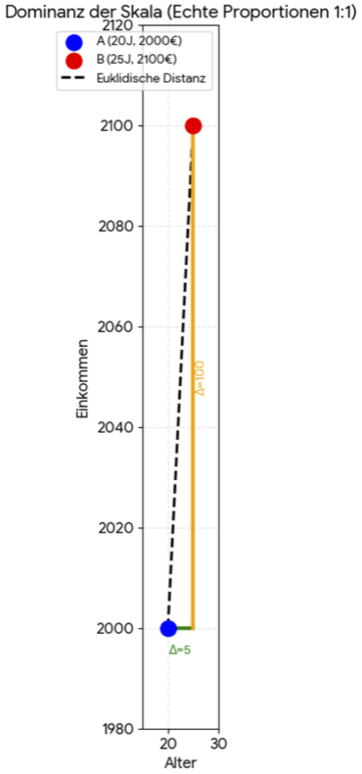
Euklidische Distanz

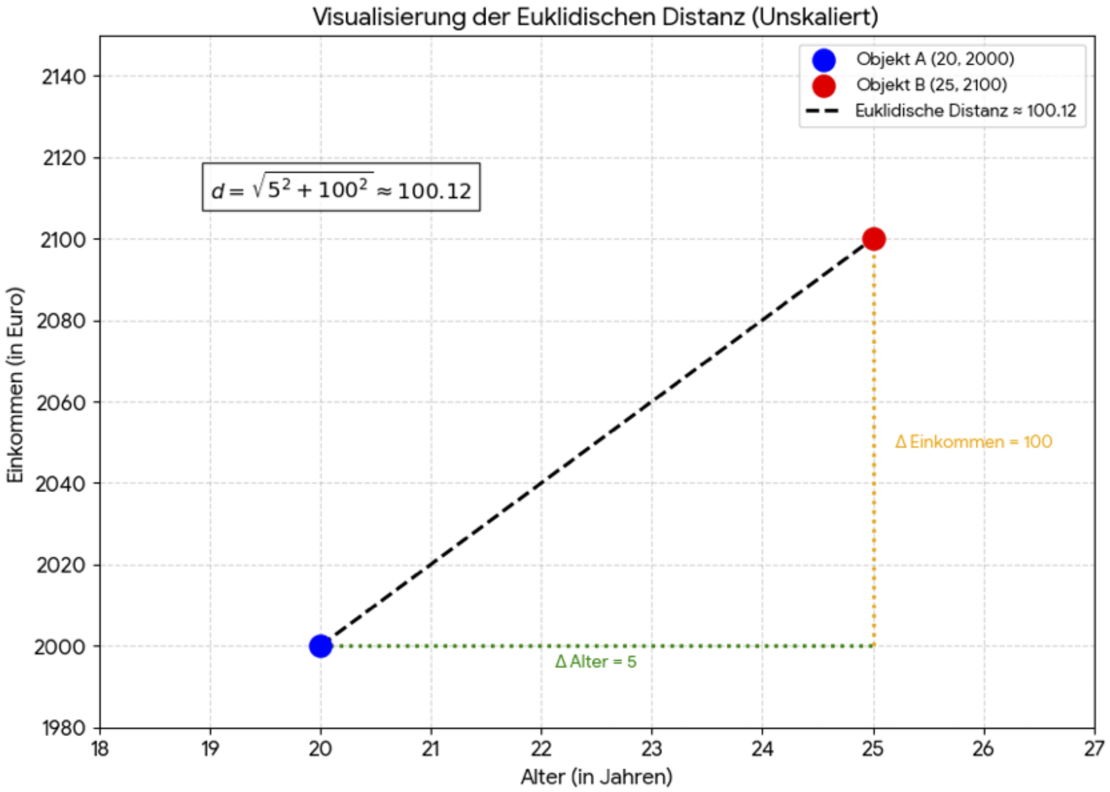
- Beispiel:
  - Objekt A: Alter 20, Einkommen 2000 CHF
  - Objekt B: Alter 25, Einkommen 2100 CHF
- Rechnung:

$$\sqrt{(25-20)^2+(2100-2000)^2}=\sqrt{5^2+100^2}=\sqrt{25+10.000}\approx 100,12$$

