Einführung in Web- und Data-Science

Prof. Dr. Ralf Möller

Universität zu Lübeck

Institut für Informationssysteme

Tanya Braun (Übungen)

Teilnehmerkreis und Voraussetzungen

Studiengänge

- Bachelor **Informatik**
- Bachelor **IT-Sicherheit**
- Bachelor **Mathematik in Medizin und Lebenswissenschaften**
- Bachelor **Medizinische Informatik**
- Bachelor **Medieninformatik**
- Bachelor **Robotik und Autonome Systeme**

Voraussetzungen

- Keine

Organisatorisches: Übungen

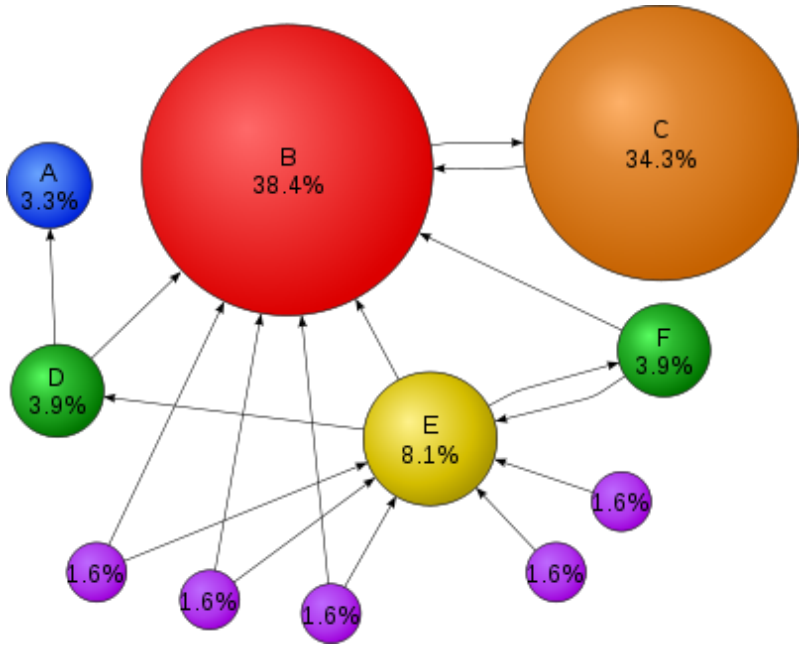
- **Vorlesung**: Donnerstags, ab dem 19. Oktober 2017
- **Übungen**: Mittwochs 13-14 Uhr, ab 25.10.
  Anmeldung über Moodle nach dieser Veranstaltung
- **Übungsaufgaben** stehen jeweils nach der Vorlesung ca. ab 18 Uhr über Moodle bereit (erstes Übungsblatt erscheint am 19.10.2017)
- **Abgabe der Lösungen** erfolgt bis Montag 12 Uhr in der IFIS-Teeküche (jeweils in der zweiten Woche nach der Ausgabe in den Kasten für die jeweilige Übungsgruppe)
- Aufgaben sollen in einer **2-er Gruppe** bearbeitet werden (also bitte Name(n), Matrikelnummer(n) und Übungsgruppennummer vermerken)
- In den Übungen am Mittwoch wird der Übungszettel besprochen, dessen Lösungen bis zum jeweils vorigen Montag abgegeben werden, und auch Fragen zum jeweils neuen Übungszettel geklärt.

Organisatorisches: Prüfung

- Die **Eintragung in den Kurs** und in eine Übungsgruppe ist **Voraussetzung**, um an dem Modul Non-Standard-Datenbanken teilnehmen zu können
- Am Ende des Semesters findet eine **Klausur** statt
- **Voraussetzung** zur Teilnahme an der Klausur sind mindestens **50% der gesamtmöglichen Punkte aller Übungszettel**

Web und Data Science

- Web Science
  - Analyse von Strukturen im Web (Mensch und Computer)
  - Formalisierung durch große Graphstrukturen und entsprechende Entscheidungsprobleme über Graphen
  - Beispiel: Pagerank (Bewertung von Webseiten)



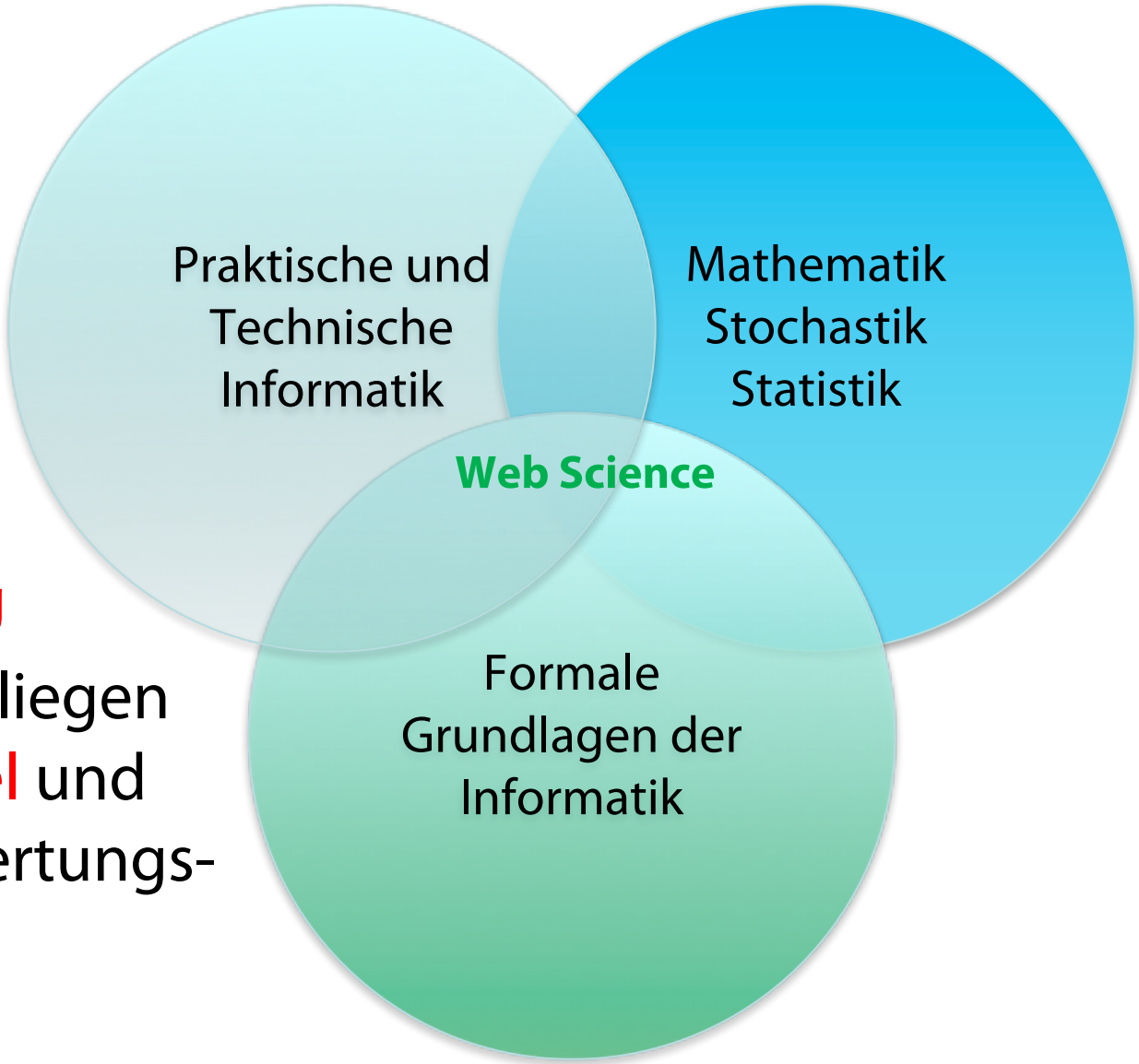
Zufallssurfer-Modell:

Größe der Kreise in etwa proportional der relativen Häufigkeit, mit der sich ein Surfer auf einer Seite befindet

[Wikipedia]

