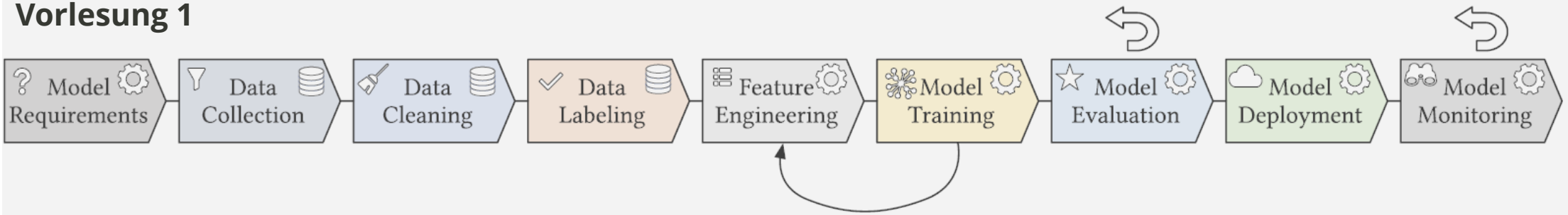
Dr. rer. nat. Valentin Khaydarov
Professur für Prozessleittechnik & Arbeitsgruppe Systemverfahrenstechnik

Regression

Vorlesung 3, Lehrveranstaltung Experimentelle Prozessanalyse

Einordnung der Vorlesung

Vorlesung 1



Vorlesung 2

Vorlesung 3 – Regr.

Vorlesung 4 – Class.

Vorlesung 5 – Clust.

Vorlesung 6 - Zeitreihenanalyse

Vorlesung 7 – Neuronale Netze

S. Amershi *et al.*, "Software Engineering for Machine Learning: A Case Study," in *Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP 2019*, May 2019, pp. 291–300, doi: 10.1109/ICSE-SEIP.2019.00042.

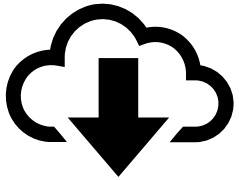
Agenda

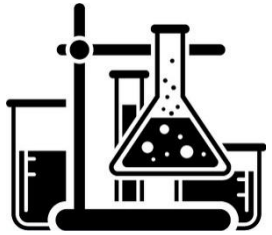
- Wiederholung
- Regressionsanalyse: Intuition und formale Problemdefinition
- Analytische und numerische Parameterschätzung
- Nichtlineare Regression
- Evaluation des Modells
- Regularisierung
- Validierung und Testen des Modells
- Zusammenfassung und Ausblick

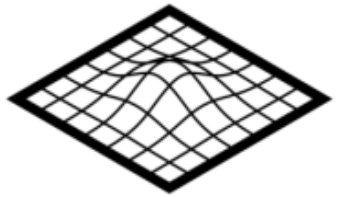
Wiederholung der letzten Vorlesung

Wiederholung

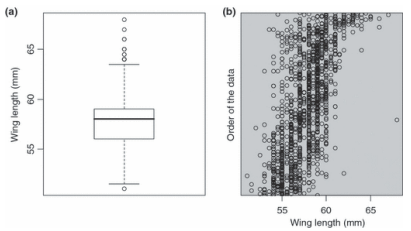
Datenbeschaffung

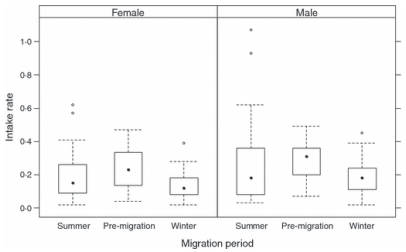


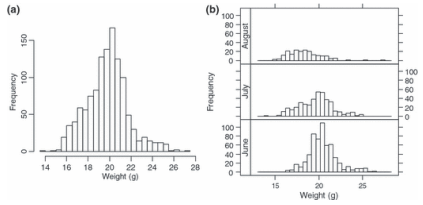


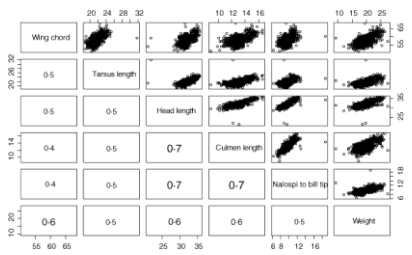


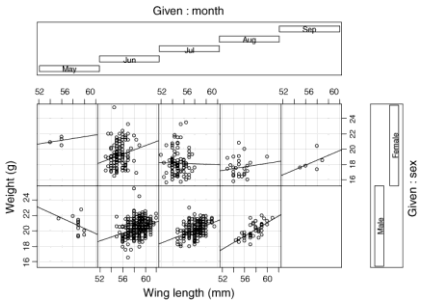
Datenexploration









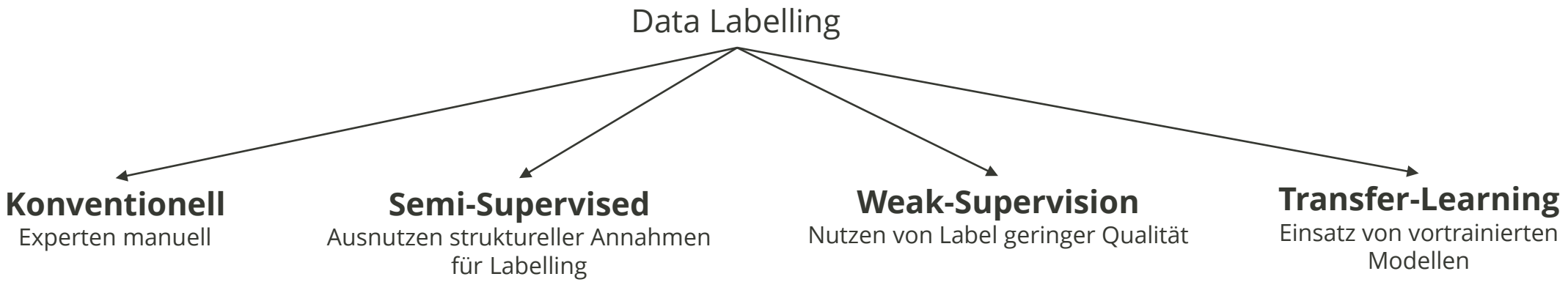


Wiederholung

Cleaning

Fehlende Werte, Ausreißer, Datendrift, Multikolinearität, Abtastrate und Verzögerungen

