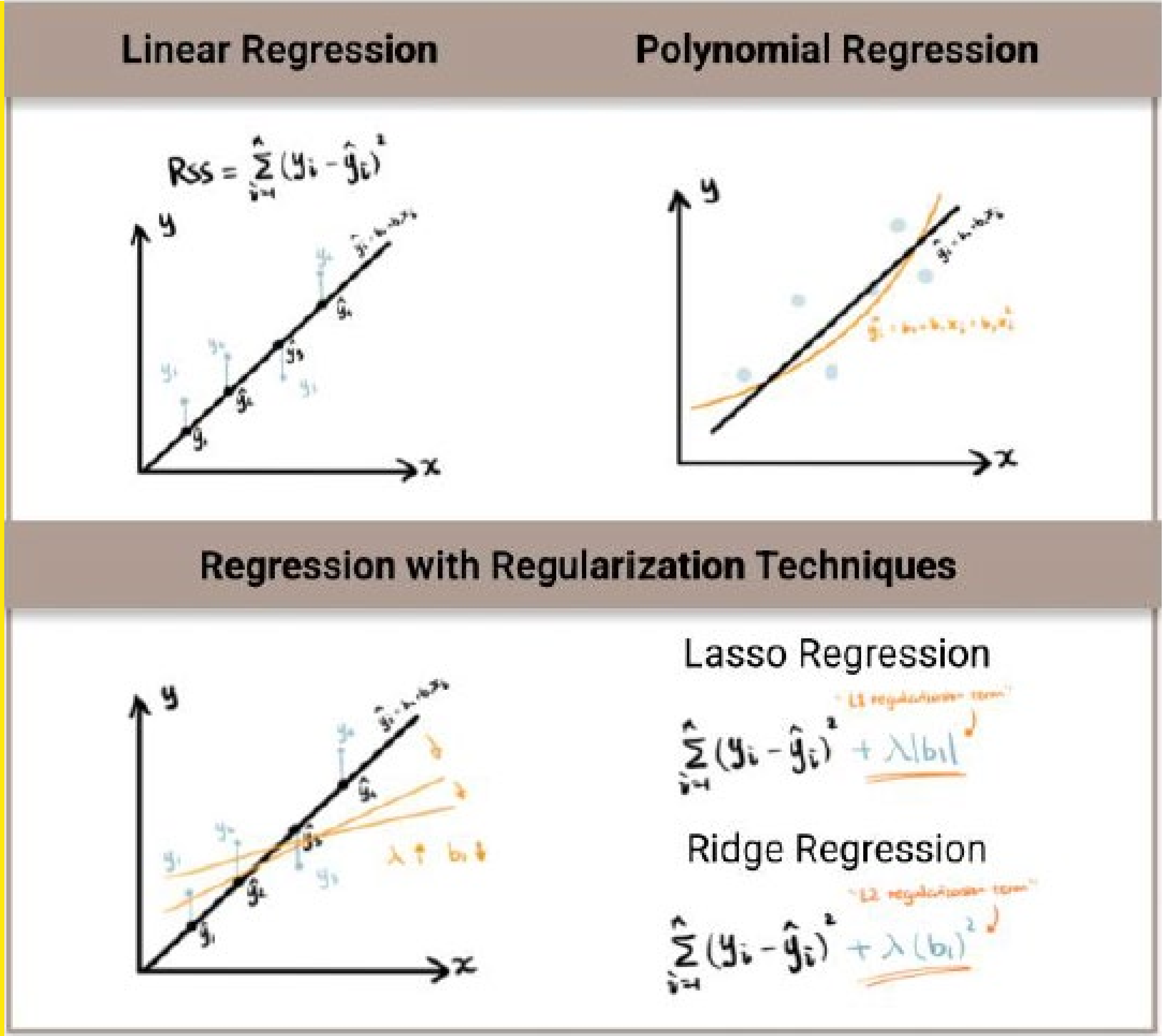
Fachhochschule Nordwestschweiz
Hochschule für Wirtschaft