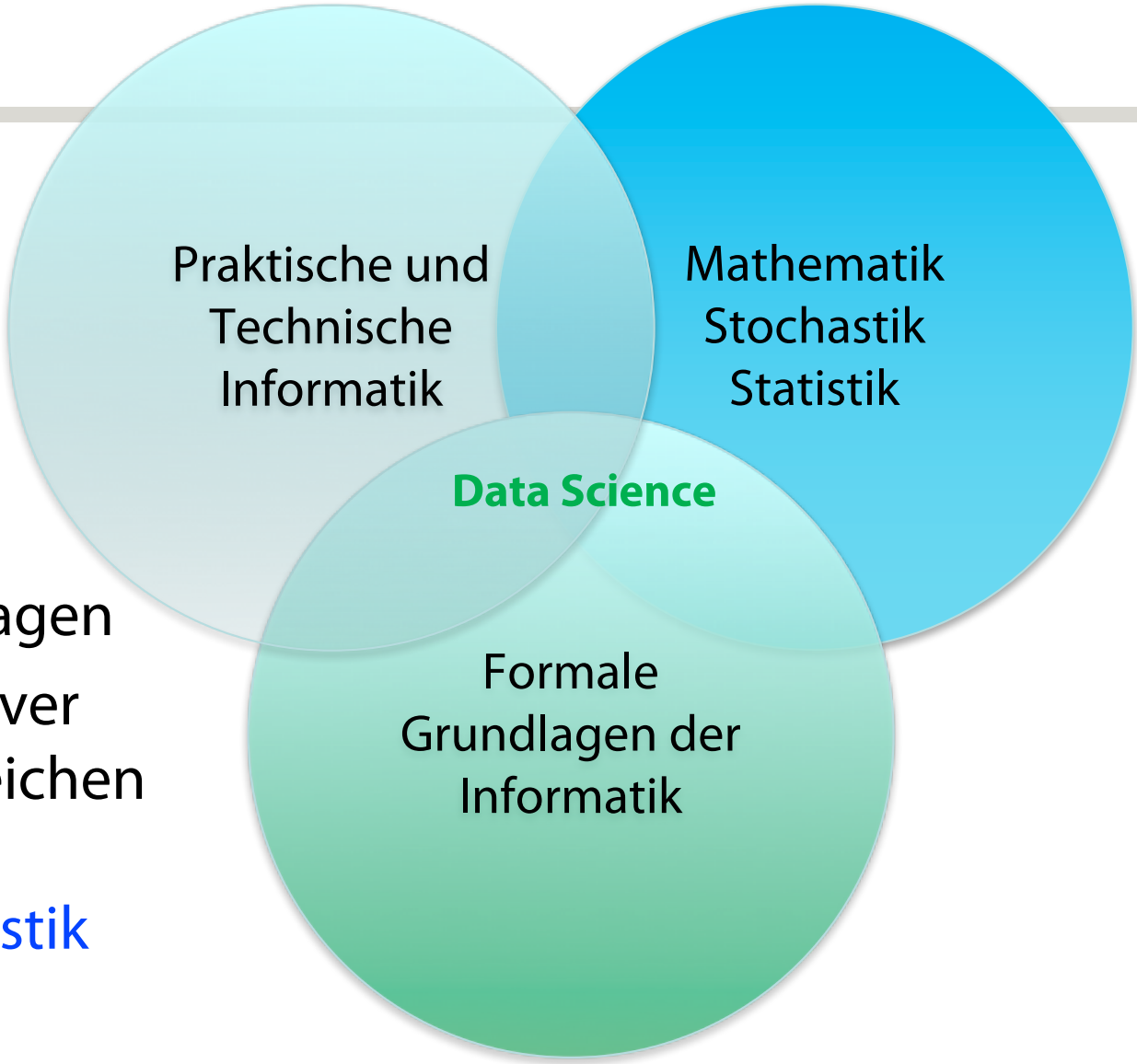
Herausforderungen für die Informatik

- Graphstrukturen extrem groß
- Verfahren zur Lösung von Entscheidungs-problemen extrem aufwendig
- Graphdaten unterliegen ständigem Wandel und so auch die Auswertungs-ergebnisse



Data Science

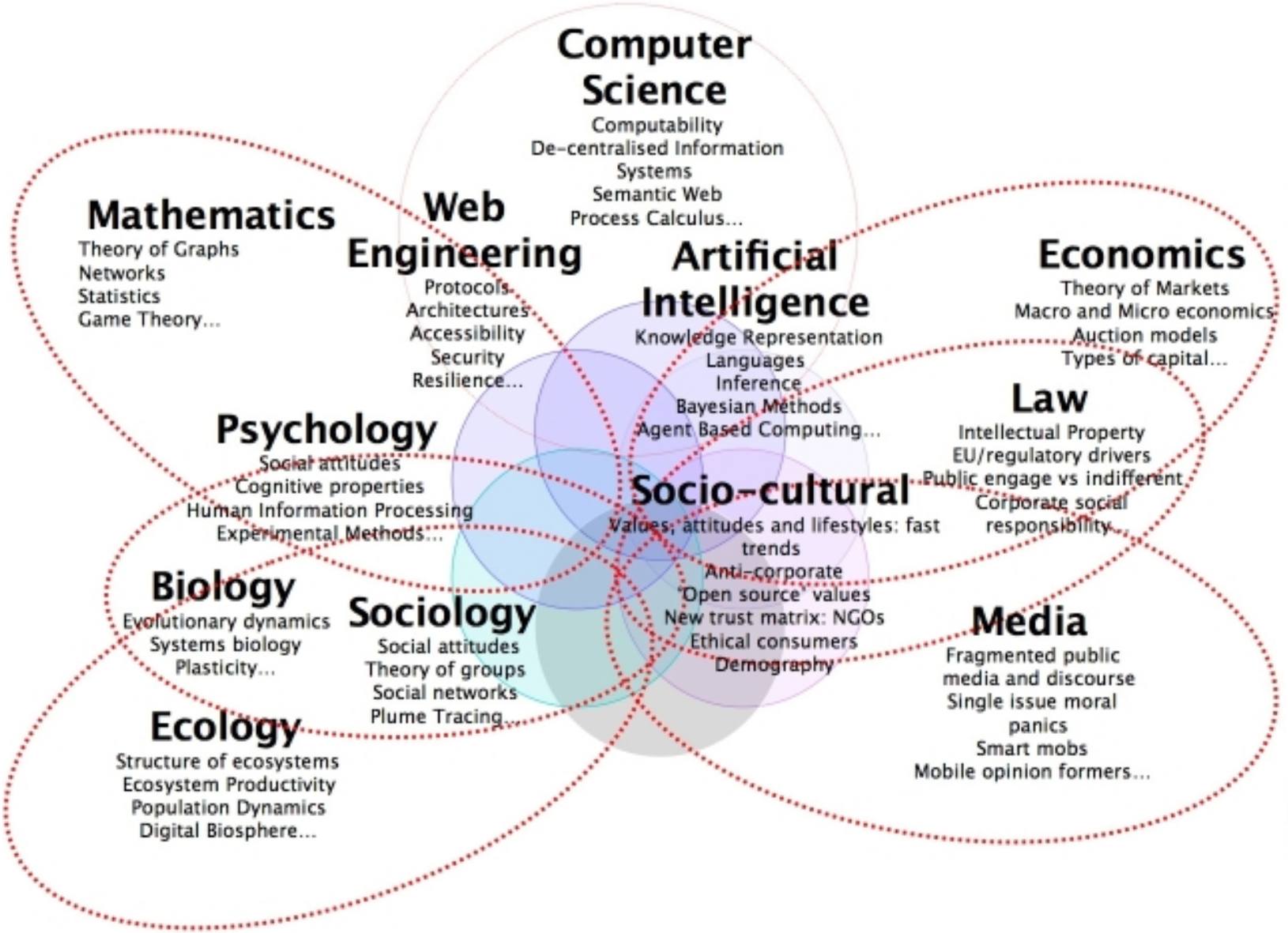
- Extraktion von Wissen aus Daten (u.a. Graphdaten)
- Begriff schon vor 50 Jahren für Informatik vorgeschlagen
- Entwicklung innovativer Konzepte in den Bereichen Logik, Datenbanken und Stochastik / Statistik (Datenanalyse und Wissensentdeckung)
- Verwendung von LADS und Analysis