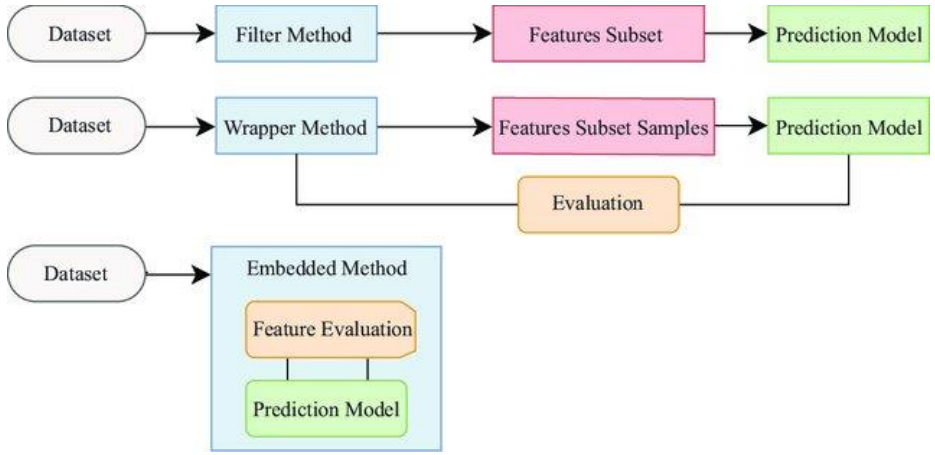
Labeling



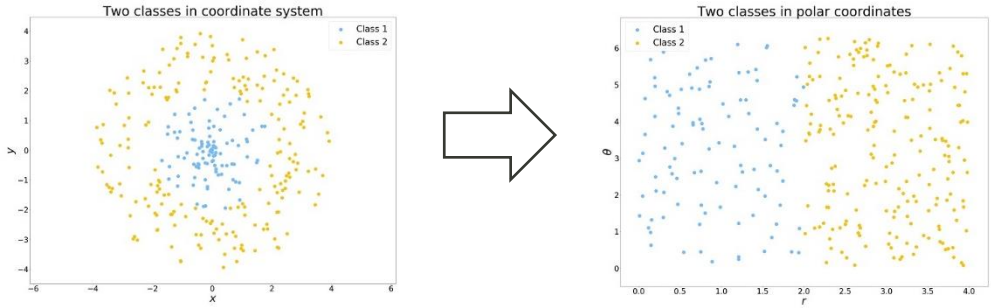
Wiederholung

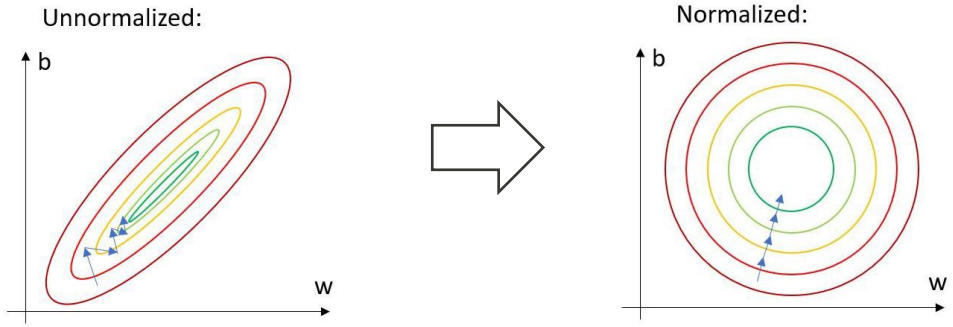
Feature Engineering

Selection



Transformation und Normalization

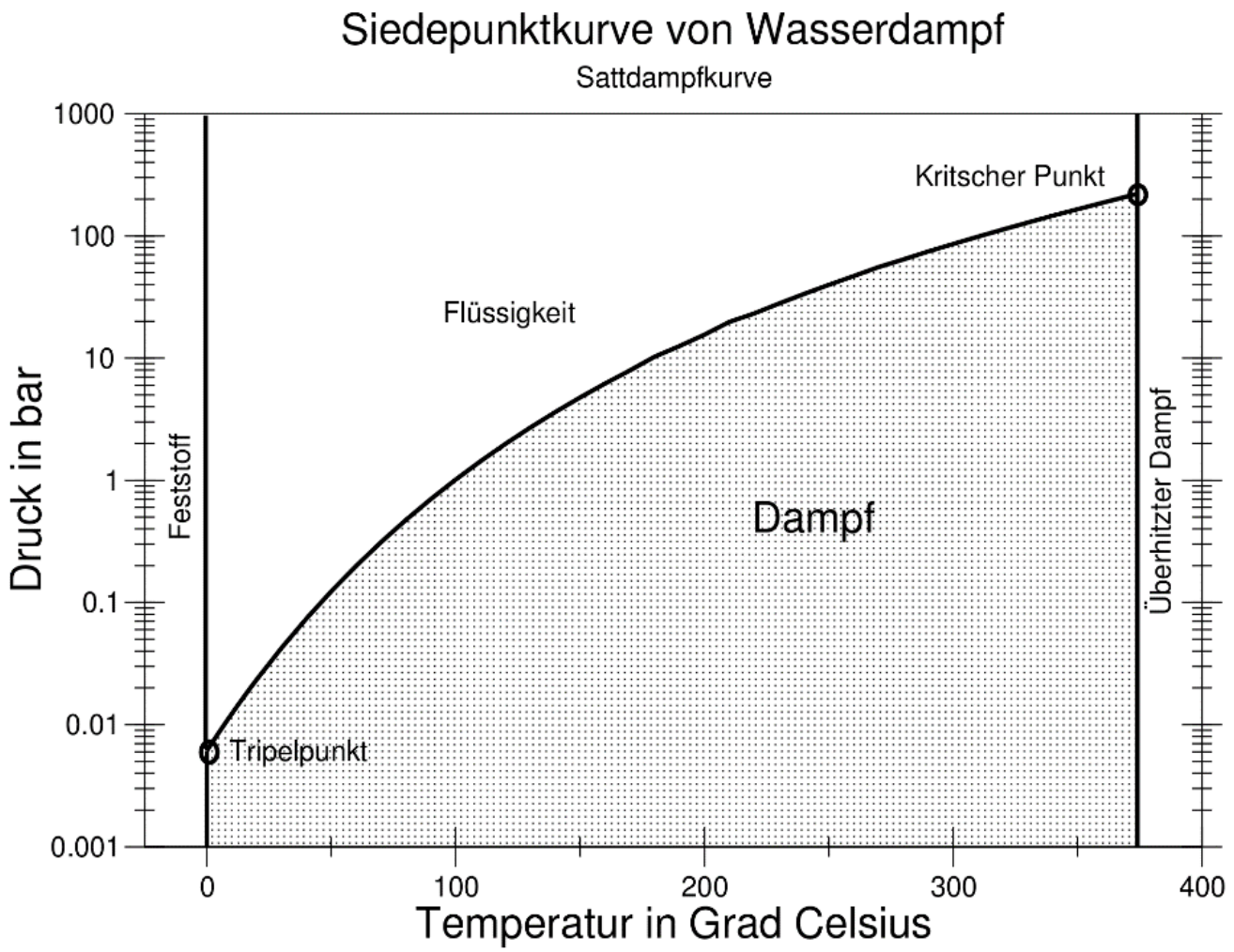




https://towardsdatascience.com/how-to-calculate-the-mean-and-standard-deviation-normalizing-datasets-in-pytorch-704bd7d05f4c

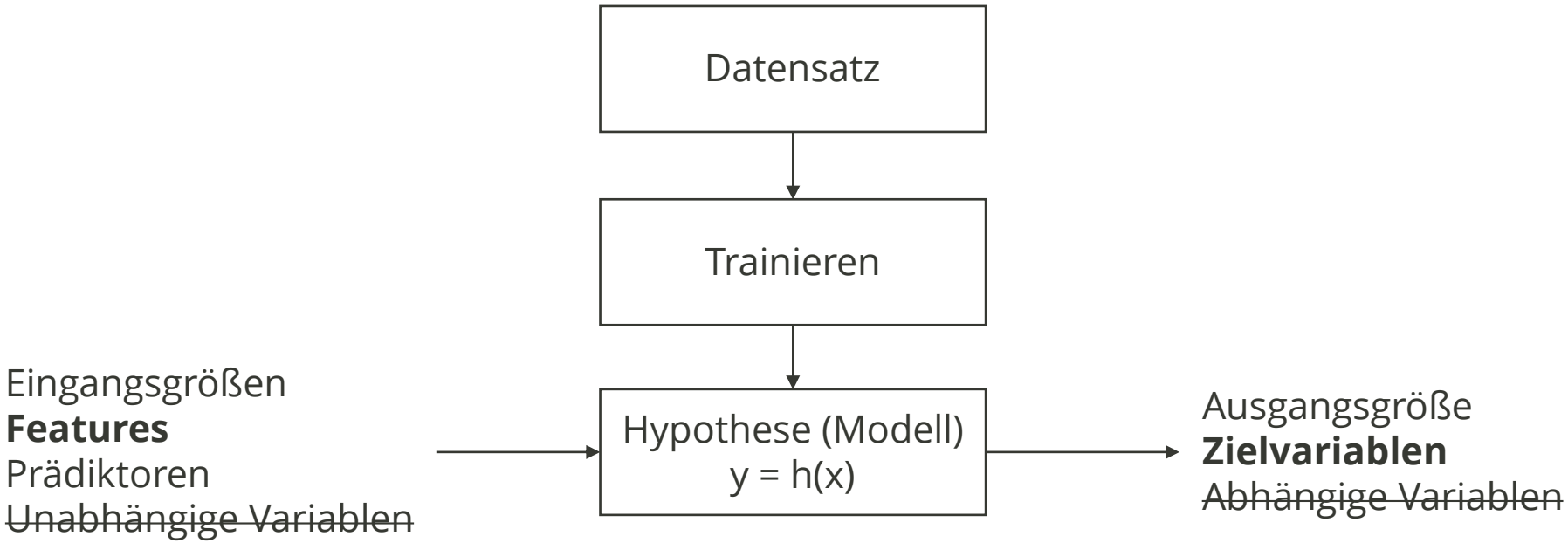
Regressionsanalyse

Beispiel: p-T Diagramm für Wasser



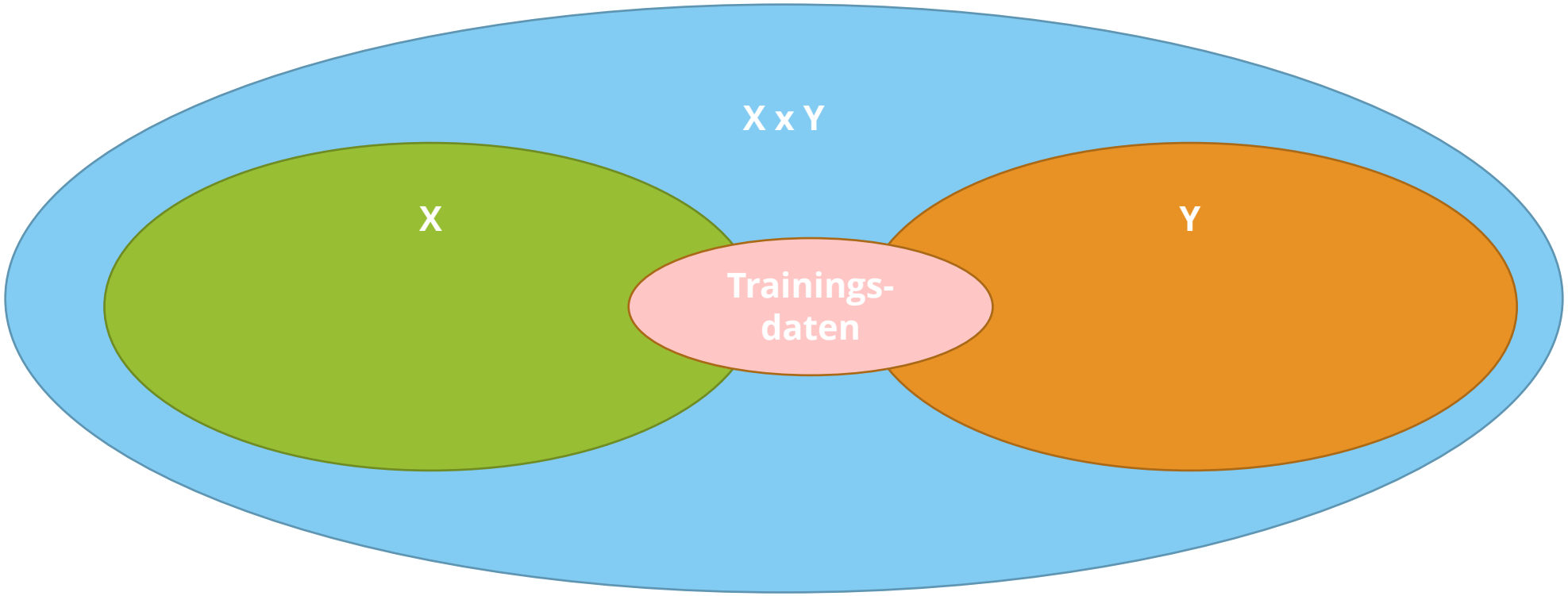
https://de.wikipedia.org/wiki/Datei:Dampfdruckkurve.svg

Intuition



Mittels eines Regressionsmodells werden **Zusammenhänge** zwischen **mehreren Merkmalen** durch ein **mathematisches Modell** abgebildet.

Intuition



Wir suchen die wahre Funktion, aber gegeben sind nur **eine Teilmenge** der Beobachtungen.