Euklidische Distanz

$$\sqrt{(25-20)^2+(2100-2000)^2} = \sqrt{5^2+100^2} = \sqrt{25+10.000} \approx 100,12$$



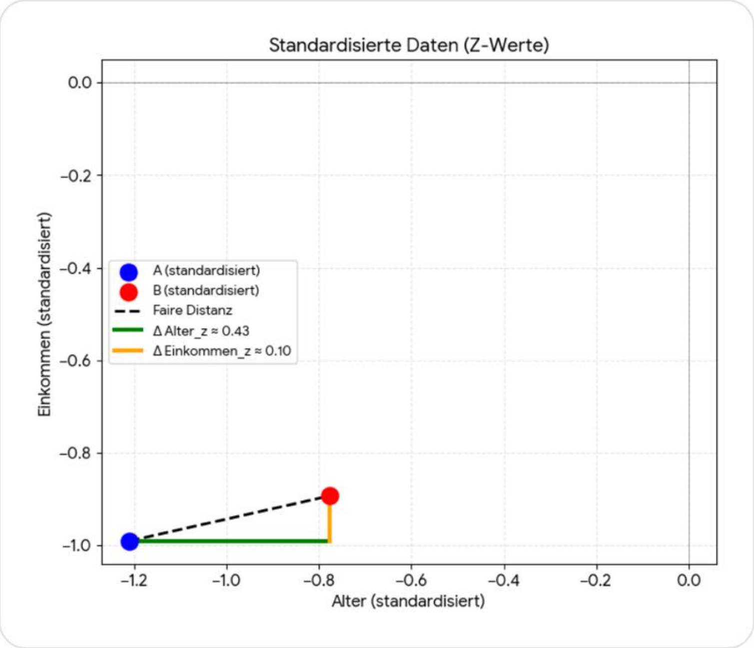


Euklidische Distanz

- Optische Verzehrung
- Die Euklidische Distanz (schwarz gestrichelt) ist fast identisch mit der orangen Linie.
- Der Altersunterschied von 5 Jahren fällt geometrisch kaum ins Gewicht.
- Für den Algorithmus sind diese beiden Personen fast nur aufgrund ihres Einkommens „verschieden“, das Alter wird zur Nebensache

Euklidische Distanz

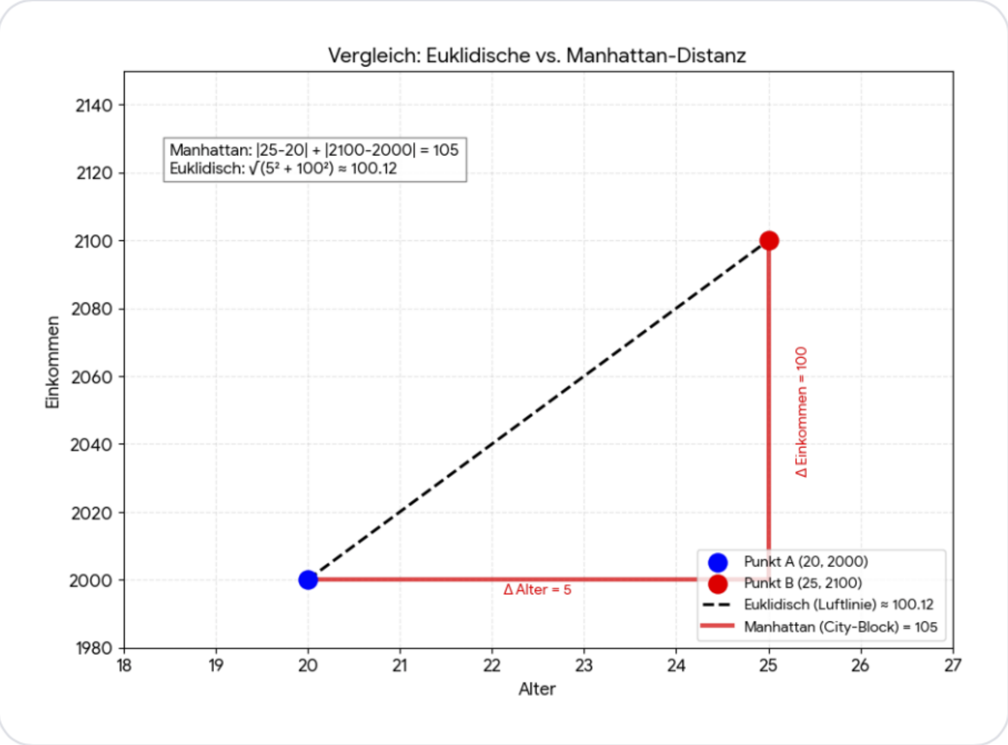
- Lösung: Standardisierung
  - Algorithmus erkennt beide Features als Gleich wichtig.
  - Die Werte liegen nun um den Nullpunkt herum. Ein negativer Wert bedeutet „unterdurchschnittlich“, ein positiver Wert „überdurchschnittlich“.