Herausforderungen für die Informatik

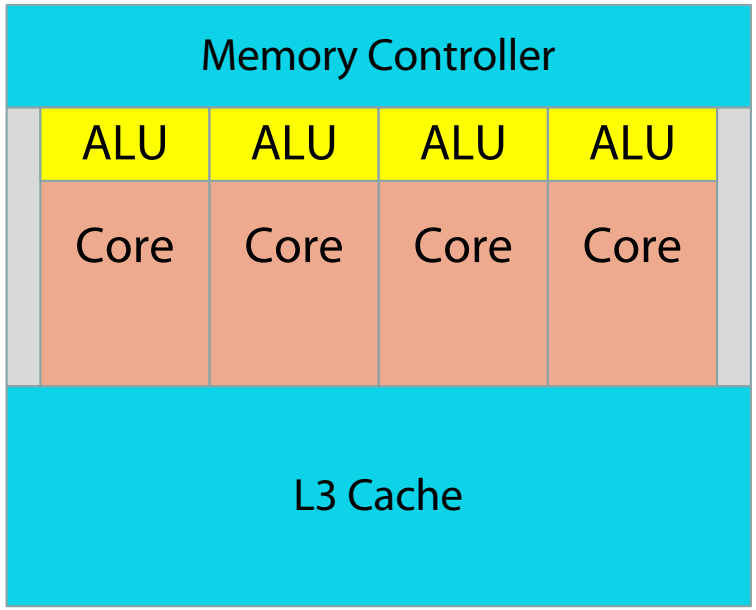
- Große Datenbestände
  - Speicher und Zugriffstechnologie
- Starker Zuwachs an Daten, hohe Dynamik
  - Hohe Datenraten und Echtzeitanforderungen
- Heterogene Datenbestände
  - Verteiltes Datenmanagement
  - Datenintegration
- Robuste Modelle der menschlichen Interpretationsvorgänge (Kognition) notwendig für Auswertung und Präsentation von Ausgaben
  - Verarbeitung von Text- und Graphikdaten nicht nur syntaktisch
  - Sprach und Videodatenverarbeitung in den Kinderschuhen
- …