Regressionsanalyse

Vorgegeben:

- n Beobachtungen mit m Eingangsgrößen $x_1, \dots, x_m$ und einer Ausgangsgröße y
  - Eingangsgrößen können ebenfalls nominal oder ordinal sein
- eine Hypothese $y = f(b, x)$

Ziel

- Ermittlung eines funktionalen Zusammenhang zwischen m Eingangsgrößen und einer ausgewählten Zielgröße y

Regressionsanalyse

Lineare Hypothesen:

- Einfachregression ($m = 1$):

$$h(x) = b_0 + bx_1 \text{ oder Gerade im } \mathbb{R}^2$$

- Mehrfachregression ($m > 1$):

$$h(x) = b_0 + b_1 x_1 = b_0 + \sum_{j=1}^{m} b_j x_j$$

Notation

- Eine Beobachtung

$$\underline{x} = \begin{bmatrix} x_1 \\ \dots \\ x_n \end{bmatrix} \quad \underline{y} = y \quad \text{oder} \quad \left(\underline{x}, \underline{y}\right)$$

- Vektor mit m Beobachtungen

$$X = \left[\underline{x}^{(1)} \quad \dots \quad \underline{x}^{(m)}\right] \quad Y = \left[\underline{y}^{(1)} \quad \dots \quad \underline{y}^{(m)}\right]$$

- Modellparameter

$$\underline{b} = \begin{bmatrix} b_1 \\ \dots \\ b_n \end{bmatrix}$$

Notation

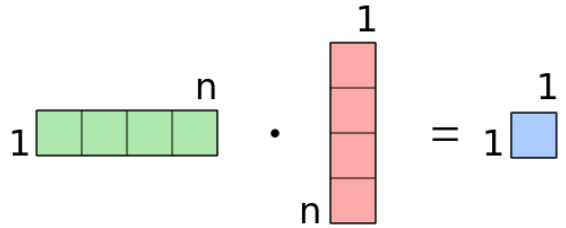
Vereinfachung:

$$h(x) = b_0 + b_1 x_1 = b_0 x_0 + b_1 x_1 = \sum_{j=0}^{m} b_j x_j \text{ mit } x_0 = 0$$

Als Matrizenmultiplikation:

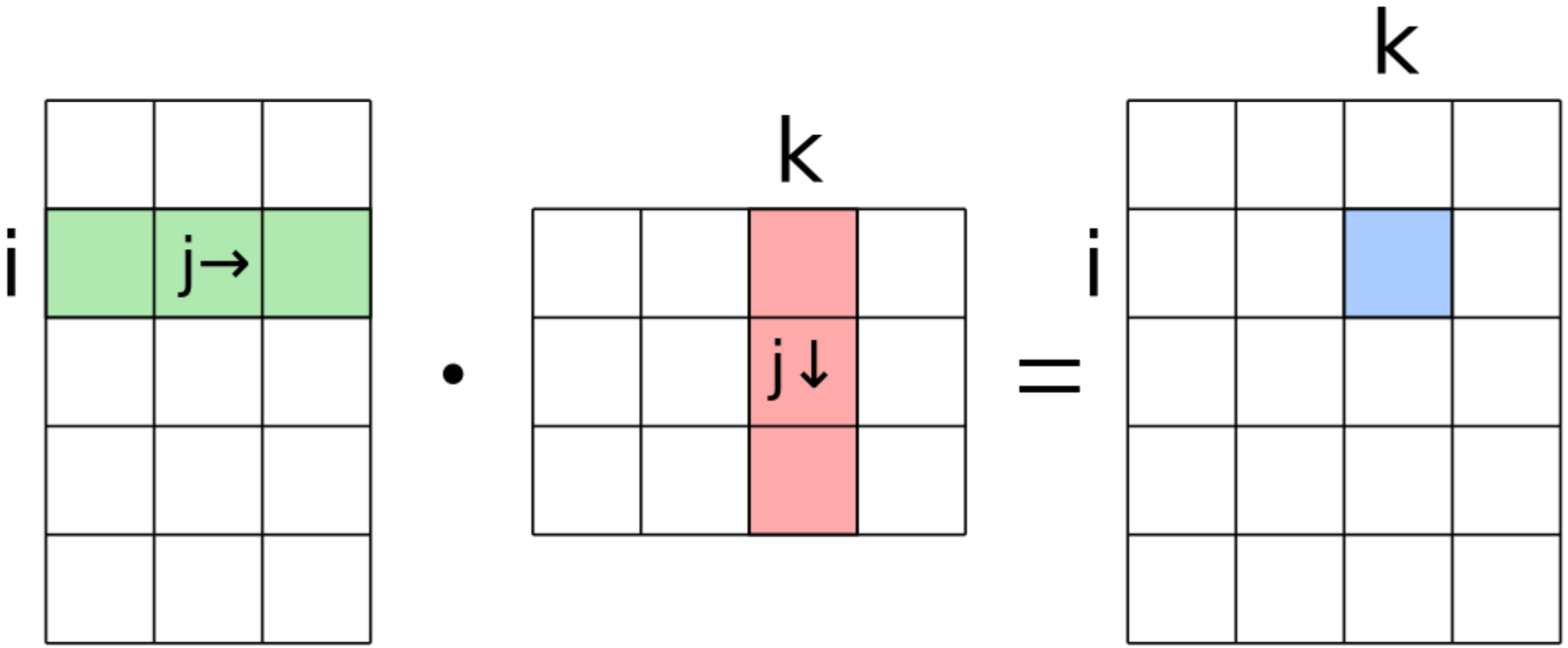
$$h(x) = \underline{x}^T \underline{b}$$

Trick: Multiplikation eines Zeilenvektors mit einem Spaltenvektors: (1 x n) * (n x 1)



https://de.wikipedia.org/wiki/Matrizenmultiplikation

Matrizenmultiplikation

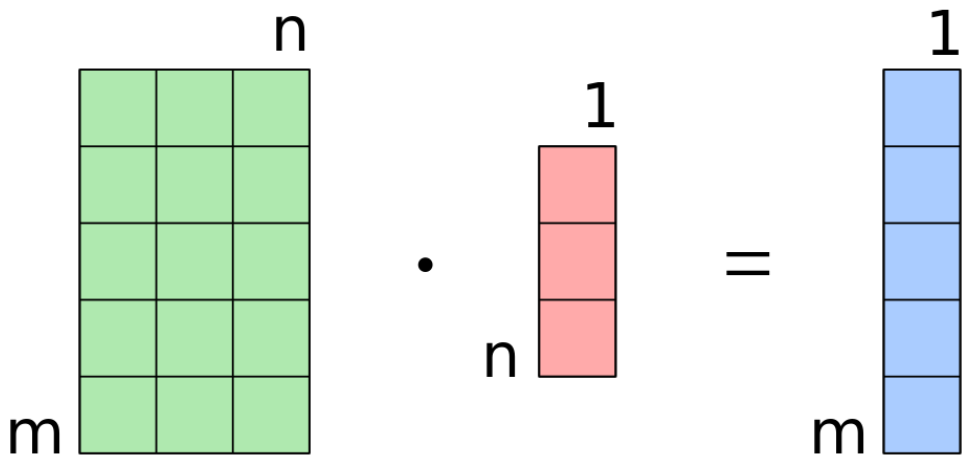


https://de.wikipedia.org/wiki/Matrizenmultiplikation

Notation

Multiplikation einer Matrix mit einem Spaltenvektors

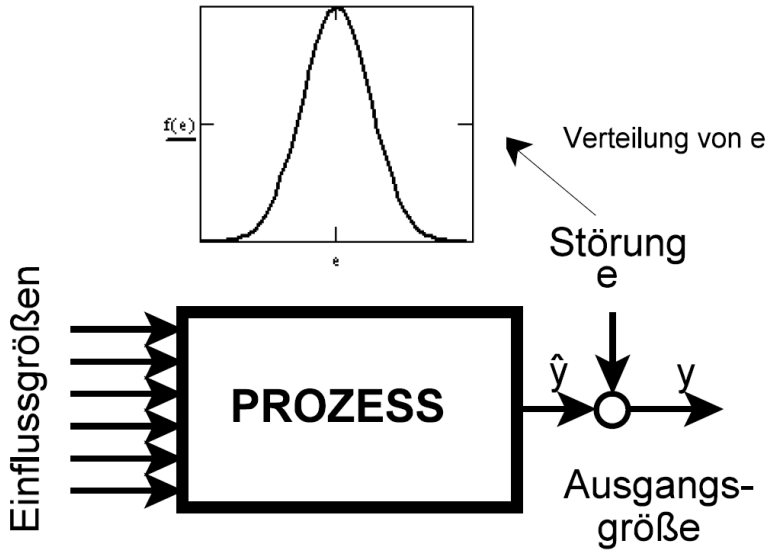
$$X^T \underline{b} = ???$$



https://de.wikipedia.org/wiki/Matrizenmultiplikation

Störgröße

- Messung der Ausgangsgröße:
  - nicht der “wahren” Wert $\hat{y}$
  - durch die Störgröße e “verfälschten” Wert y
  - $y = \hat{y}(x_1, x_2, \dots, x_m) + e$



- Störgröße e:
  - unkorreliert
  - Erwartungswert von Null
  - homogene Varianz
  - Ergebnis von additiven Überlagerung sehr vieler Störungen
    => normalverteilt (“zentralen Grenzwertsatz der mathematischen Statistik”)

Parameterschätzung

Parameterschätzung

$$h(x) = \underline{x}^T \underline{b}$$

Das Modell ist durch $\underline{b}$ parametrisiert.

Was ist ein guter Vektor $\underline{b}$?

Wie lässt sich die Güte des gefundenen Vektor $\underline{b}$ und des Modells beurteilen?

Gibt es einen optimalen Vektor $\underline{b}$?

Welche Methoden gibt es um einen guten/optimalen Vektor $\underline{b}$ zu finden?

Fehlerfunktion

Eine einfache Möglichkeit:

$$Err(\underline{b}) = \sum_{i=1}^{n} \left| Y - h(\underline{b}, X) \right|$$

Quadratische Fehlersumme:

$$SSE(\underline{b}) = \sum_{i=1}^{n} {e_i}^2 = \sum_{i=1}^{n} \left(Y - h(\underline{b}, X) \right)^2 = \left(Y - h(\underline{b}, X) \right)^T \left(Y - h(\underline{b}, X) \right)$$

Warum SSE? **Markow-Gauß-Theorem**

Satz von Markow-Gauß

Der Kleinste-Quadrate-Schätzer ist die beste lineare erwartungstreue (BLUE – best linear unbiased estimator) Schätzfunktion, wenn die zufällige Störgrößen:

- unkorreliert (immer der Fall für unabhängige Zufallsvariablen)
- im Mittel Null sind
- eine endliche konstante Varianz haben

Minimierungsproblem

Es sollen diejenigen optimalen Parameter $\underline{b}$ ausgewählt werden, bei denen die Summe der quadrierten Anpassungsfehler minimal wird:

$$\min_{\underline{b}} \sum_{i=1}^{n} \left(Y - h(\underline{b}, X)\right)^2$$

Methoden:

- Analytisch (exakt)
- Numerisch (approximativ)

Analytische Parameterschätzung

Schätzung $\hat{\underline{b}}$ der Koeffizienten $\underline{b}$ nach Methode der kleinsten Fehlerquadrate:

$$SSE(\underline{b}) = \sum_{i=1}^{n} {e_i}^2 = \underline{e}^T \underline{e} = \left(Y - X^T \underline{b}\right)^T \left(Y - X^T \underline{b}\right)$$