Manhattan-Distanz (Taxi Prinzip)

Distanzmass, das den Abstand zwischen zwei Punkten nicht als direkte Luftlinie, sondern als Summe der absoluten Differenzen ihrer Einzelmaße berechnet

Weniger anfällig für Ausreisser



Overfiting

- Overfitting bedeutet, dass ein Modell die Trainingsdaten zu gut auswendig lernt – inklusive Zufallsschwankungen und Rauschen – und dadurch auf neuen Daten schlecht abschneidet.
- Es wirkt dann intelligent, ist aber in Wahrheit nicht mehr generalisierungsfaehig.
- Das Modell passt sich so stark an die Trainingsdaten an, dass es Muster sieht, die in Wirklichkeit gar keine sind.
- Es passiert weil das Modell:
  - zu komplex ist (z. B. tiefe neuronale Netze, grosse Entscheidungsbaeume)
  - zu wenig Daten hat
  - zu lange trainiert wurde
  - Rauschen in den Daten als echtes Muster interpretiert

Overfiting

- Ein Entscheidungsbaum koennte so stark wachsen, dass er jeden einzelnen Trainingspunkt perfekt trennt – sogar Ausreisser.
- Auf neuen Daten versagt er dann, weil diese „Regeln“ nicht allgemein gueltig sind.
- Erkennung anahnd von typischen Muster:
  - Trainingsfehler sehr klein
  - Validierungs-/Testfehler deutlich groesser

Overfiting

- Wie verhindert man Overfitting?
  - Cross-Validation
  - Regularisierung (L1, L2)
  - Fruehzeitiges Stoppen
  - Mehr Trainingsdaten
  - Dropout bei neuronalen Netzen
  - Modell vereinfachen

Overfiting: Beispiel

- Ausgangslage: Wir haben ein kleines Dataset mit 6 Trainingspunkten. Die Aufgabe: Vorhersagen, ob ein Kunde kauft (1) oder nicht (0) basierend auf einem Score
- Trainingsdaten:

Kunde	Score	Kauf (Label)
A	10	0
B	20	0
C	30	1
D	40	1
E	50	1
F	60	0

Overfiting: Beispiel

- Zwei Modelle
- Wir vergleichen:
  - Ein einfaches Modell (z. B. logistische Regression)
  - Ein überkomplexes Modell (z. B. ein Entscheidungsbaum, der bis zum Maximum wächst)

Overfiting: Beispiel

- Ein einfaches Modell (z. B. logistische Regression)
  - Das Modell findet eine glatte Grenze, z. B.: Ab Score 35 steigt die Kaufwahrscheinlichkeit stark an.
- Fehler auf Trainingsdaten:
  - 1 Fehler (Kunde F wird falsch klassifiziert)
  - Trainingsgenauigkeit: 5/6 = 83 %
- Fehler auf Testdaten: Wir testen auf 4 neuen Kunden. 0 Fehler 100% Testgenauigkeit