Web und Data Science: Sozio-technische Sicht

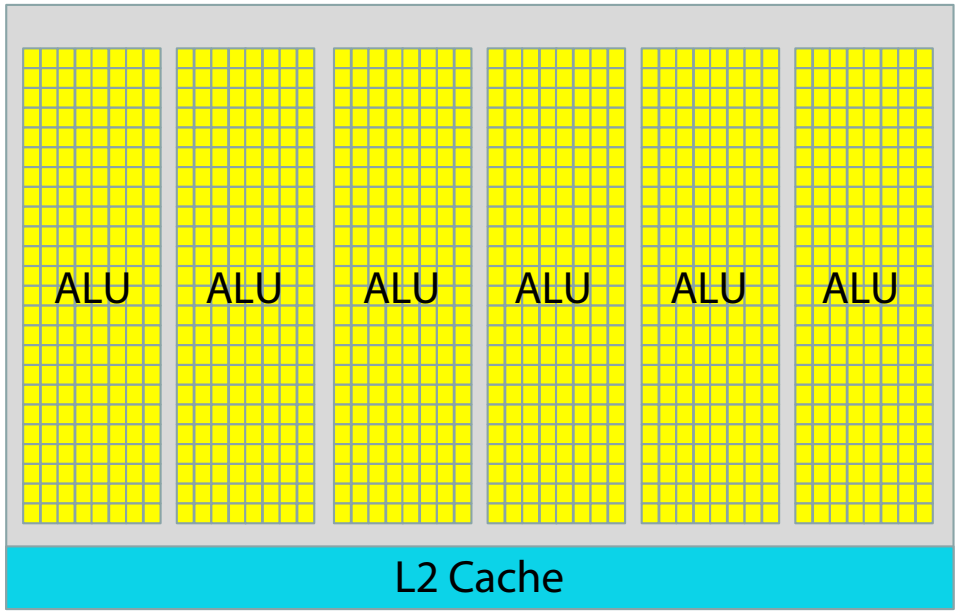


Hardware-Sicht: CPU vs GPU

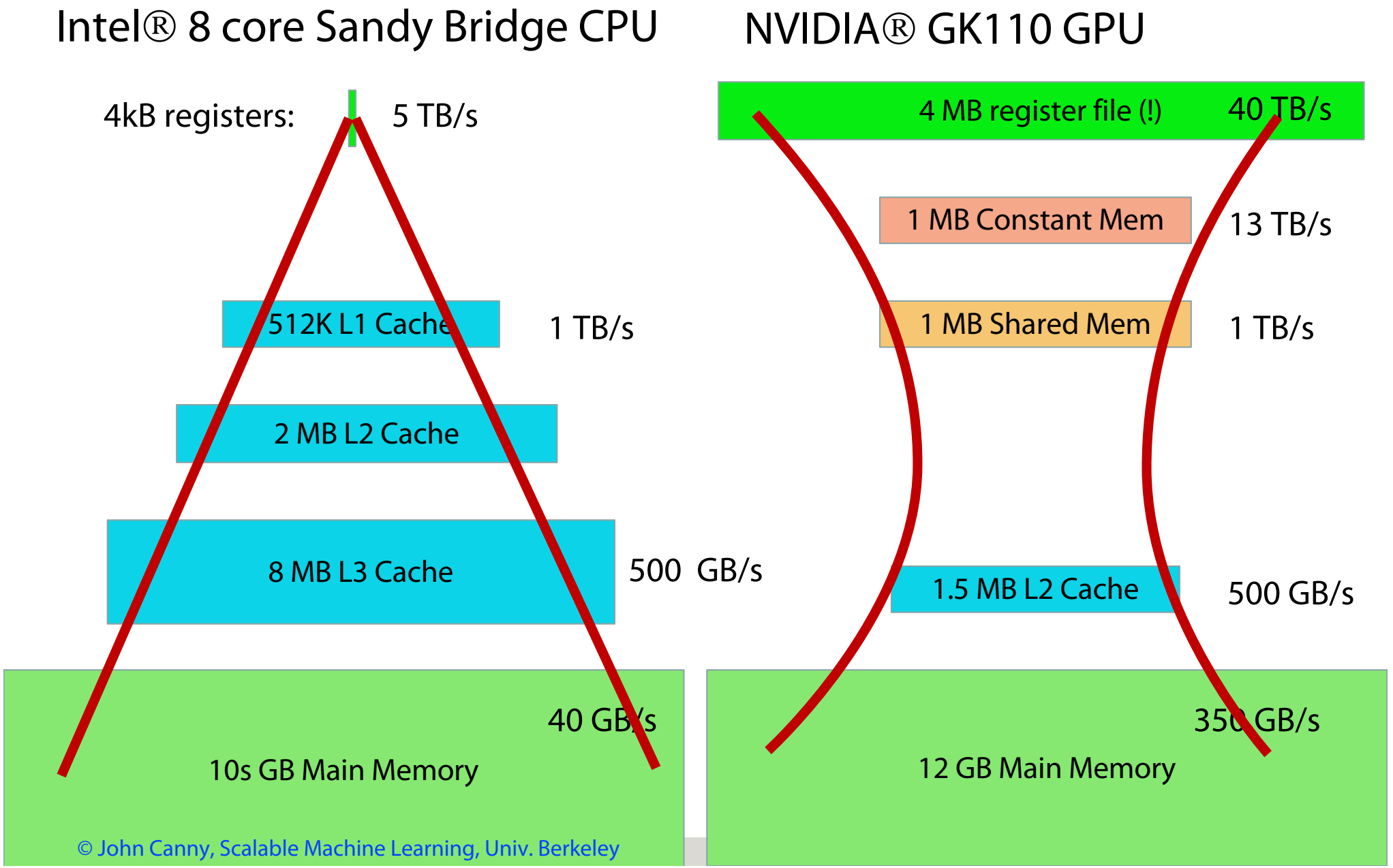
Intel® CPU



NVIDIA® GPU

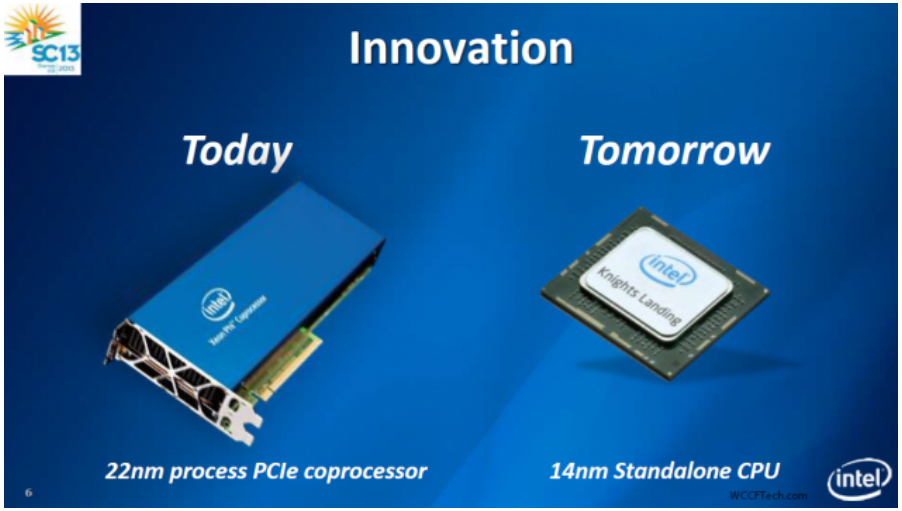


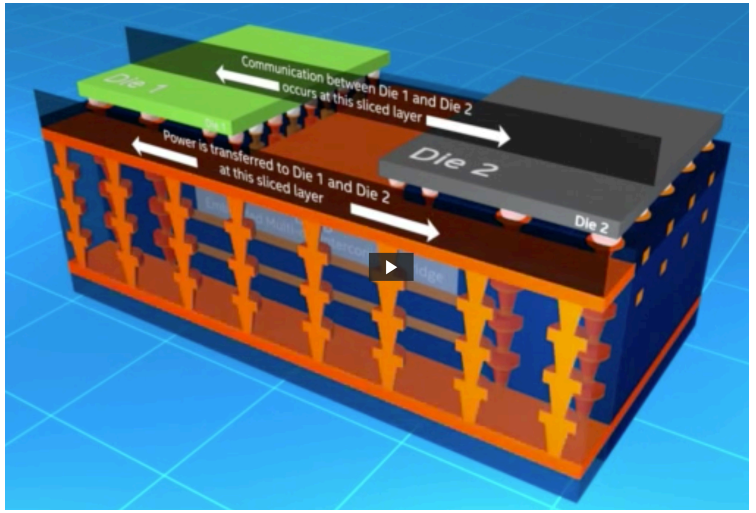
Hardware-Sicht: CPU vs GPU



Neue Hardwarestrukturen

- Heute: CPU (mit n Kernen) und externem Co-Prozessor (PCI-Bus)



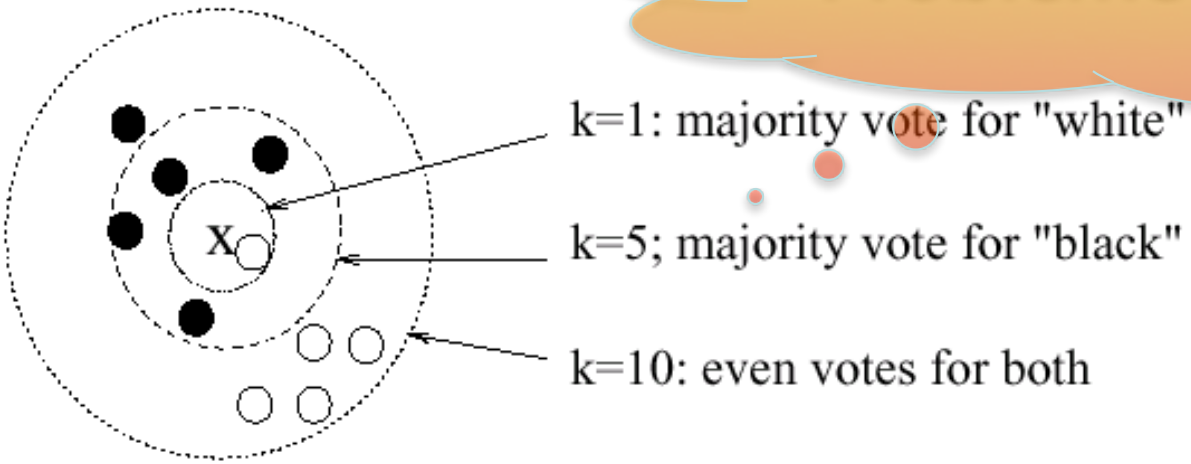


- Morgen: CPU und GPU integriert
  - z.B. Intel "Knight's Landing" Prozessor
- Morgen: CPU und FPGA eng verzahnt
  - Embedded Multi Die Interconnect Bridge
  - Dynamisch rekonfigurierbare Datenfluss-Hardware für spezielle Aufgaben

Speicheranforderungen für Anwendungen

- Annahme: Gegeben viele Datenpunkte
  - Beispielmerkmale: (x, y, Farbe), Farbe $\in$ { white, black }
- Anfrage: Datenpunkt ohne Wert für bestimmtes Merkmal
  - Beispiel: Merkmal Farbe ohne Wert
- Anfragebeantwortung (Klassifikation des Anfragepunkts): Mehrheitsvotum der k-nächsten Nachbarn (kNN-Verfahren)

Probleme erkennbar?



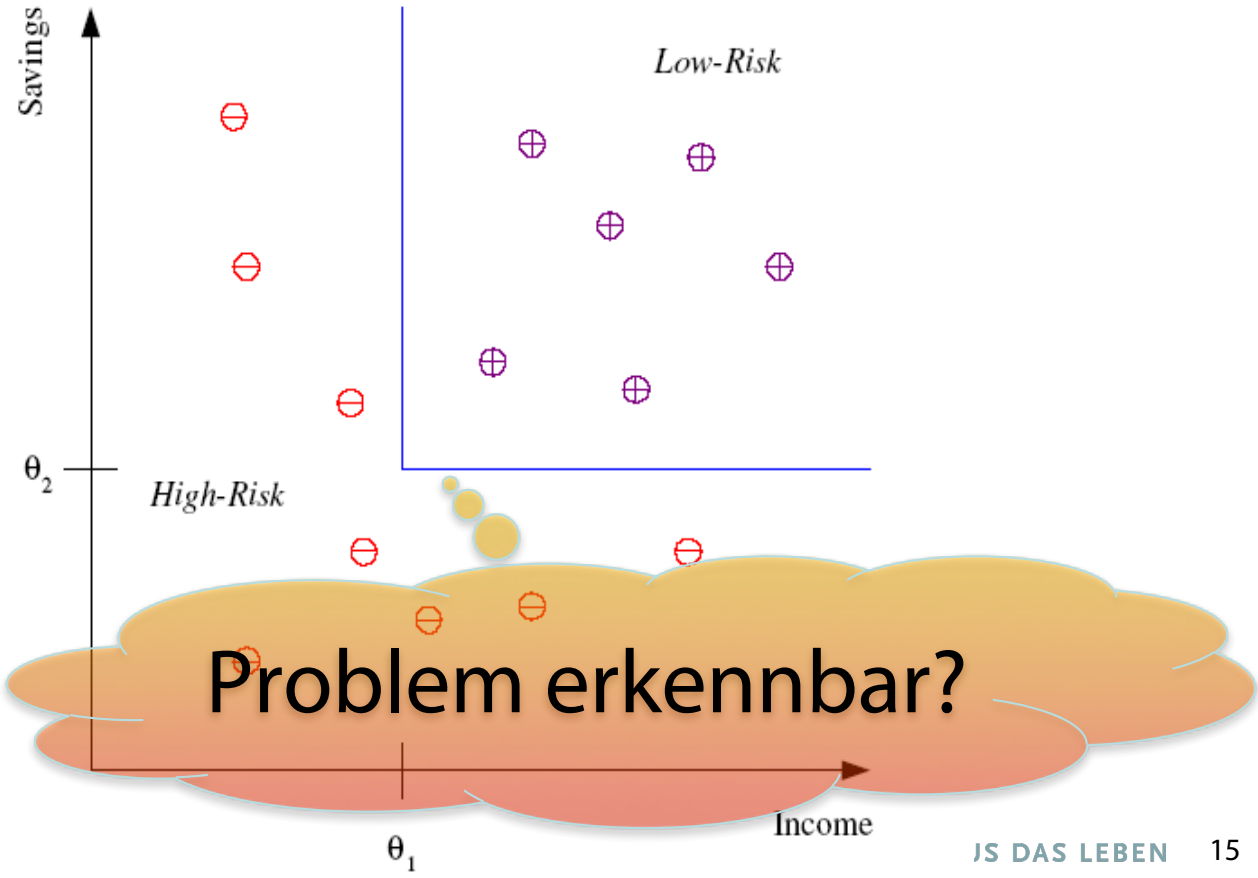
Fix, E., Hodges, J.L. Discriminatory analysis, nonparametric discrimination: Consistency properties. Technical Report 4, USAF School of Aviation Medicine, Randolph Field, Texas, **1951**

Probleme mit kNN

- Klassifikationsergebnis stark von k abhängig
- Hoher Speicherbedarf
- Effizienter Zugriff auf "Nachbarn" erfordert weitere Maßnahmen (noch mehr Speicherbedarf)

Generalisierung von Daten möglich?