Maschinelles Lernen – Regression



– Hans Friedrich Witschel, Andreas Martin

Running Example – Hausverkauf

Hausgrösse ("square feet")	Grösse Grundstück	Zimmer	Granit?	Extra Bad?	Verkaufspreis
3529	9191	6	0	0	$205,000
3247	10061	5	1	1	$224,900
4032	10150	5	0	1	$260,000
2397	14156	4	1	0	$189,900
2200	9600	4	0	1	$195,000
3536	19994	6	1	1	$265,000
2983	9365	5	0	1	$230,000

Die absolute Baseline – «Constant»

- **Vorhersage**: das arithmetische Mittel oder der Median der Zielvariablen auf den Trainingsdaten
- Orange: «Constant»

Parametrische Methoden

- Grundannahme: die Zielvariable y kann durch eine Funktion (z.B. linear oder Polynom) der Attribute x_i vorhergesagt werden
  z.B. linear:

$$y = w_0 + \sum_i w_i x_i = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_n x_n$$

- Aufgabe: die besten Werte für $w_0, w_1, \ldots, w_n$ finden

Einfache Lineare Regression

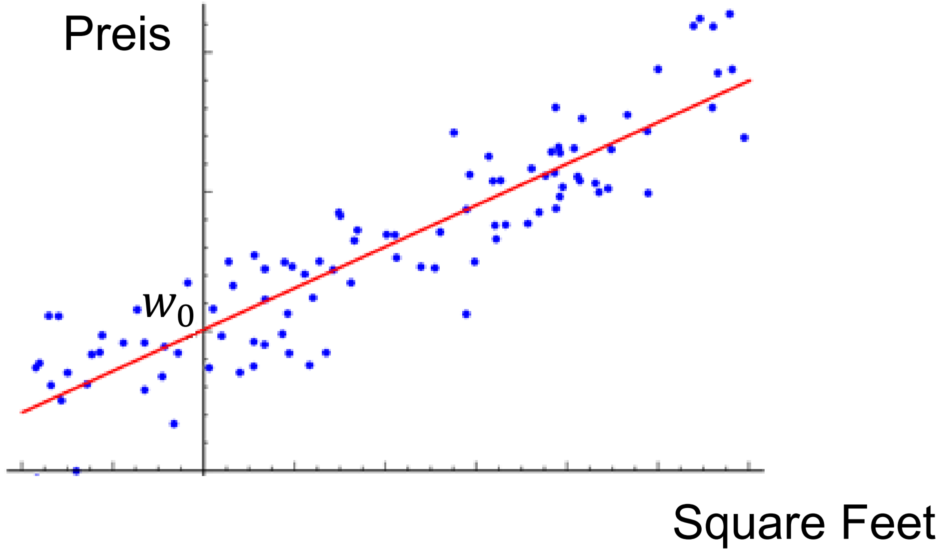
- Einfachster Fall: nur ein Attribut x, lineare Funktion