Score	Tatsächlich	Modell sagt
15	0	0
25	0	0
45	1	1
55	1	1

- Fazit: Model generalisiert gut

Overfiting: Beispiel

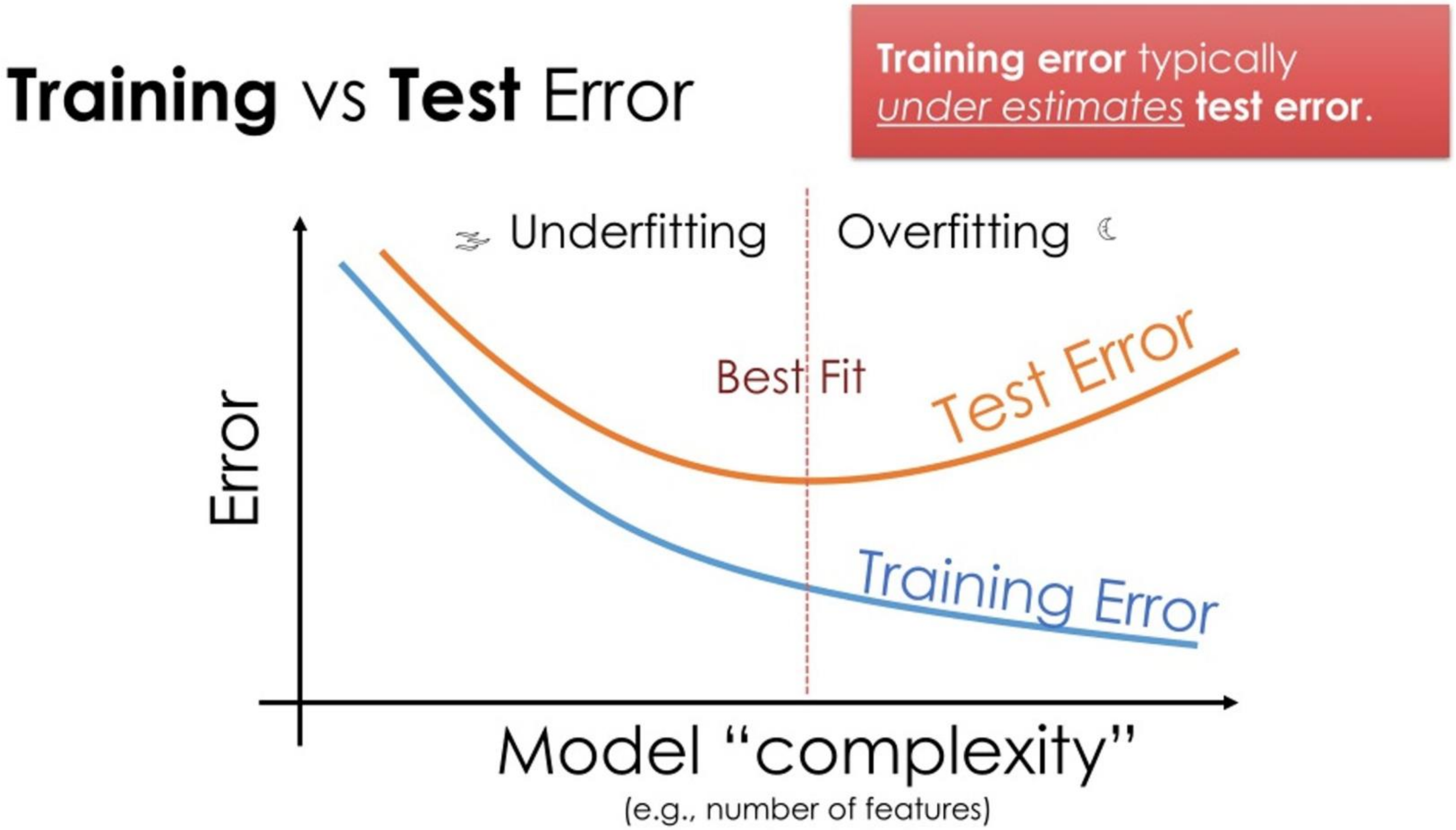
- Der Entscheidungsbaum lernt jede kleine Besonderheit:
  - Kunde F (Score 60) ist 0 → Baum baut eine Spezialregel
  - Kunde A (Score 10) ist 0 → Baum baut eine weitere Spezialregel
- Das Modell **trennt jeden Trainingspunkt perfekt.**