- Repräsentation der Daten durch Parameter
  - Wenn (Einkommen > θ_1 $\wedge$ Ersparnisse > θ_2), dann kreditwürdig ($\oplus$), sonst nicht ($\ominus$)
- Nur 2 Parameter nötig: (θ_1, θ_2)
- Modell fordert geringen Speicher



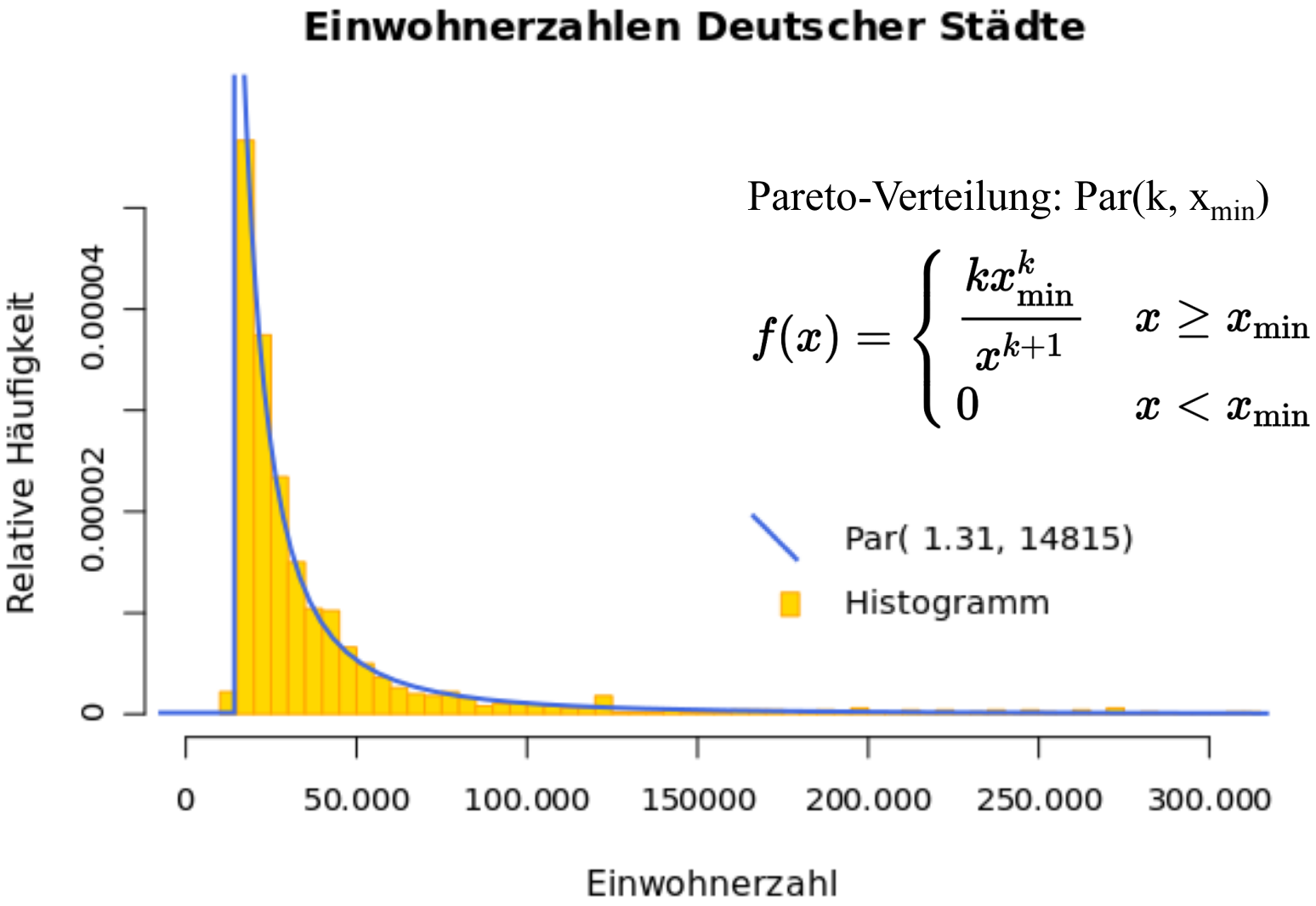
Beispiel

- Anzahl von Städten mit bestimmten Einwohnerzahlen schätzen

- Daten: Liste von Einwohnerzahlen (einige kommen mehrfach vor)

- Explizites Modell: Zählerfeld aufbauen
  - Zählerfeld ggf. sehr groß
- Implizites Modell: Potenzgesetz $y=ax^b$
  - a und b bestimmen
  - Aufwendiges Optimierungsproblem lösen

Begriff der "Verteilung"



Pareto-Verteilung: Par(k, x_{min})

$$f(x) = \begin{cases} \frac{k x_{\min}^k}{x^{k+1}} & x \geq x_{\min} \\ 0 & x < x_{\min} \end{cases}$$

[Wikipedia]

Rechnen auf realistischen Daten

Unterstützung von **großen Modellen** durch Cluster-basiertes maschinelles Lernen

Daten Parallelismus: Daten auf Clusterknoten aufgeteilt, jeder Knoten hat lokale Kopie des geteilten Modells

Modell-Parallelismus: Modell auf verschiedene Knoten verteilt. Keiner verarbeitet gesamte Daten,

In beiden Fällen gilt:
Knoten müssen kommunizieren,
um Modell zu bestimmen

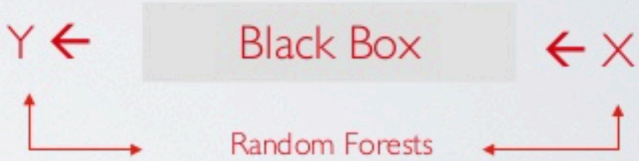
Data Models vs. Algorithmic Models

Data Modeling

vs.

Algorithmic Modeling

$Y \leftarrow F(X, \text{random noise, parameters})$



We understand the world

How well 'my data model' works
Statisticians, Data Analysts, Data Miners
Linear Regression
Logistic Regression
Known Distributions
Confidence Intervals
Predictor Variables & Goodness of Fit

We don't understand the world

The world produces data in a black-box
Data Scientists
Machine Learning, AI & Neural Nets
Random Forests, SVM, GBT
Unknown Multivariate Distributions
Iterative
Predictive Accuracy

"Statistical Modeling: The Two Cultures" Leo Breiman, 2001

Data Science London

Über den Kurs

- Repräsentationssprachen
  - Modellierungsansatz
  - Entscheidungs- und Berechnungsprobleme
  - Verfahren zu deren Lösung
- Gewinnung von Modellen aus Daten als Optimierungsproblem (Algorithm. Datenanalyse, Data Mining, Maschinelles Lernen)
  - Überwachtes Lernen,
  - Unüberwachtes Lernen
    - Deep Learning, Transduktives Lernen, Reinforcement-Lernen
  - Transfer-Lernen
- Verwendung von Modellen in Anwendungen
  - Prädikation, Forensik, Verstehen der Realität (Science-Aspekt!)
  - Autonome Aktivitäten/ Agenten

Fach-Sem.	KP	Kernbereich Informatik	101	13	Mathematik	32	5	F
1	30	CS1000 Einführung in die Programmierung	10	1	MA1000 Lin. Algebra u. Disk. Strukturen 1	8	1	
					MA2000 Analysis 1	8	1	
2	30	CS1001 Algorithmen und Datenstrukturen	8	1	MA1500 Lin. Algebra u. Disk. Strukturen 2	8	1	
		CS1201 Technische Grundlagen 1	6	1	MA2500 Analysis 2	4	1	
3	30	CS1202 Technische Grundlagen 2	6	1				CS:
		CS1002 Einführung in die Logik	4	1				
		CS2000 Theoretische Informatik	8	1				
		CS2300 SW-Engineering	6	1				
4	30	CS2301 SW-Engineering Praktikum	6	0	MA2510 Stochastik 1	4	1	
		CS2100 Rechnerarchitektur	4	1				
		CS2150 Betriebssysteme und Netze	8	1				
		CS2700 Datenbanken	4	1				
5	33	CS3000 Algorithmendesign	4	1				CS:
		CS3010 Mensch-Computer-Interaktion	4	1				CS:
6	27	Wahlpflicht Informatik	8	1-2				
		CS3990 Bachelorarbeit Informatik Kolloquium	12 3					