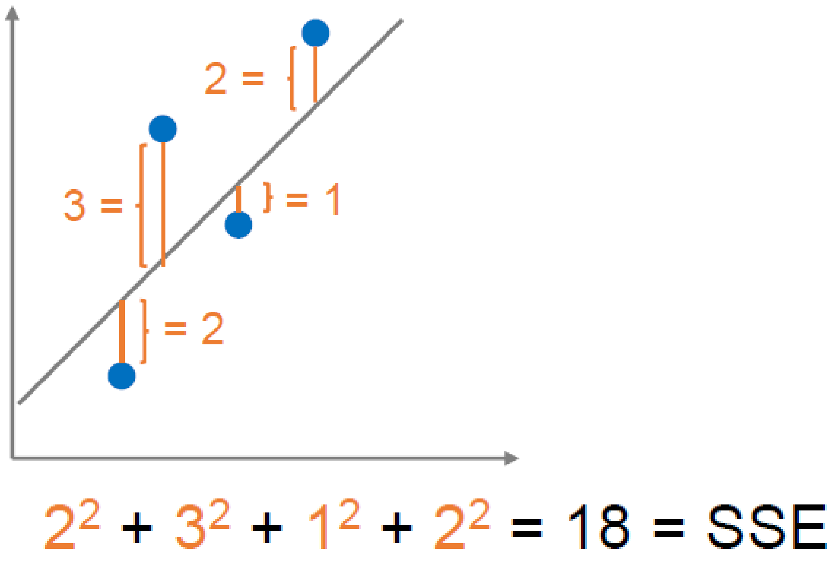
$$Preis = w_0 + w_1 \cdot sq.ft.$$

- Wie finde ich w_0 und w_1?
  → mittels SSE = Sum of Square Errors

- $SSE = \sum_i (\underbrace{Preis_i}_{\text{Echter Preis (blaue Punkte)}} - \underbrace{[w_0 + w_1 \cdot sq.ft._i]}_{\text{Durch Fkt. geschätzter Preis (Punkte auf der Linie)}})^2$

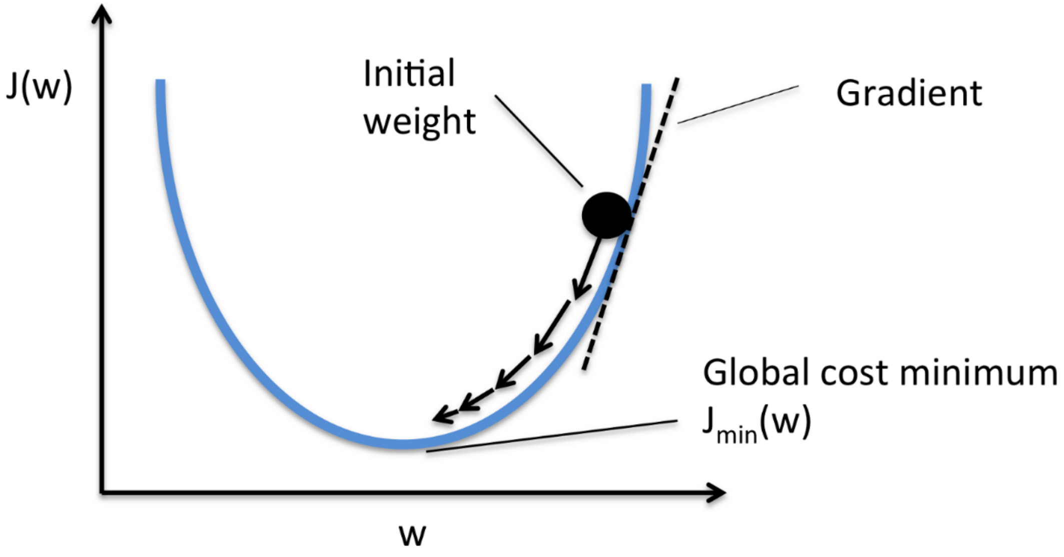
Echter Preis (blaue Punkte) *Durch Fkt. geschätzter Preis (Punkte auf der Linie)*