Notwendige Bedingung:

$$\frac{\partial SSE(\underline{b})}{\partial \underline{b}} = X^T \left(Y - X^T \underline{b}\right) = 0$$

Grundgleichung zur Berechnung der Regressionskoeffizienten

$$\hat{\underline{b}} = (X^T X)^{-1} X^T Y$$

$X^T X$ darf nicht singulär sein

Analytische Parameterschätzung

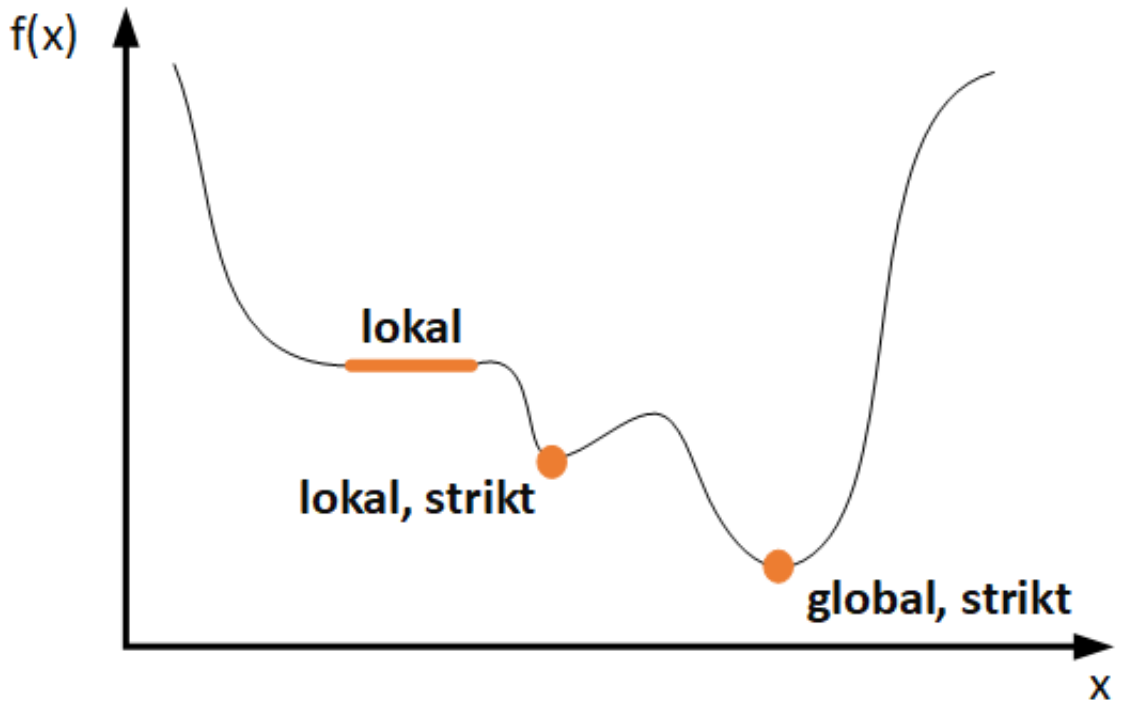
Aber:

- $X^T X$ kann in der Realität doch singular sein, wenn:
  - Inputvariablen abhängig sind
  - Anzahl von Parametern weniger als Beobachtungen im Datensatz

  → mehr als eine Lösung
- Fluch der hohen Dimensionen
  - Dimension entsprich der Anzahl von Beobachtungen
  - Matrixinversion ist keine gute Idee (100.000 x 100.000 – Aufgabe für einen HPC)

Numerische Optimierung

- eine Klasse von (iterativen) Optimierungsverfahren zur Suche eines (lokalen) Optimums für (nicht-lineare, komplexe) Optimierungsprobleme.



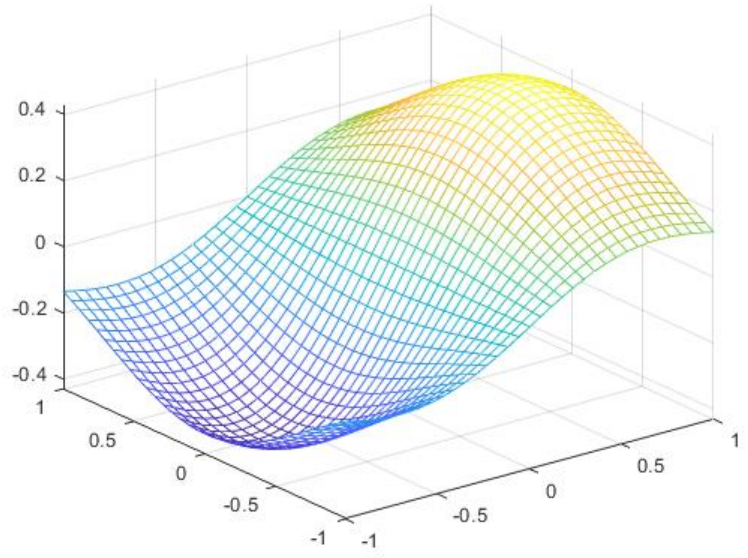
Verfahren:

- Grid search
- Random search
- Monte-Carlo
- Simplex-Verfahren
- Gradient descent
- Stochastic gradient descent
- ...

Gradient

Der Gradient einer Funktion $f(x)$:

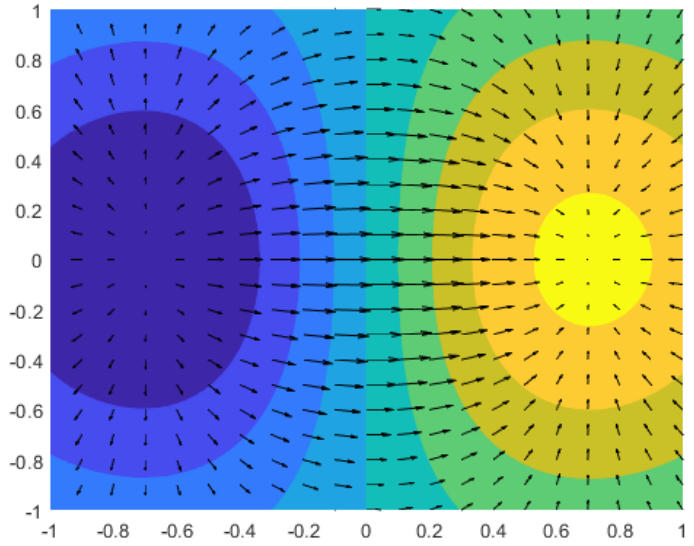
$$grad\ f(x) = \nabla f(x) = \frac{df}{dx} = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \cdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$



Die **Länge** des Gradienten gibt die **Steigung** der Funktion an diesem Punkt.

Die **Richtung** des Gradienten weist in **Richtung des steilsten Anstieges** an diesem Punkt.

An **Extrempunkten** sind alle **partielle Ableitungen** (Komponenten des Gradienten) gleich **Null**.



Gradient descent = Abstiegsverfahren

Die Suchrichtung entspricht dem steilsten Abstieg

$$\underline{d} = -\nabla f\left(\underline{x}_i\right)$$

Die Schrittweite wird berechnet wie folgt

$$\underline{s} = -\alpha \nabla f\left(\underline{x}_i\right)$$

Der nächste Suchpunkt wäre dann

$$\underline{x}_{i+1} = \underline{x}_i - \alpha \nabla f\left(\underline{x}_i\right)$$

α – Learning rate (Hyperparameter)

Startpunkt: Näherungswert

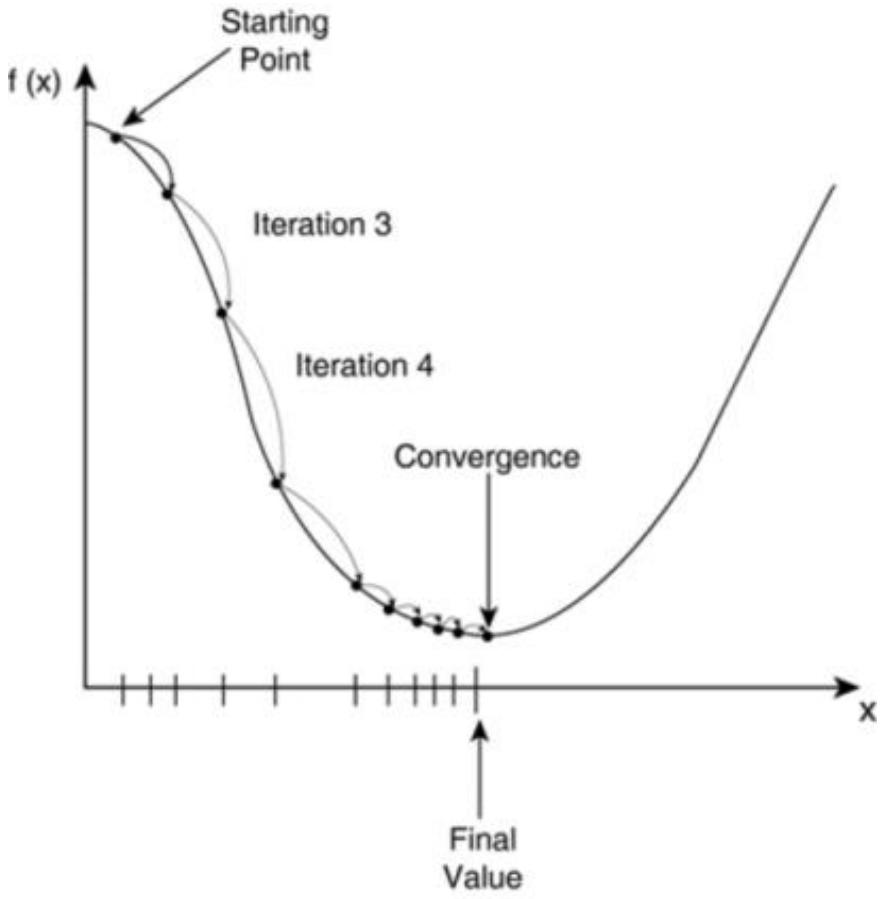


Bild: https://zkmablog.com/2017/02/11/was-ist-das-gradientenabstiegsverfahren/

Demonstration

Matlab-Livescript

Nichtlineare Regression

Nichtlineare Regression

- Transformation der nichtlineare Modellansätze in parameterlineare Modellansätze
  - $y = ae^{bx} \rightarrow \ln y = \ln a + bx$
  - $y = \frac{x}{ax+b} \rightarrow \frac{1}{y} = a + b\frac{1}{x}$