Web and Data Sc.	36	7
CS1800 Efg in Web und Data Science	4	1
CS3050 Codierung und Sicherheit	4	1
MA3110 Numerik I	4	1
CS3051 Parallel-verarbeitung	4	1

Bache

Fach-Sem.	KP	Kernbereich Informatik	101	13	Mathematik	32	5	Fachübergr. Bereich	11	0	Web and Data Sc.	36	7
1	30	CS1000 Einführung in die Programmierung	10	1	MA1000 Lin. Algebra u. Disk. Strukturen 1	8	1				CS1800 Efg in Web und Data Science	4	1
					MA2000 Analysis 1	8	1						
2	30	CS1001 Algorithmen und Datenstrukturen	8	1	MA1500 Lin. Algebra u. Disk. Strukturen 2	8	1				CS3050 Codierung und Sicherheit	4	1
		CS1201 Technische Grundlagen 1	6	1	MA2500 Analysis 2	4	1						
3	30	CS1202 Technische Grundlagen 2	6	1				CS2450 Werkzeuge für wiss. Arbeiten	2	0	MA3110 Numerik I	4	1
		CS1002 Einführung in die Logik	4	1									
		CS2000 Theoretische Informatik	8	1									
		CS2300 SW-Engineering	6	1									
4	30	CS2301 SW-Engineering Praktikum	6	0	MA2510 Stochastik 1	4	1						
		CS2100 Rechnerarchitektur	4	1									
		CS2150 Betriebssysteme und Netze	8	1									
		CS2700 Datenbanken	4	1									
5	33	CS3000 Algorithmendesign	4	1				CS					
		CS3010 Mensch-Computer-Interaktion	4	1				CS					
6	27	Wahlpflicht Informatik	8	1-2									
		CS3990 Bachelorarbeit Informatik Kolloquium	12 3										

CS3130 Algorithmische Datenanalyse 8	1
CS3100 Signal-verarbeitung 8	1
CS3204 Künst. Intel. 1 4 oder CSxxx Logik für Inform.	1

Bache

Fach-Sem.	Kernbereich Informatik	KP	Prüf
1	Basismodul Theoretische Informatik CS4000 Algorithmik (WS) oder CS4020 Spezifikation und Modellierung (SS)	6	1
	Basismodul Praktische Informatik CS4130 Webbasierte Informationssysteme (SS) oder CS4150 Verteilte Systeme (WS)	6	1
	Basismodul Technische Informatik CS4160 Echtzeitsysteme (WS) oder CS4170 Parallelrechnersysteme (SS)	6	1
2 + 3	3 Vertiefungsmodule a 12 KP aus folgender Liste	36	3
	CS4501 Algorithmik, Logik und Komplexität CS4502 Parallele und verteilte Systeme CS4503 Ambient Computing und Anwendungen CS4504 Cyber Physical Systems CS4505 Systemarchitektur CS4506 Datensicherheit CS4507 Softwareverifikation CS4508 Datenmanagement CS4509 Internet-Technologien CS4510 Signalanalyse CS4511 Lernende Systeme CS4512 Bildgeb. Systeme und inverse Probleme CS4520 Fallstudie zur prof. Produktentwicklung		

Fachübe

mindest

CS5840 er (W

EC4001 AI (W

EC4010 W (W

PS5810 W Lehrtätigk

PS5830 St Business

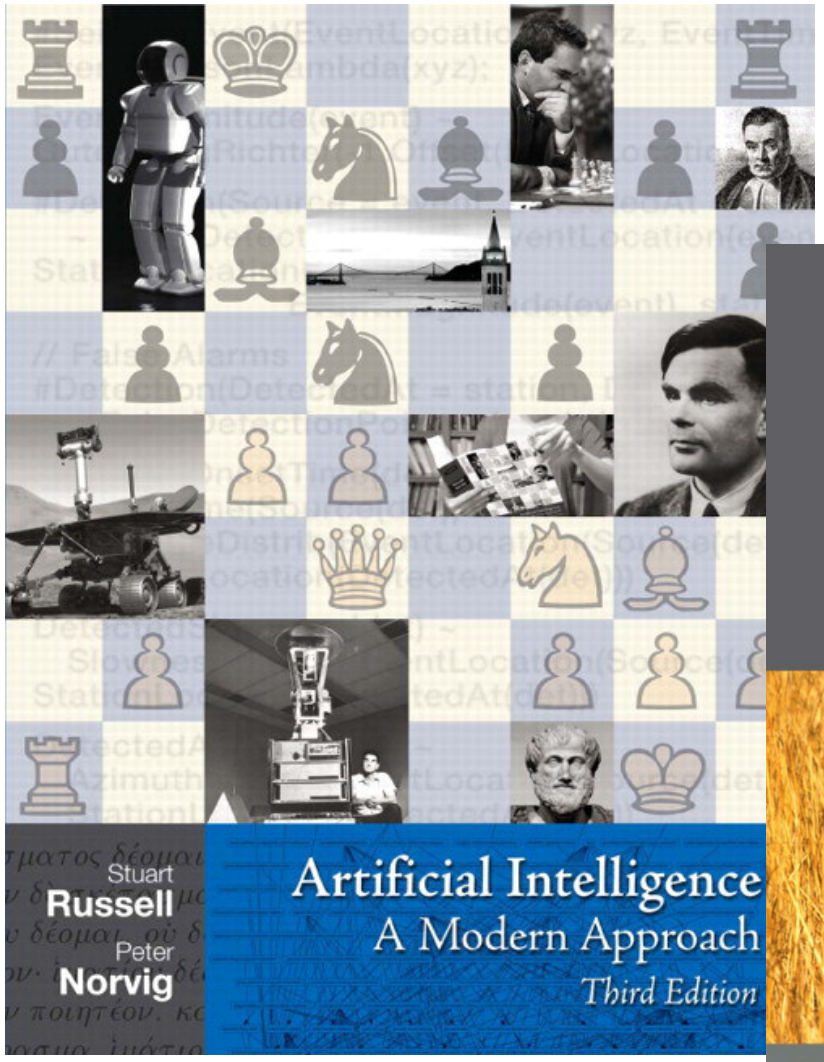
Web and Data Science	KP	Prüf
CS5131 Web and DB Mining Agents (WS)	8	1
MA4030		

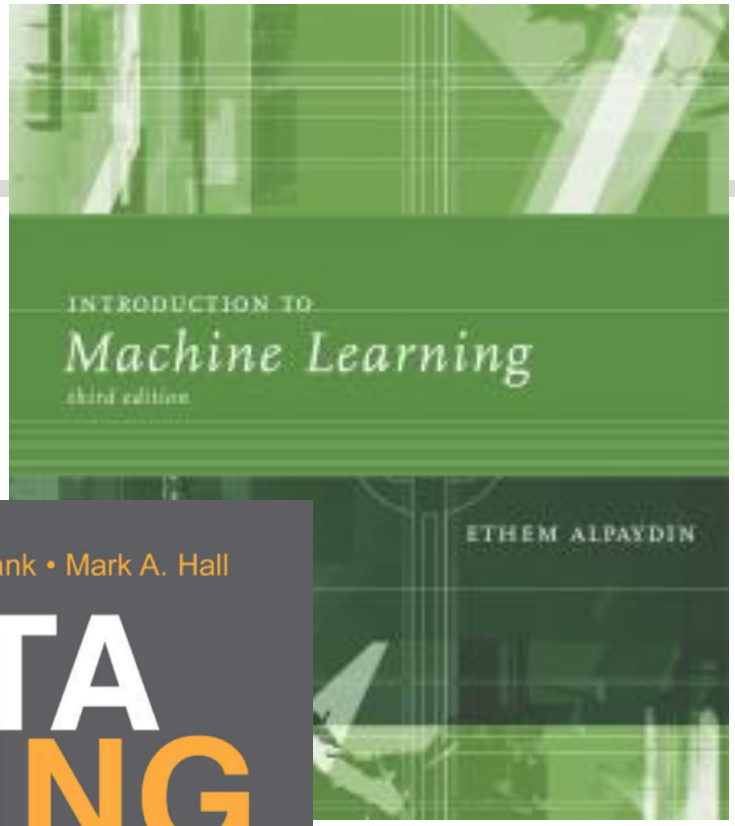
Vert. Th. Web and Data Science:
CS4250 Computer Vision
MA4610 Stochastische Prozesse
CS5275 Ausgewählte Methoden der Signalanalyse und Verbesserung
MA4940 Test- und Schätztheorie
MA4341 Zeitreihenanalyse
MA4040 Numerik 2