- Fehler auf Trainingsdaten
  - 0 Fehler
  - Trainingsgenauigkeit: 100 %

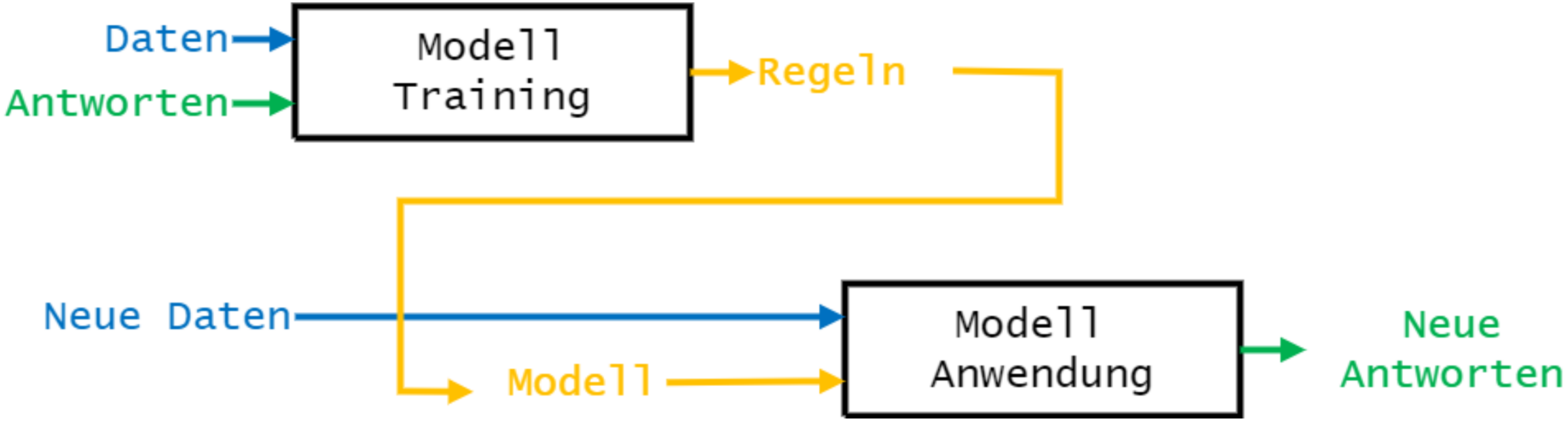
Score	Tatsächlich	Modell sagt
15	0	**1** (Fehler)
25	0	**1** (Fehler)
45	1	1
55	1	**0** (Fehler)

- Fehler auf Testdaten. Gleiche Testdaten wie oben: 3 Fehler, Testfehler 75%
- Fazit: Das Modell hat die Trainingsdaten **auswendig gelernt**, aber **nichts verstanden**.

Overfiting: Beispiel



Modeltraining versus Modelanwendung



Modeltraining versus Modelanwendung

- Modeltraining = Das Modell lernt aus Daten.
- Modelanwendung = Das Modell nutzt sein gelerntes Wissen, um Vorhersagen zu machen.