- Lösen des parameter-nichtlineare Problem

$$SSE(\underline{b}) = \min_{\underline{b}} \sum_{i=1}^{n} \left(Y - h(\underline{b}, X)\right)^2$$

$\Rightarrow$ nichtlineares Optimierungsproblem

$\Rightarrow$ Numerisches Lösungsverfahren

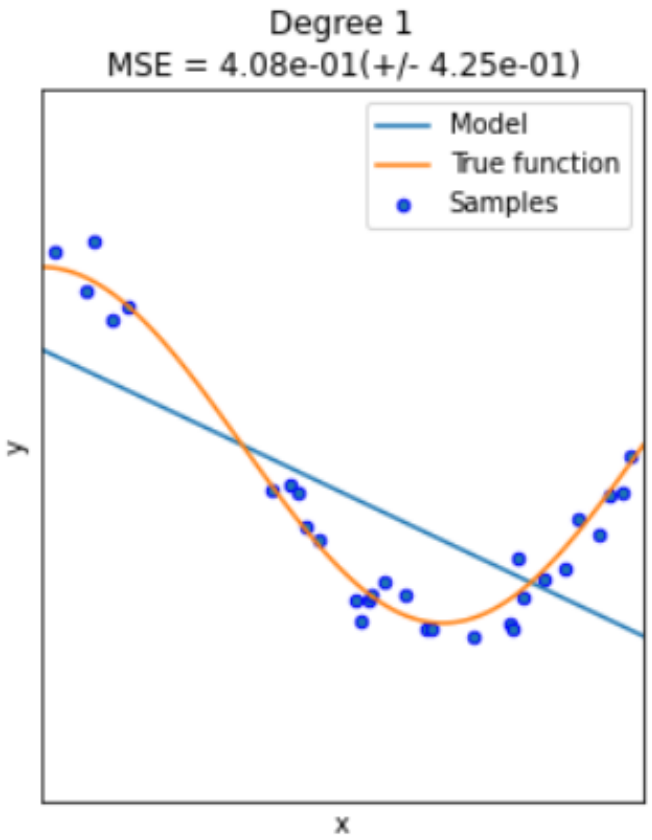
Evaluation des Modells

Parameterschätzung

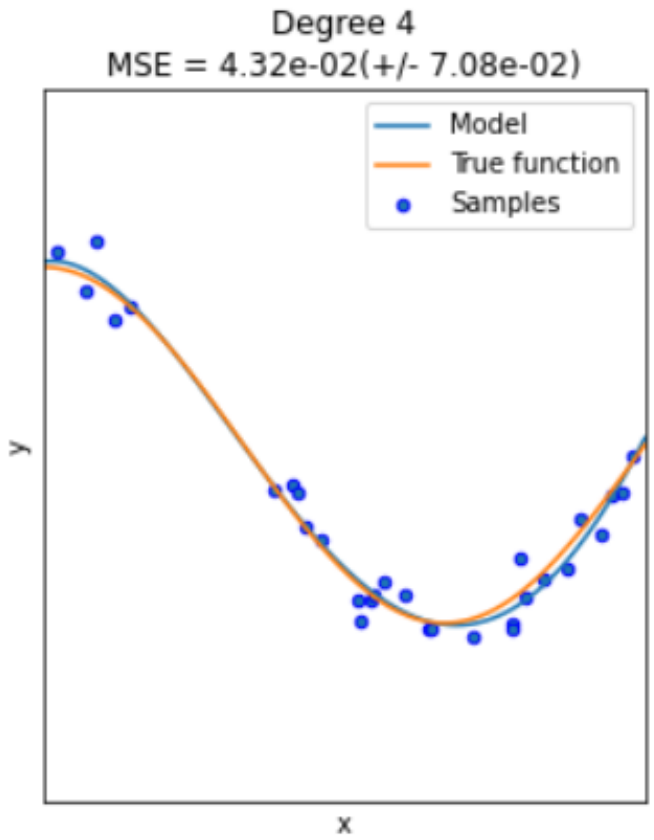
Im Anschluss an die Parameterschätzung sind folgende Aspekte zu untersuchen

- Bewertung der Güte des durch die Regressionsanalyse gefundenen Modells
  - Reststreuung s_r^2
  - Bestimmtheitsmaß B
- Adäquatheitstest für das Modell
- Signifikanztest für die Modellparameter
- Überprüfung der anfangs formulierten Voraussetzungen bezüglich der Verteilung der Störgröße (Reststreuungsanalyse)