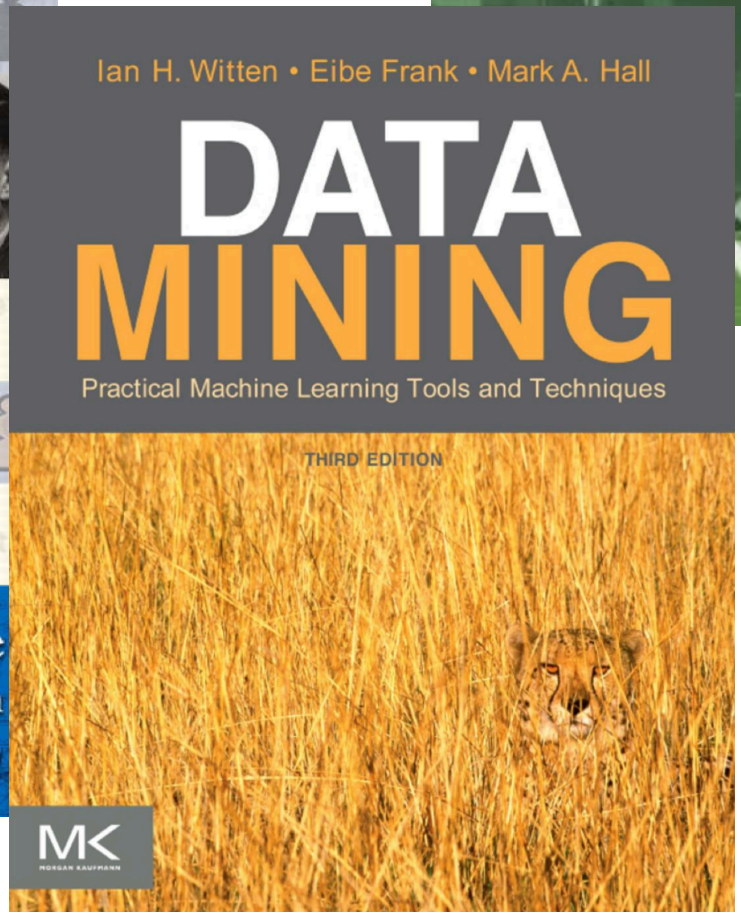
CS4290 Vertiefende Themen Web and Data Science	4	1
CSxxx Projektprakt. Web and Data Science	4	0

Master

Literatur

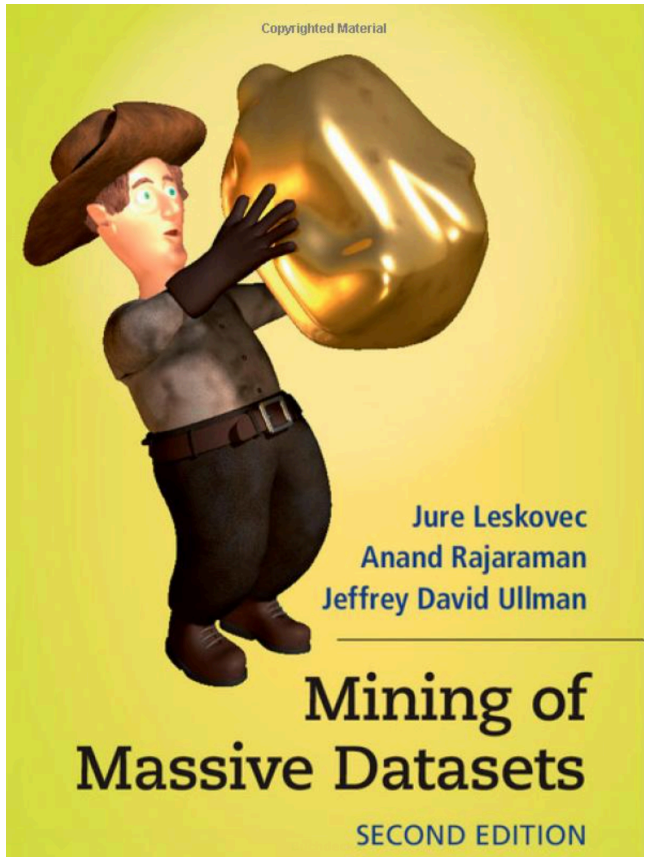

Stuart
Russell
Peter
Norvig
Artificial Intelligence
A Modern Approach
Third Edition


INTRODUCTION TO
Machine Learning
third edition
ETHEM ALPAYDIN


Ian H. Witten • Eibe Frank • Mark A. Hall
DATA
MINING
Practical Machine Learning Tools and Techniques
THIRD EDITION
MK

Literatur

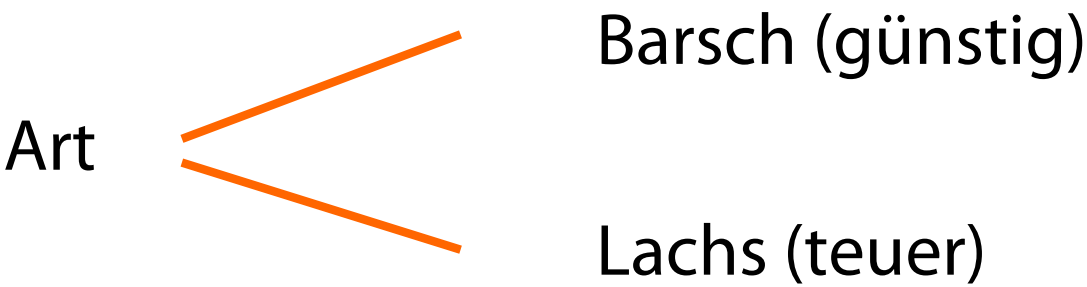
- Stuart Russell, Peter Norvig, **Artificial Intelligence – A Modern Approach**, Pearson, 2009 (oder 2003er Ed.)
- Ian H. Witten, Eibe Frank, Mark A. Hall, Data Mining: **Practical Machine Learning Tools and Techniques**, Morgan Kaufmann, 2011
- Ethem Alpaydin, **Introduction to Machine Learning**, 3rd Ed., MIT Press, 2014
- Jure Leskovec, Anand Rajaraman, Jeffrey D. Ullman, **Mining of Massive Datasets**, 2nd Ed., Cambridge University Press, 2014



- Viele zusätzliche Bücher, Präsentationen, und Videos im Web

Ein erweitertes Beispiel

"Sortierung von Fischen auf einem Förderband nach Arten durch Bildverarbeitung"



Problemanalyse

Verwende Kamera und nehme Bilder auf,
um Merkmale zu bestimmen:

- Länge
- Helligkeit
- Breite
- Anzahl und Form der Flossen
- Position des Mundes usw.

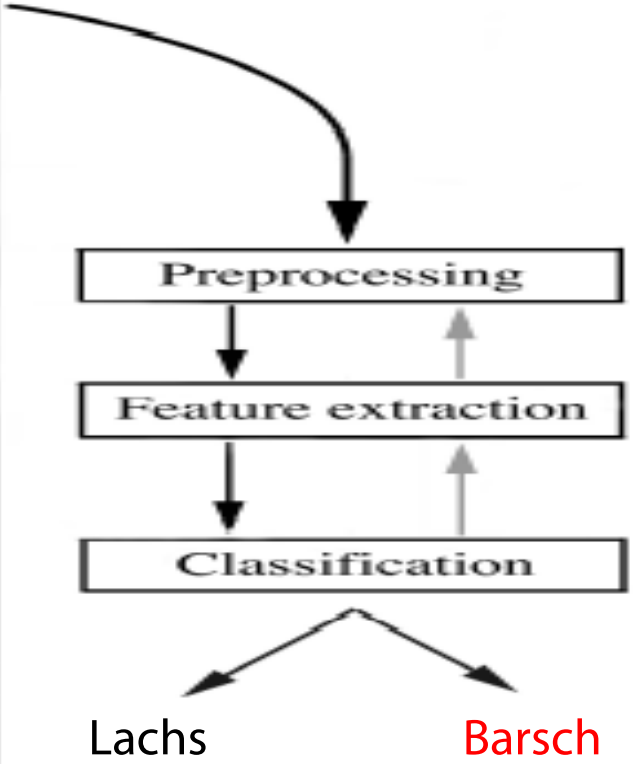
Menge aller möglichen Merkmale

Ziel: Wähle die relevanten aus

Vorverarbeitung

- Verwende Segmentierungsoperator, um Fisch und Hintergrund zu unterscheiden
- Fischregion wird verwendet, um Merkmalswerte zu extrahieren (geeignete Merkmale vorher festgelegt)
- Merkmalswerte weiterverarbeitet durch Klassifikator (Barsch oder Lachs)