Modeltraining

- Das Training ist der Prozess, bei dem ein Machine-Learning-Modell aus Beispielen lernt:
  - Das Modell bekommt Daten (z. B. Bilder, Texte, Messwerte).
  - Es passt seine internen Parameter an, um Fehler zu minimieren.
  - Es braucht oft viel Rechenleistung und Zeit.
  - Ziel: Ein Modell erzeugen, das generalisiert.
- Beispiele:
  - Ein neuronales Netz lernt aus 1 Mio. Katzen- und Hundebildern, wie Katzen und Hunde aussehen.
  - Ein Sprachmodell lernt aus Textkorpora Grammatik, Zusammenhänge und Bedeutungen.
  - Ein Regressionsmodell lernt aus historischen Preisen, wie Immobilienpreise entstehen

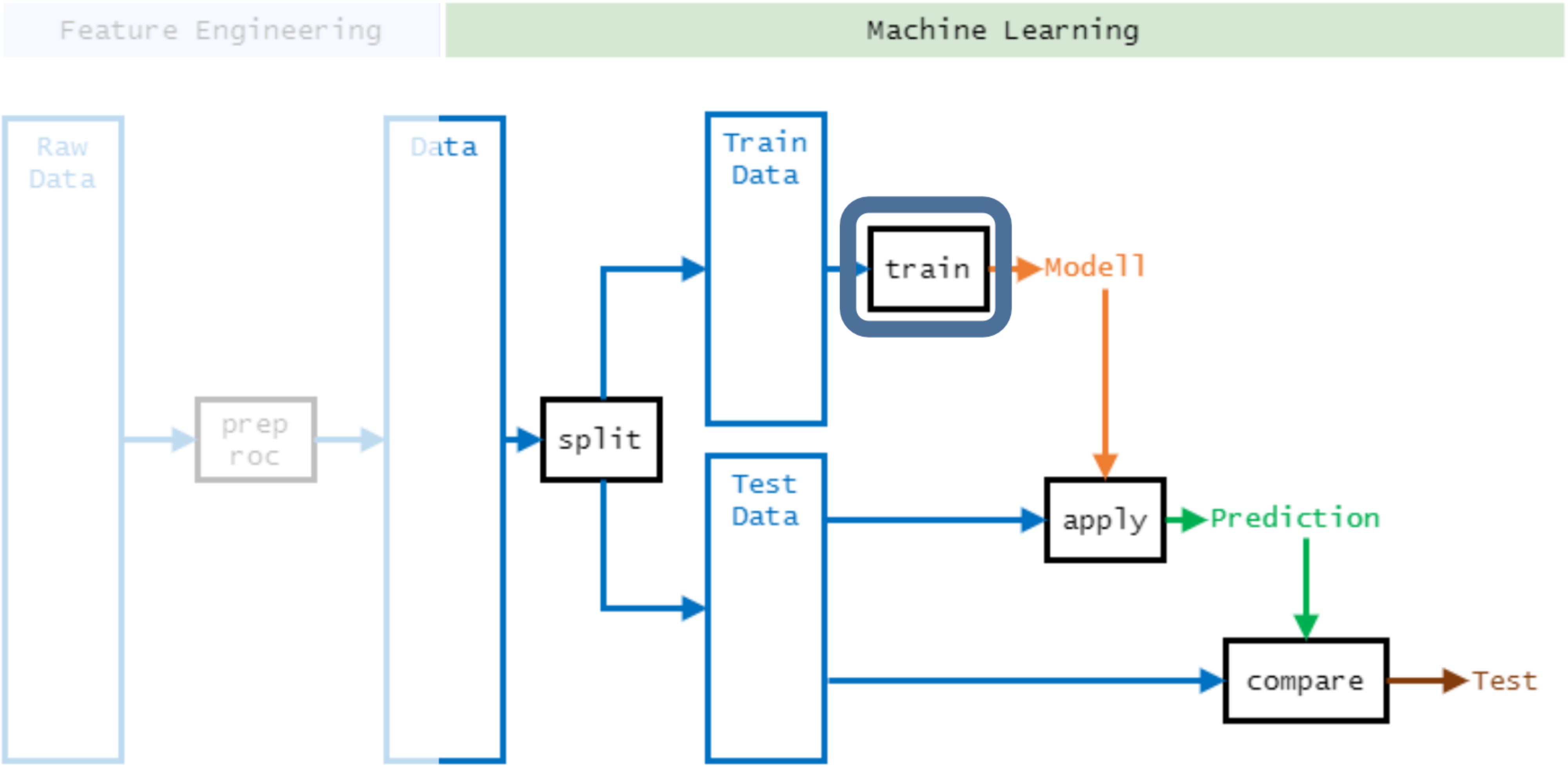
Modelanwendung

- Die Anwendung (Inference) ist der Einsatz des fertig trainierten Modells:
  - Das Modell bekommt neue, unbekannte Daten.
  - Es macht eine Vorhersage oder Entscheidung.
  - Es ist viel schneller als Training.
  - Ziel: Nutzen des gelernten Wissens.
- Beispiele:
  - Das trainierte Katzen/Hunde-Modell klassifiziert ein neues Foto.