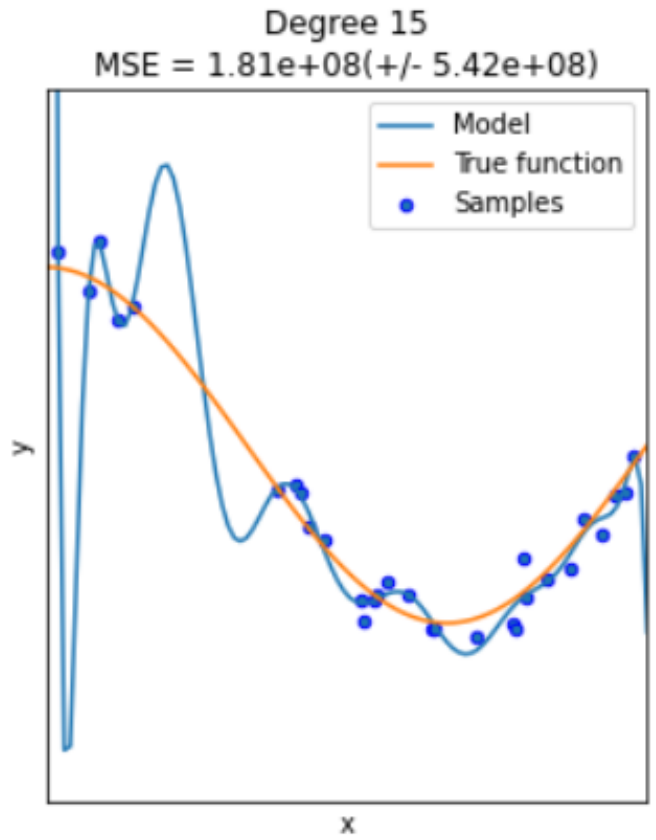
Adäquatheitstest



Underfitting



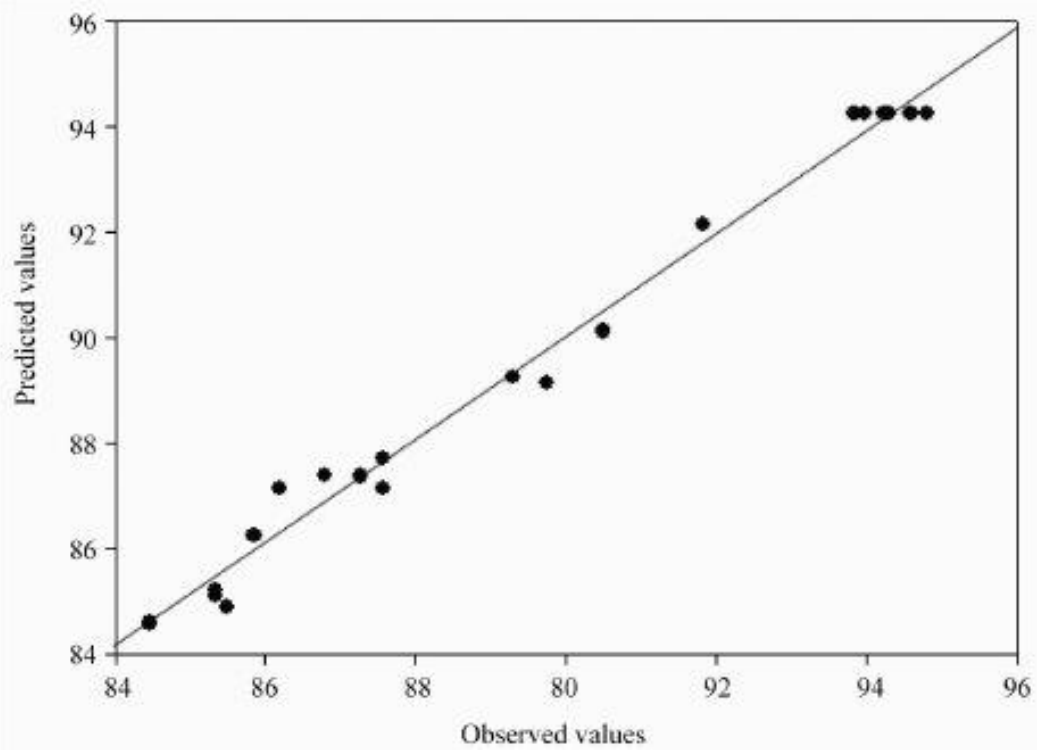
optimal



Overfitting

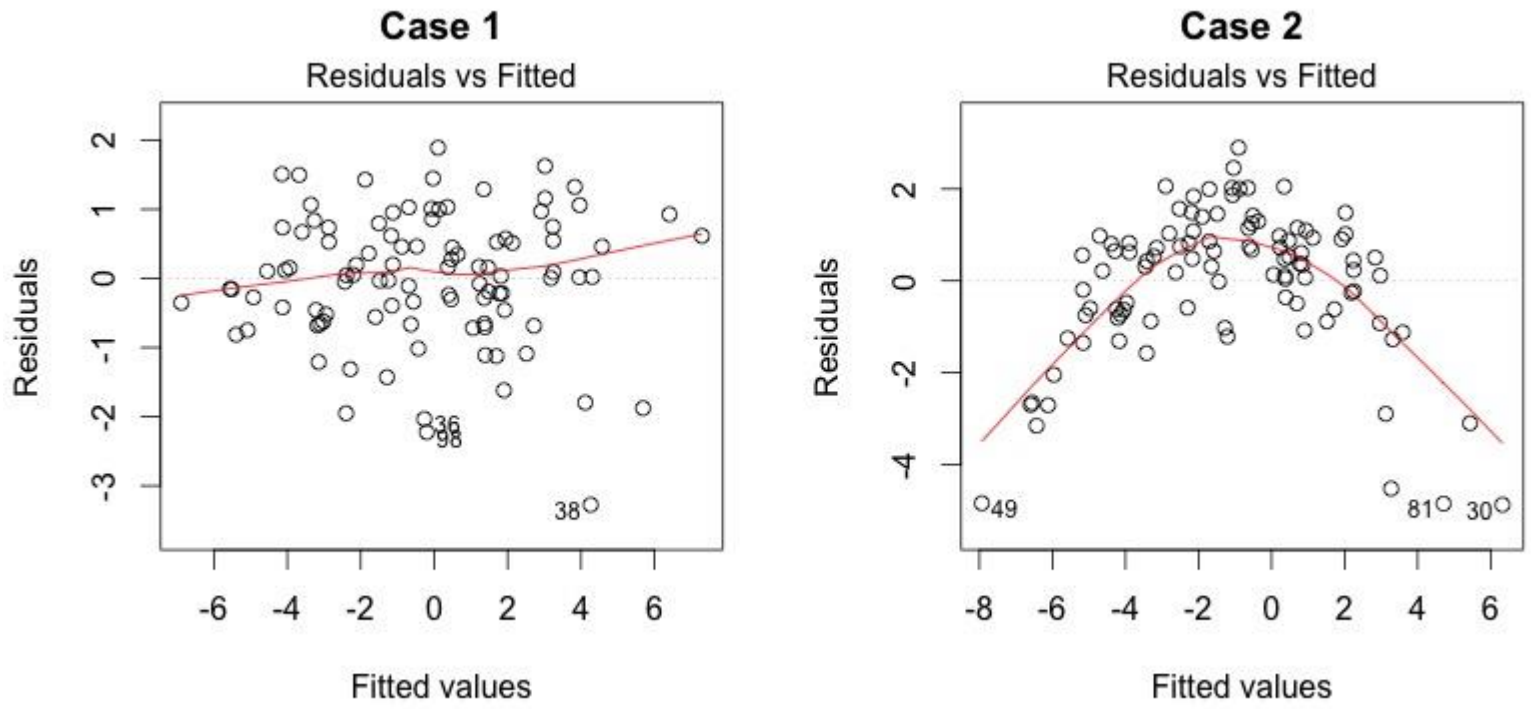
https://datascience.foundation/sciencewhitepaper/underfitting-and-overfitting-in-machine-learning

Parity plot



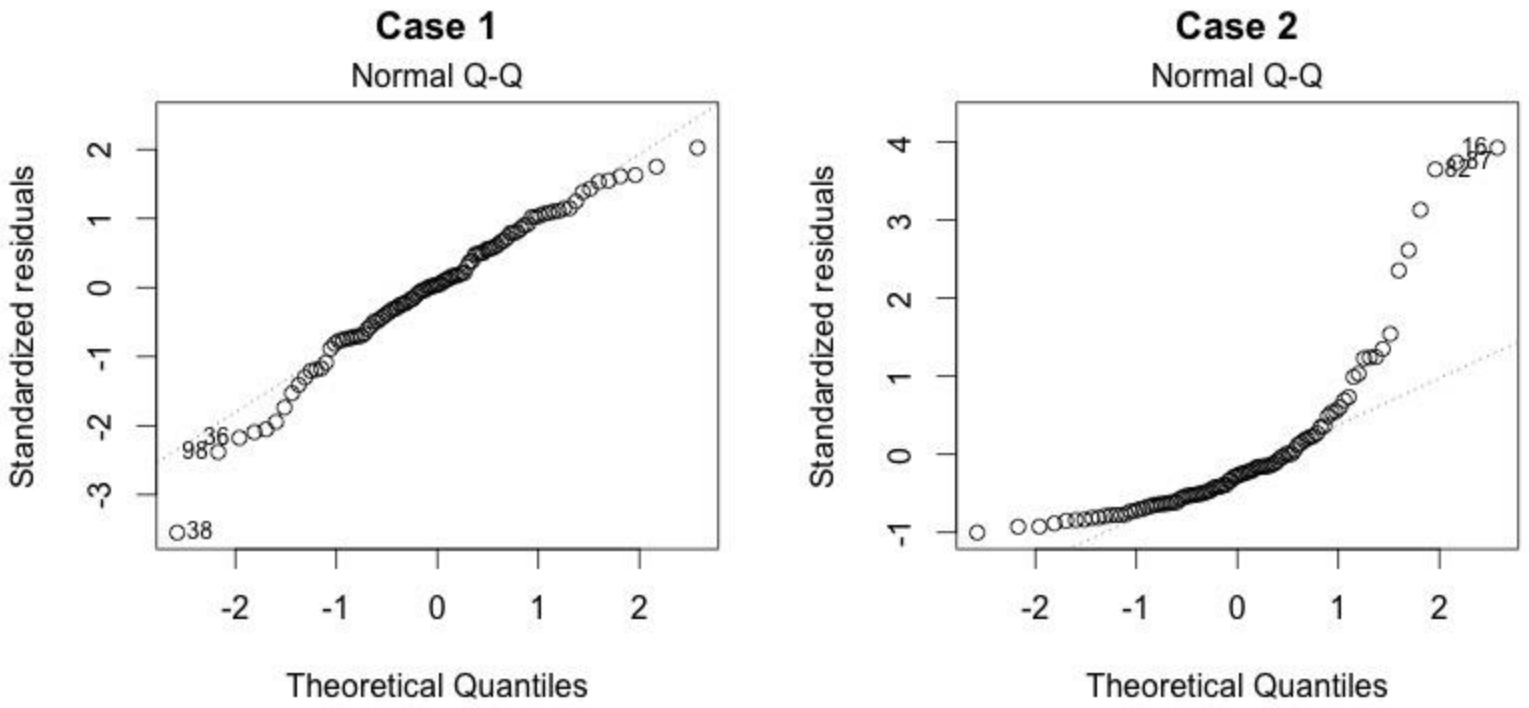
doi:10.4236/abb.2011.24028

Residuals vs Fitted



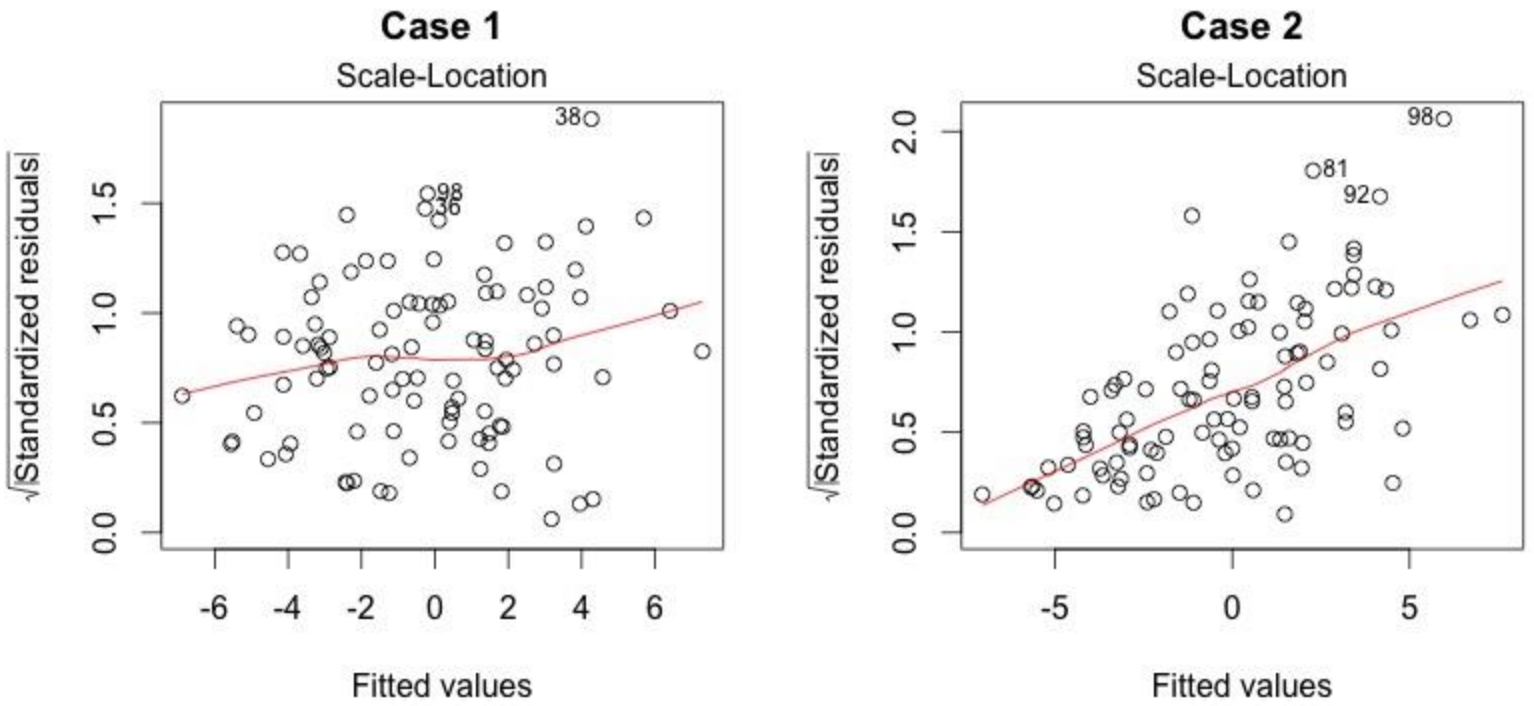
https://data.library.virginia.edu/diagnostic-plots/

Q-Q-Plot



https://data.library.virginia.edu/diagnostic-plots/

Scale-Location Plot



https://data.library.virginia.edu/diagnostic-plots/

Gesamttest auf Signifikanz

Fragestellung: hat **mindestens eine der Prädiktorvariablen** einen statistisch signifikanten Einfluß auf die Zielvariable

Null-Hypothese: $H_0: b_j = 0$, Alternative: $H_1: b_j \neq 0$ für alle $j \in \{1,2 \dots k\}$

Ermittlung mit F-Test:

$$F_n = \frac{S^2_{reg}}{S^2_{y|x}} > F_{k;n-k-1;1-\alpha}$$ oder p-Wert größer als α

$S^2_{reg} = \frac{1}{k}\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2$ - die Varianz der Regression und

$S^2_{y|x} = \frac{1}{n-k-1}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ - die Residualvarianz

$n-k-1$ – die Freiheitsgrade, n – Anzahl von Beobachtungen, k – Anzahl der Inputvariablen

Test für die Signifikanz einzelner Merkmale

Fragestellung: hat **eine konkrete Prädiktorvariable** einen statistisch signifikanten Einfluß auf die Zielvariable

Null-Hypothese: $H_0: b_j = 0$, Alternative: $H_1: b_j \neq 0$

Ermittlung mit t-Test:

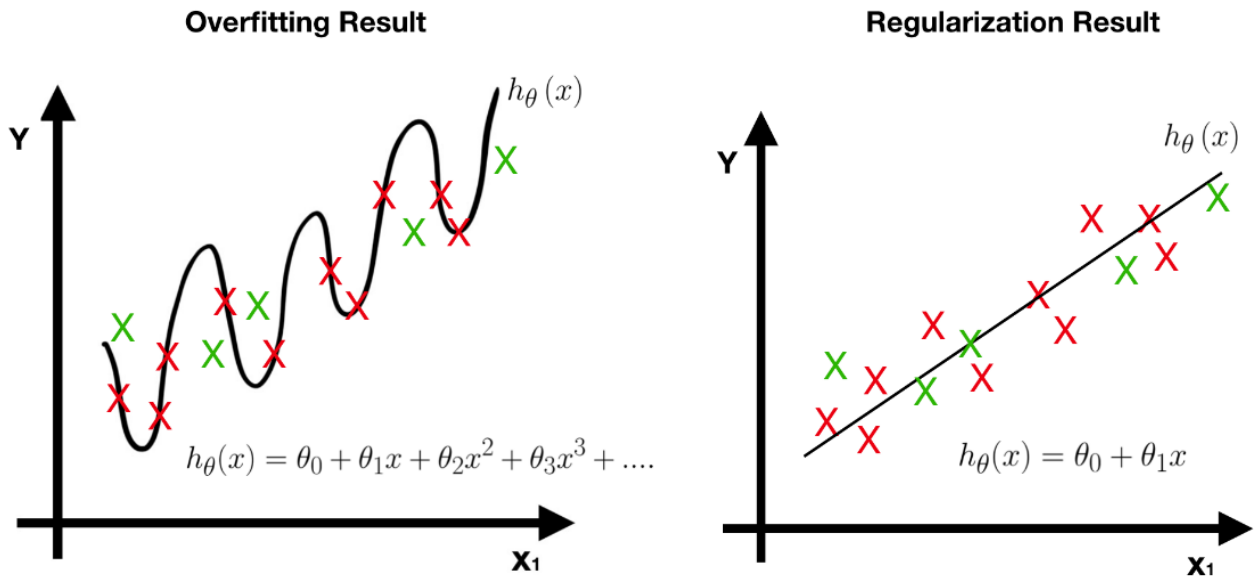
$$T_n = \left|\frac{\hat{b}_j}{\hat{s}_{bj}}\right| > t_{n-k-1;1-\alpha/2}$$ oder p-Werte größer als α

wo $\hat{s}_{bj}$ der Standardfehler von $\hat{b}_j$

i	$\hat{b}_i$	$\hat{s}_{b_i}$	p-Wert
0	5.54	2.62	
1	0.39	0.14	0.008
2	0.23	0.09	0.020
3	0.001	0.12	0.994

Regularisierung

Regularisierung



Regularization ist ein Ansatz zur künstlichen Einschränkung von wenig relevanten Modellelementen (z.B. Input-Features oder Neuronen), z.B. L2-Regularization

Regularization Term

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n}\theta_j{}^2$$

start at θ_1

Regularization Parameter

Bilder: https://medium.com/@qempsil0914/courseras-machine-learning-notes-week3-overfitting-and-regularization-partii-3e3f3f36a287

Optimierung von Hyperparametern

Nachdem der Typ bzw. die Architektur des Modells definiert wurde, sollen **Modell- und Trainingsparameter** optimiert (andere Ebene der Optimierung als beim Modell-Training) werden.

Exemplarische Parameter:

- Ordnung des Regressors
- Art des Kernels im SVM
- Anzahl von Schichten und Neuronen in NN

- Beiwerte und Art der Regularisierung
- Learning-Rate des Abstiegsverfahrens

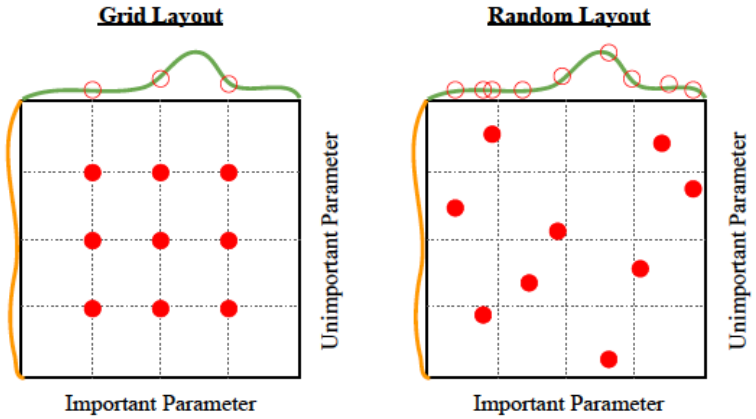
Direkte Verfahrens (Parallelisierung):

- **Grid-Search**

Alle mögliche Kombinationen von Parametern werden evaluiert

- **Random-Search**

Zufällig ausgewählte Kombination von Parametern



Iterativ: **Bayesian, Gradient descent** usw.

Bild: https://srdas.github.io/DLBook/DL_images/HPO1.png

Validierung und Testen des Modells

Model testing

Modelltests dienen der Bewertung der Modellgüte hinsichtlich der Qualitätsmetriken (**nicht unbedingt dasselbe wie die Zielfunktion**).

Qualitätsmetriken: MSE, MAE, MAPE

Datenaufteilung:

- Training (70-80%) – Optimierung des Modells
- Validation (15-20%) – **unabhängige Bewertung** der Hyperparameteroptimierung
- Test (15%) – **unabhängige Bewertung** des resultierenden Modells (inkl. Hyperparameter-Tuning)

k-fold Cross Validation:

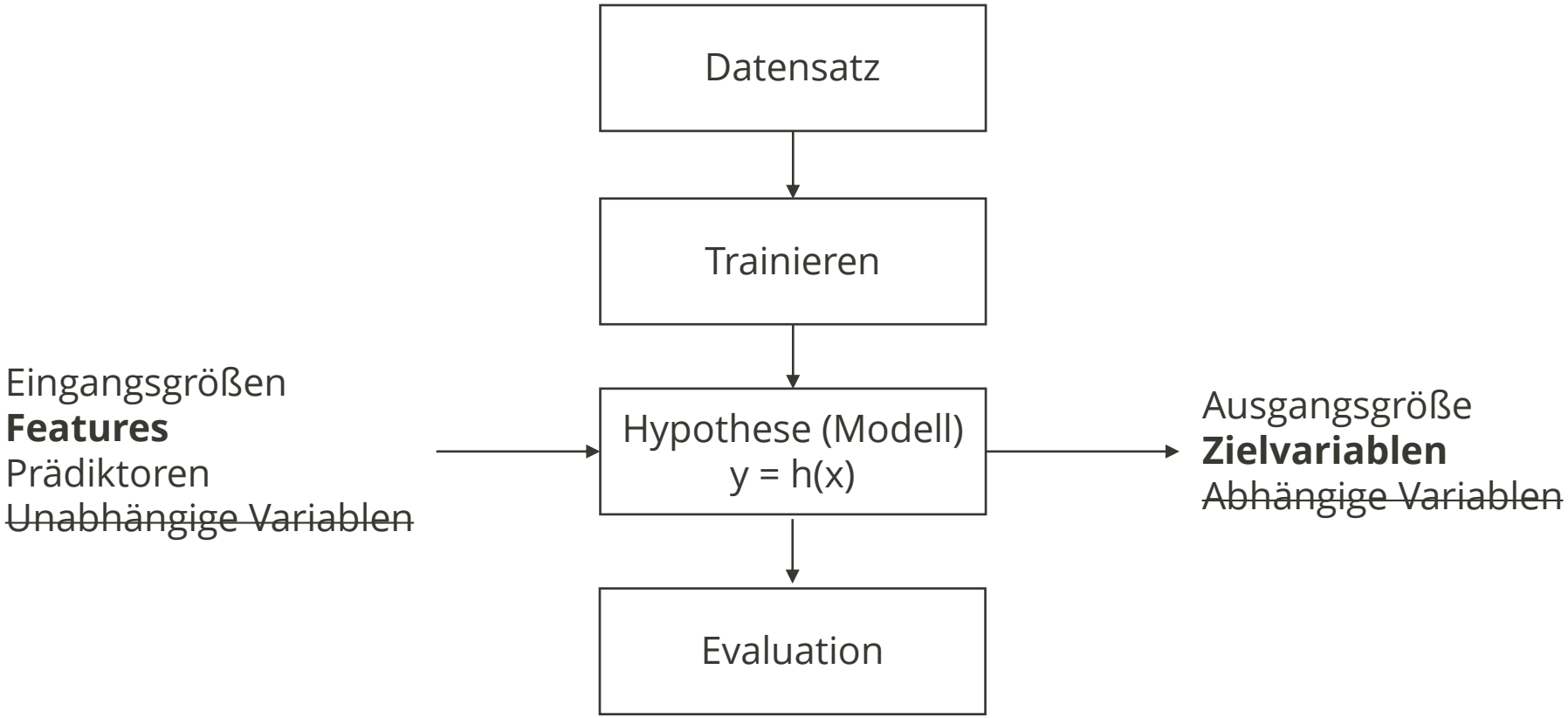
- Aufteilung des Datensatzes auf k Anteilen, die für Evaluation des Modells verwendet werden

