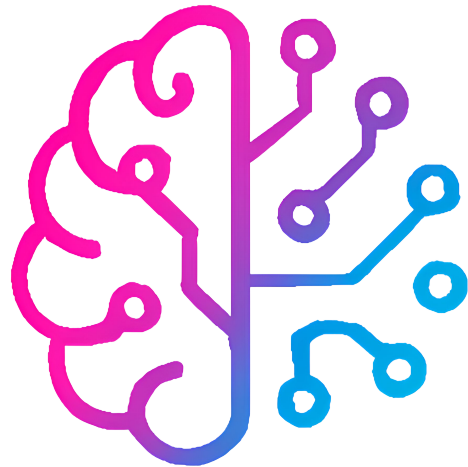


# Machine Learning Club

Zweites Treffen



# Gewinner der ersten Challenge

# Unterschiedliche Arten der Regression



# Was ist Regression?

- Regression ist ein **Supervised Learning-Verfahren**, genau wie Klassifikation.
- Aber: Statt Klassen vorherzusagen, sagt das Modell eine Zahl voraus.
- Ziel: Zusammenhang zwischen Merkmalen (Features) und einer kontinuierlichen Zielgröße (Target) finden.







Beispiele:

- Temperatur morgen in °C
- Alter einer Person basierend auf dem Gesicht
- Immobilienpreis in €



# Warum ist Regression wichtig?


- Ermöglicht **quantitative Vorhersagen**
- Oft Grundlage für wirtschaftliche und technische Entscheidungen
- Wird in vielen Bereichen verwendet:
  -  **Aktien**prognosen
  -  **Immobilien**bewertung
  -  **Medizinische** Messwerte
  -  **Preis**berechnung in E-Commerce



# Ziel einer Regression

- Das Modell versucht, eine **Funktion zu lernen**:

$$f(x_1, x_2, \dots, x_n) = y$$

- $x_1, x_2, \dots, x_n$ : Eingabewerte (z. B. Alter, Größe, Temperatur)
- $y$ : Zielwert (z. B. Preis, Zeit, Anzahl)
-  Ziel: Wenn ich neue Werte eingebe, gibt das Modell eine möglichst gute Schätzung für  $y$



# Lineare Regression



# Lineare Regression – die einfachste Form

- Das Modell versucht, eine **Gerade** zu finden, die die Daten beschreibt
- Beispiel: Je höher ein Merkmal, desto größer der vorhergesagte Wert
- Mathematisch:

$$y = a_1x_1 + a_2x_2 + \dots + b$$

- $a_1, a_2, \dots$ : Gewichtungen (werden gelernt)
  - $b$ : **Bias** (Verschiebung der Linie)
-  Vorteil: Sehr **einfach**, gut verständlich
-  Nachteil: Funktioniert **nur** bei **einfachen Zusammenhängen**







# Polynomiale Regression

- **Erweiterung** der linearen Regression:
- Modell nutzt auch quadratische, kubische usw. Terme:

$$y = a_1x + a_2x^2 + a_3x^3 + \dots$$

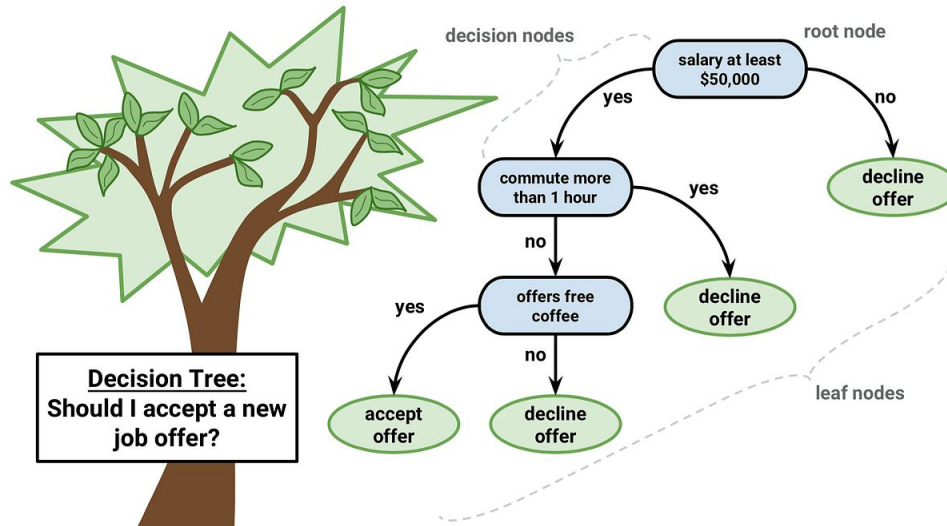
- Kann **gekrümmte Verläufe** beschreiben
-  Vorteil: **Flexibler** als einfache Gerade
-  Nachteil: Kann zu komplex werden → **Overfitting** (Modell passt sich zu stark an)

# Decision Tree Regression



# Decision Trees

Decision Trees sind einfache Modelle die Kategorisieren basierend auf einer Folge von Fragen mit Ja/Nein Antworten:





# Decision Tree Training

1. Ziel Variable festlegen
2. Besten Split finden
  - a. Über alle Features iterieren
  - b. Für jeden Feature mögliche Split Points festlegen
  - c. für jeden Split:
    - i. Datensatz in zwei Teile teilen
    - ii. Mean Squared Error ausrechnen
  - d. Split des Features auswählen der den Fehler minimiert
3. Decision Node kreieren
4. Rekursiv auf Kinder anwenden mit restlichen Features
5. Wiederholen bis ein Stop Kriterium erreicht ist



# Random Forest

- Kombiniert die Vorhersage mehrerer Decision Trees für eine akkuratere und robustere Vorhersage
- Ziehe gleichverteilt aus den Trainingsdaten und Features neue Trainingsdaten und Features und trainiere decision Trees auf den neuen Daten
- Bilde den Mittelwert der Vorhersagen (Aggregation)
- Weniger anfällig für hohe Varianz und Overfitting



# Regressionsarten Übersicht

- **Lineare** Regression: Einfachstes Modell, gute Interpretierbarkeit
- **Polynomiale** Regression: Modelliert **Kurven** statt Geraden
- **Ridge / Lasso** Regression: **Regularisierte** Varianten zur **Vermeidung** von **Overfitting**
- **Entscheidungsbäume** für Regression: Wenn-Dann-Regeln mit numerischem Output
- **Neuronale Netze**: Komplexe, nichtlineare Beziehungen möglich

# Wie bewertet man Regressionsmodelle?

## Mean Absolute Error (MAE)

- Durchschnittlicher Fehler (Betrag)







## (Rooted) Mean Squared Error ((R)MSE)

- Bestraft große Fehler stärker

## $R^2$ (Bestimmtheitsmaß)

- Wie gut erklärt das Modell die Variation der Daten?
- $R^2=1$ : perfekte Vorhersage,  $R^2=0$ : keine Erklärungskraft

# Typischer Ablauf

1.  Daten sammeln & aufbereiten
2.  Zielgröße bestimmen (z. B. Hauspreis)
3.  Merkmale auswählen (z. B. Wohnfläche, Lage)
4.  Modell trainieren
5.  Fehler messen und Modell verbessern
6.  Modell verwenden: Vorhersage für neue Fälle





# Regression vs. Klassifikation

Aufgabe	Klassifikation	Regression
Ziel	<b>Klasse</b> vorhersagen	<b>Zahl</b> vorhersagen
Beispiel	„Wird es regnen?“ (Ja/Nein)	„Wie viele mm Regen?“
Typ des Outputs	<b>Diskret</b> (z. B. 0 oder 1)	<b>Kontinuierlich</b> (z. B. 3.6)
Modell gibt zurück	<b>Klasse</b> oder Wahrscheinlichkeit	<b>Reelle Zahl</b>



Beide lernen aus Beispielen mit **bekannten Ergebnissen**.