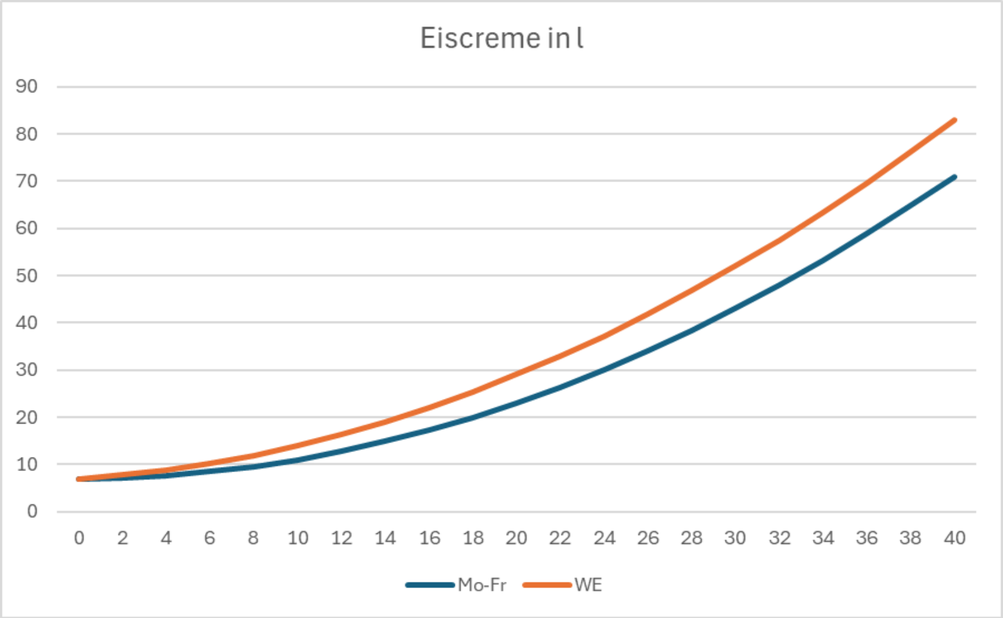


Gradient Descent

- Aufgabe: finde w_0 und w_1, so dass SSE minimal ist!
- Gradient Descent, am Beispiel von w_0:
  a. Wähle einen anfänglichen Wert von w_0 und w_1
  b. Berechne die Steigung von SSE bezüglich w_0
    – Leite SSE nach w_0 ab
    – Setze Square Feet und echten Preis des Trainingsbeispiels, sowie w_1 und aktuellen w_0-Wert ein
    – Steigung positiv → w_0 verringern (um «Schrittweite»)
    – Steigung negativ → w_0 vergrössern



Andere parametrische Verfahren



- Multiple lineare Regression:
  a. Mehr Attribute x_i, d.h. mehr Gewichte w_i zum Lernen
  b. Sonst gleiches Prinzip
  c. Probleme
    – Manchmal sind Zusammenhänge nicht linear
    – Manchmal spielen Interaktionen zwischen Attributen eine Rolle
    – Bsp.: $\text{Eiscrememenge } (l) = 7 + 0.04 \cdot Temperatur^2 + 0.3 \cdot Temperatur \cdot Wochenende$

- Andere Funktionen, z.B. Polynome
  a. Können auch «gemischte» Terme enthalten, sowie nichtlineare (siehe Bsp. oben)
  b. Prinzip des Gradient Descent bleibt gleich

Multiple Lineare Regression – Interpretation

- Wie kann man das interpretieren?

$$Preis = 92544 + 34.55 \cdot houseSize + 3.43 \cdot lotSize + \ldots$$

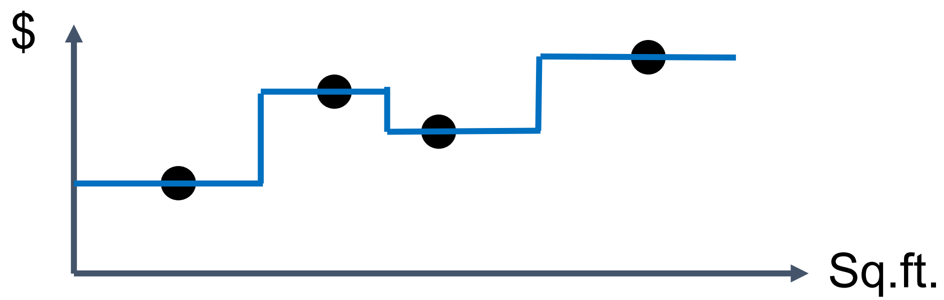
- Wieviel kostet also ein «square foot» Wohnfläche?