Vorteil: kein separater Validerungsdatensatz erforderlich

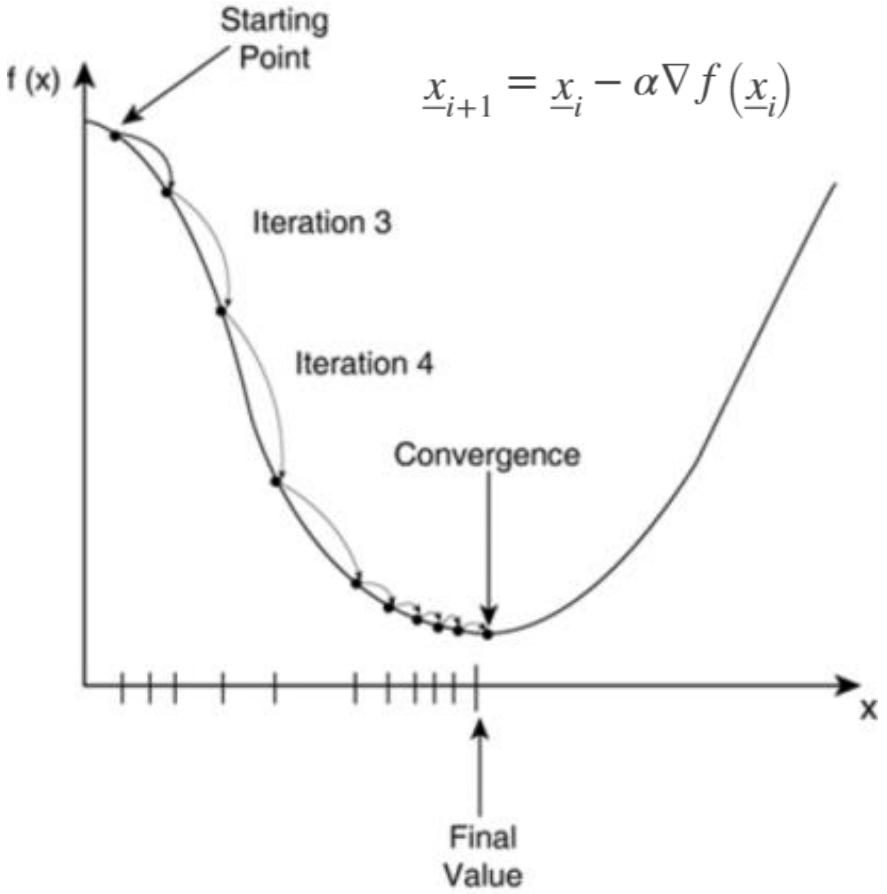
https://de.wikipedia.org/wiki/Kreuzvalidierungsverfahren#/media/Datei:KfoldCV.gif

Zusammenfassung

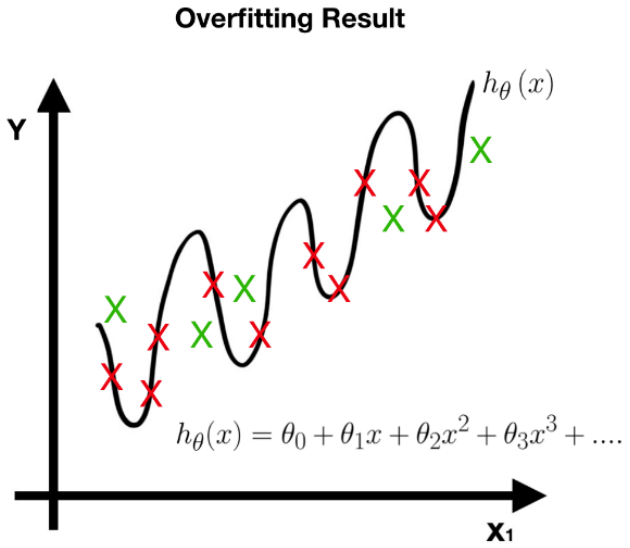
Regressionsanalyse

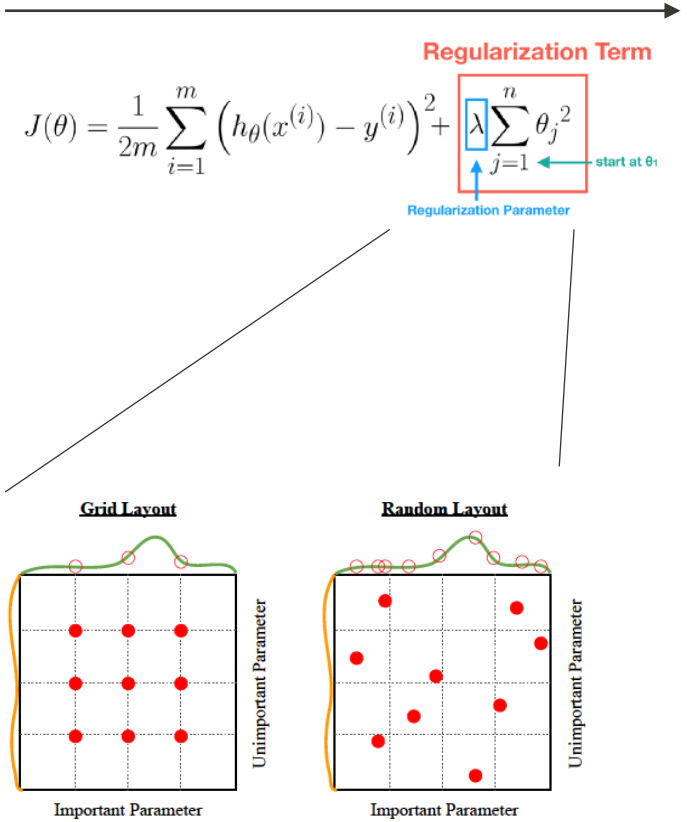


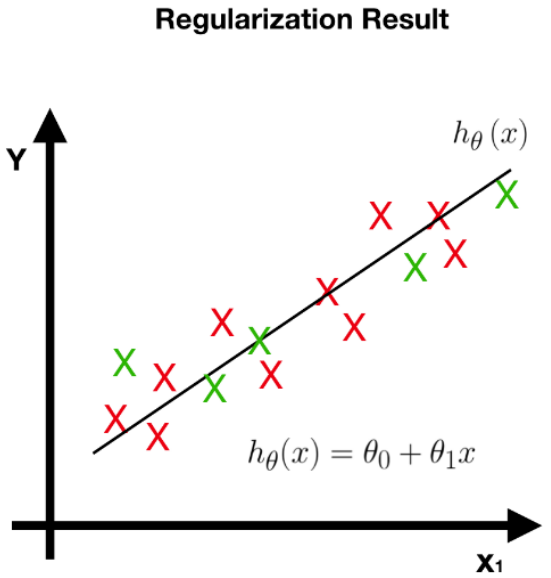
Numerische Optimierungsverfahren



Regularisierung und HPO

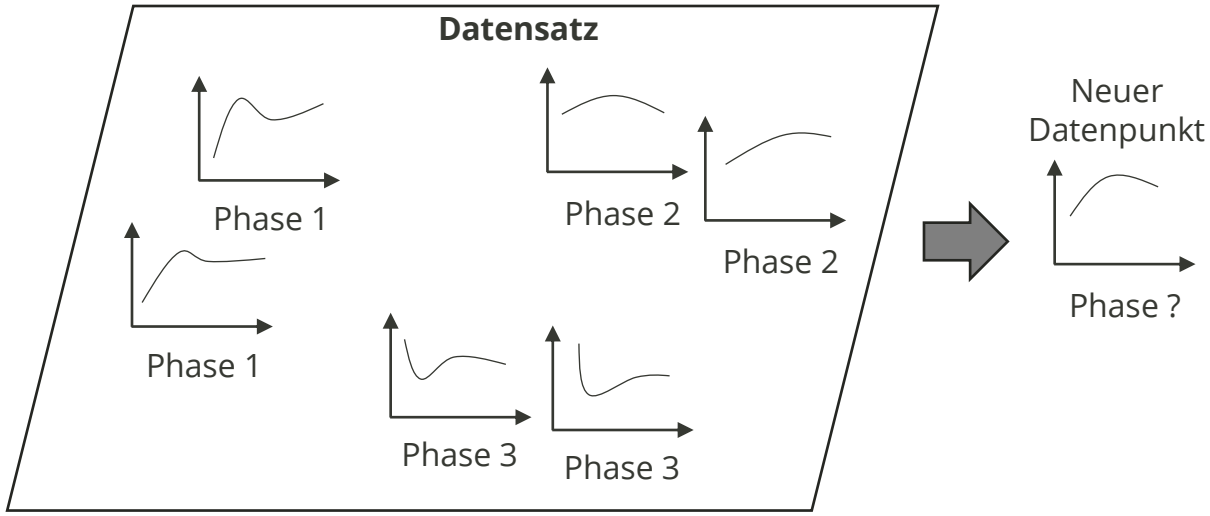


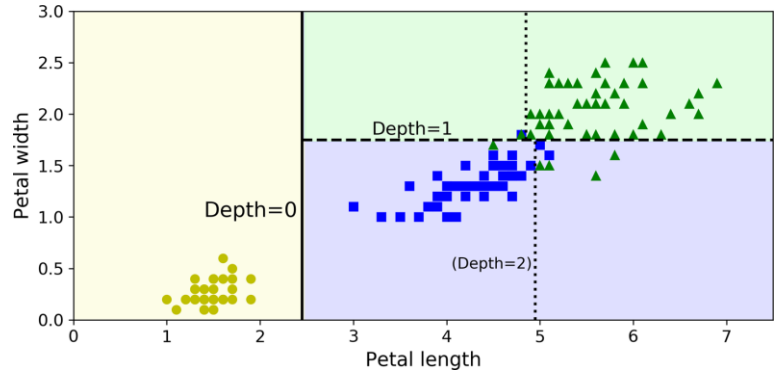


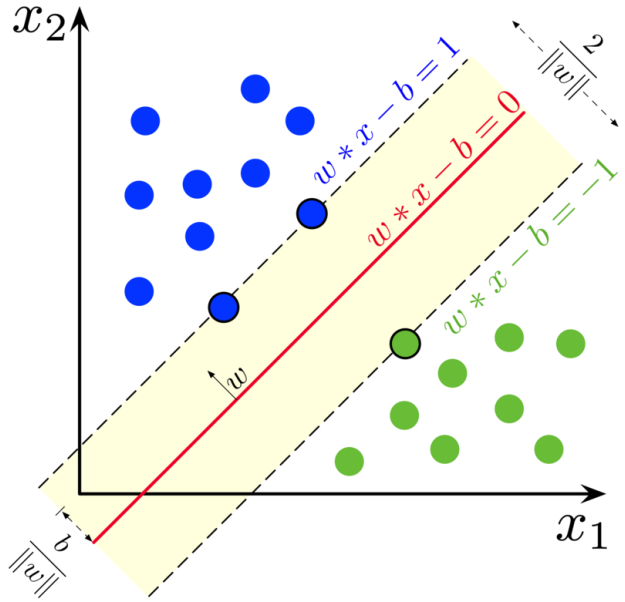


Bilder: https://medium.com/@qempsil0914/courseras-machine-learning-notes-week3-overfitting-and-regularization-partii-3e3f3f36a287

Nächste Vorlesung







Adaptiert aus A. Géron, Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, 2019.

PROCESS CONTROL SYSTEMS
PROCESS SYSTEMS ENGINEERING

Dr. rer. nat. Valentin Khaydarov
Email: valentin.khaydarov@tu-dresden.de
Telefon: 0351 463 33387

Vielen Dank für Ihre Aufmerksamkeit!