  - Ein Chatbot beantwortet eine Frage basierend auf seinem trainierten Sprachmodell.
  - Ein Kreditrisikomodell bewertet einen neuen Kreditantrag.

Modeltraining versus Modelanwendung

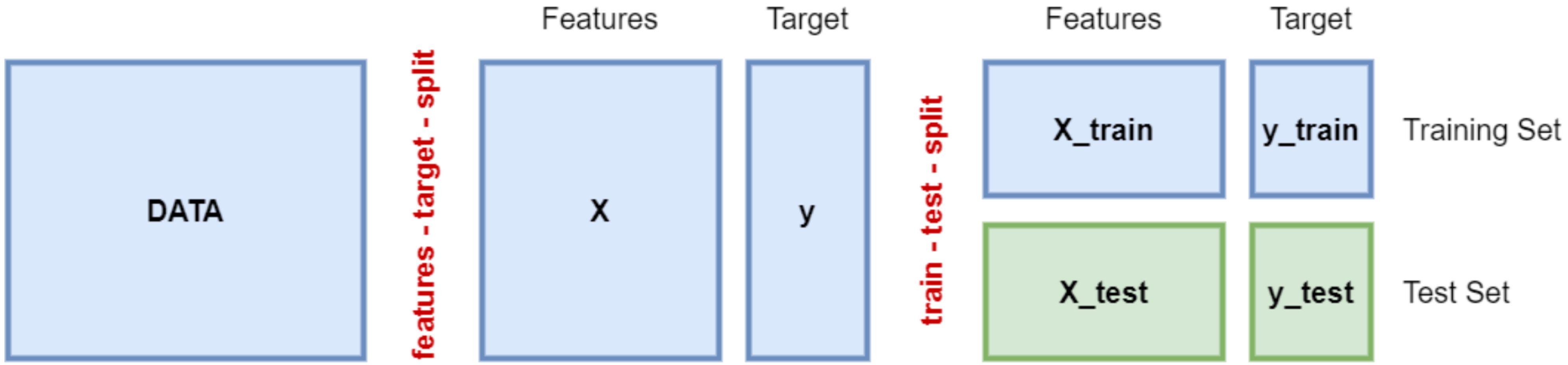
Phase	**Modeltraining**	**Modelanwendung**
Zweck	Lernen	Vorhersagen
Daten	Trainingsdaten	Neue Eingabedaten
Aufwand	Hoch (Rechenzeit, Energie)	Gering
Ergebnis	Trainiertes Modell	Output (z. B. Klassifikation)
Beispiel	Katzen/Hunde-Modell wird trainiert	Modell erkennt neue Katze

Modeltraining versus Modelanwendung



Modeltraining versus Modelanwendung

- tatsächlich sind beim Supervised Learning (Klassifikation und Regression) mit scikit-learn zwei aufeinander folgende Splits notwendig



- features - target - split: eine Spezialität von scikit-learn, die Features werden als Matrix (X) erwartet, die Target-Werte als Vektor (y), andere ML Tools (z.B. R) gehen zum Identifizieren von Features und Target andere Wege