Wenn die Antwort eine Zahl ist → benutzen wir Regression

# **Zweite Challenge: Leihfahrrad Nachfrage vorhersagen**

# Leihfahrrad Übersicht

- Nachhaltiger, budgetfreundlicher Ansatz für die Mobilität
- Der globale Fahrrad-Sharing-Markt wurde 2024 auf 9 Milliarden US Dollar geschätzt
- Steigenden Bevölkerungszahlen und Verkehrslast erhöhen die Nachfrage
- Oft effizienter als traditionelle Transportsysteme auf kurzen Strecken

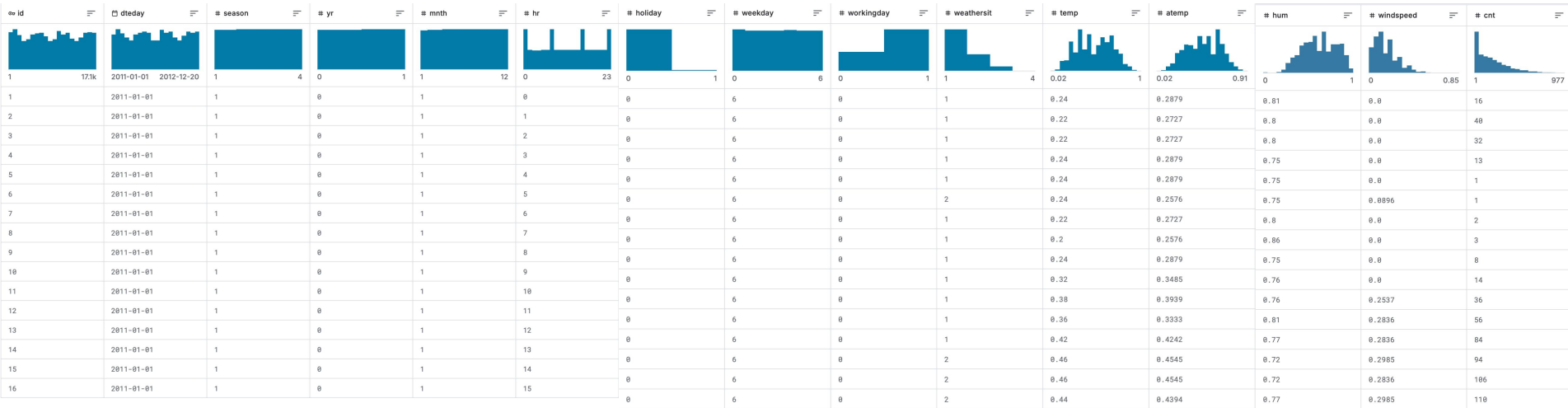


# Unser Datensatz

- Vorhersage der Anzahl von Nutzern gegeben der Uhrzeit und dem Wetter
- Ungefähr 17000 Datenpunkte
- Kann auch benutzt werden, um besondere Ereignisse vorherzusagen



# Aufbau der Daten





# Evaluation - Root Mean Square Error

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N \|y(i) - \hat{y}(i)\|^2}{N}}$$

$y(i)$  = i-ter richtiger Wert,  $\hat{y}(i)$  = i-ter vorhergesagter Wert,  $N$  = Anzahl Datenpunkte

- Niedrigster RMSE ist der beste
- Bestraft stark falsche Vorhersagen



# Regeln

- Max. 4 Leute pro Team
- Wettbewerb endet am **06.07 um Mitternacht**
- Nur **öffentliches Leaderboard**
- Max. 5 Abgaben pro Tag



# Preise

- Erster Platz: Titel des “**Regressions Ritter**” mit Pokal
- Bragging Rights ein Leben lang
- Gewinner stellen beim nächsten Treffen ihre Lösungen vor



# Kontakt

## Machine Learning Club

[contact@machine-learning.club](mailto:contact@machine-learning.club)

<https://machine-learning.club>