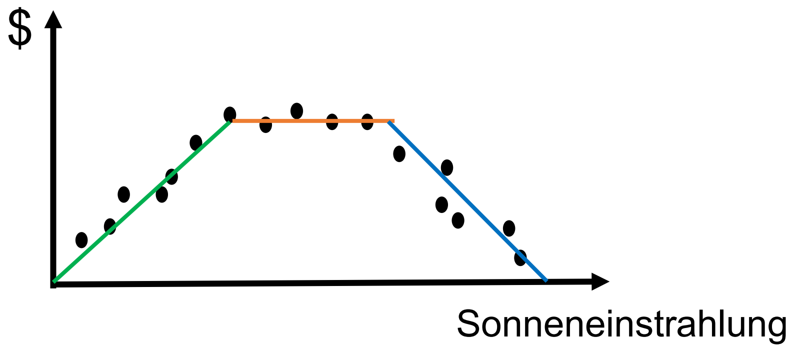
	name	coef
1	intercept	92544.2
2	houseSize	34.5512
3	lotSize	3.43763
4	bedrooms	-4708.79
5	granite=0	3472.01
6	granite=1	-3472.01
7	bathroom=0	-14054.5
8	bathroom=1	14054.5

Erklärung?

Nichtparametrische Verfahren – z.B. knn

- K nächste Nachbarn berechnen
- Vorhersage = (gewichteter) Mittelwert der Zielvariablen-Werte dieser Nachbarn
- Z.B. 1-NN für Hauspreise

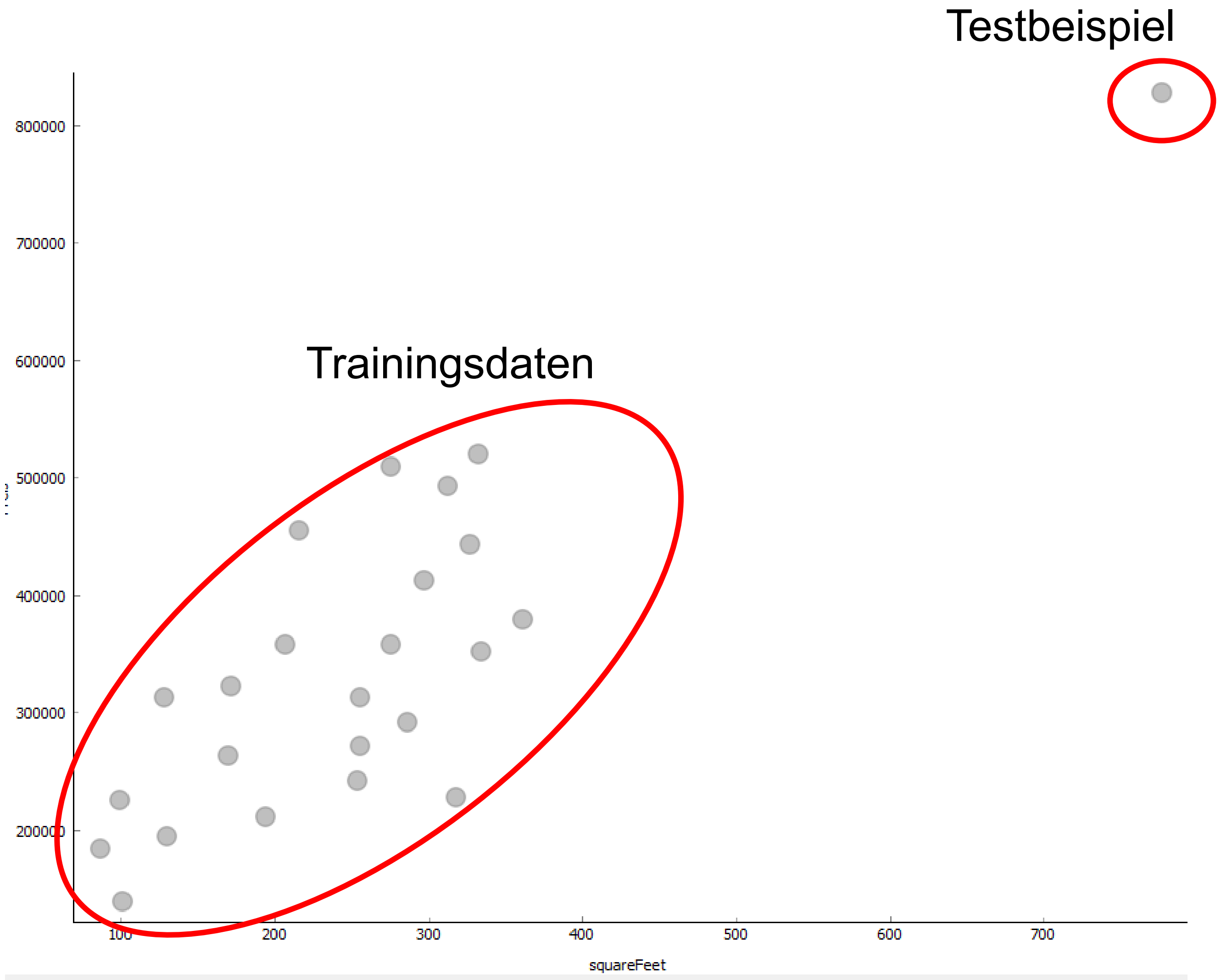




- **Vorteil**: kann gut sein, wenn Funktion nicht bekannt (z.B. ob linear) oder «ungewöhnlich» ist
- **Nachteil**: schlecht im Interpolieren in «dünn besetzten Regionen»

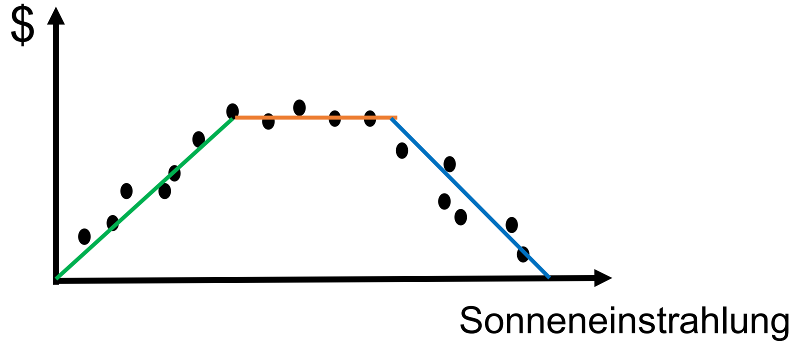
Lineare Regression vs. knn

- Preisfrage: wer sagt besser voraus? LR oder knn?



Entscheidungsbäume und Gradient Boosting

- Funktionieren genauso wie bei Klassifikation
  a. Vorhersagen sind «diskretisiert»
  b. Kombination möglich, z.B. M5P: Baum, dessen Blätter LR-Modelle sind
     → erlaubt wieder «kontinuierliche» Vorhersagen
  c. Gradient Boosting erlaubt Kombination mehrerer schwacher Modelle zu einem starken (wie bei Klassifikation)



- **Vorteil**: kann Interaktionen zwischen Attributen gut abbilden, teilweise auch Nicht-Linearität

Parametrisch vs. Nicht-parametrisch

	Parametric	Non-parametric
Accuracy (good guess of function type is available)	****	***
Accuracy (function type unknown)	*	***
Speed of learning	***	****
Speed of prediction	****	*
Tolerance to data sparsity	***	*
Tolerance to irrelevant attributes	***	*
Tuning effort	**	***
Explanation ability	****	*

Worüber wir später noch mehr lernen…

- **Evaluation**: Wie gut ist ein Regressionsmodell?
  a. Einfachstes Mass: Mean Absolute Error = mittlere (absolute!) Abweichung vom echten Wert
  b. Kann direkt in der Einheit der Zielvariable interpretiert werden (z.B. $ oder Liter)

- **«Komplexität»**: manchmal ist ein Modell zu sehr auf Eigenheiten der Trainingsmenge angepasst («Overfitting»)
  a. Für Lineare Regression: Komplexität kann durch Regularisierung gemindert werden (*Ridge* oder *Lasso*)

- Wie gesagt: später mehr…