Klassifikation



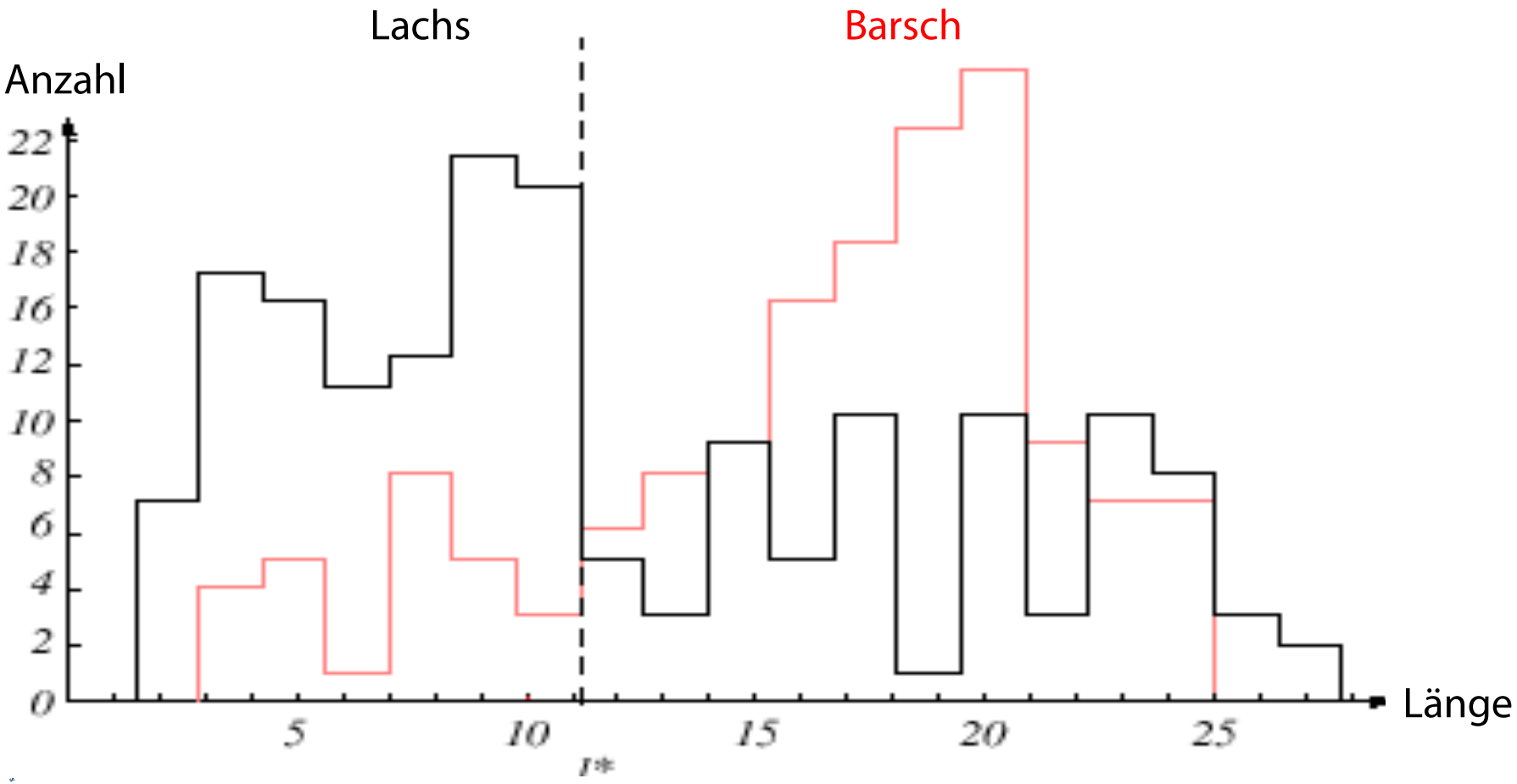
Bestimmung geeigneter Merkmale

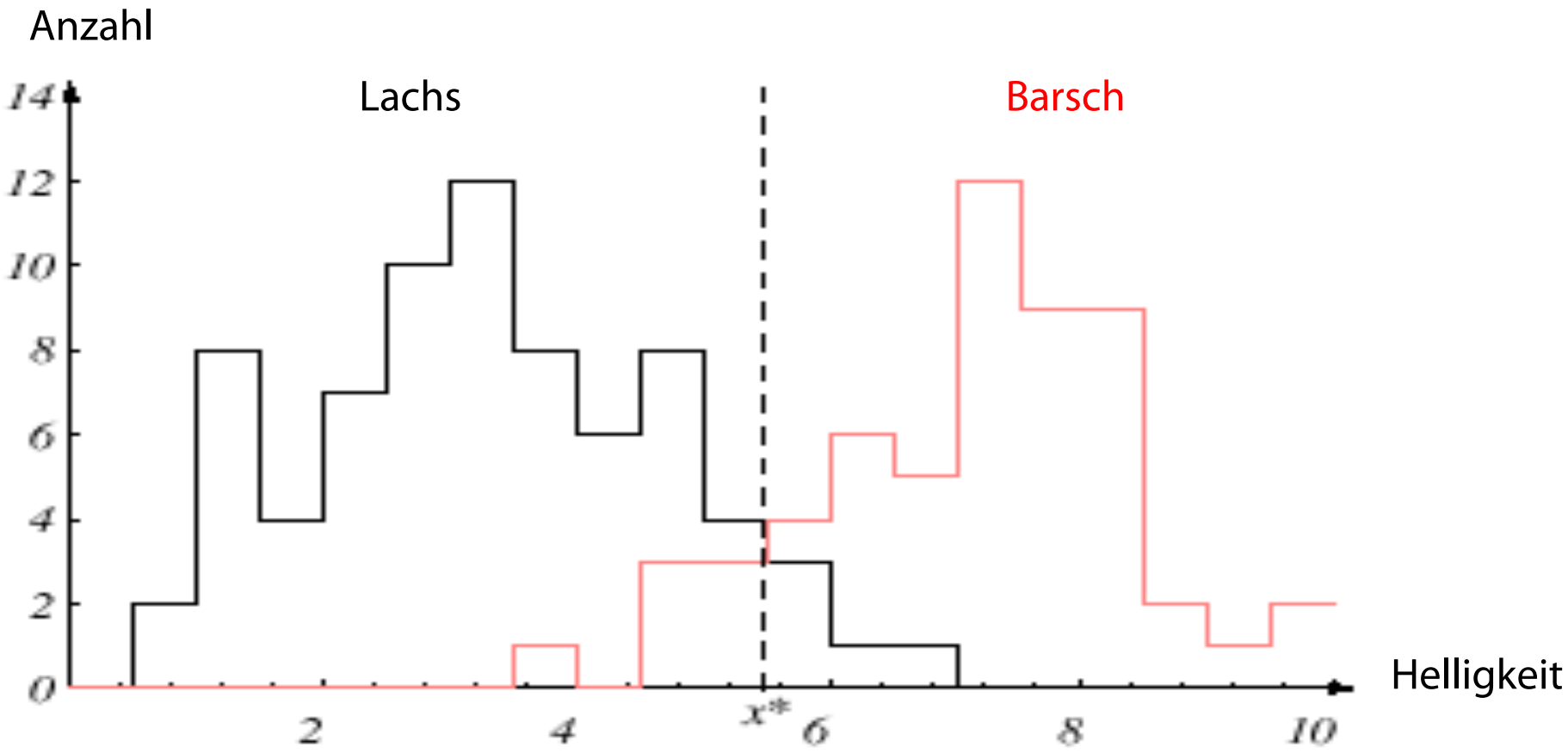
- Wir benötigen einen Experten, um die Merkmale festzulegen, mit denen man Barsche und Lachse richtig klassifizieren kann
- Wie wäre es mit Länge als Merkmal zur Unterscheidung?

Länge allein ist kein gutes Merkmal!

→ Hohe Kosten bei Fehlentscheidung

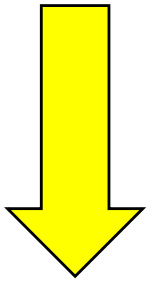
Wie wäre es mit Helligkeit?




Anzahl
Lachs
Barsch
14
12
10
8
6
4
2
0
2
4
x*
6
8
10
Helligkeit

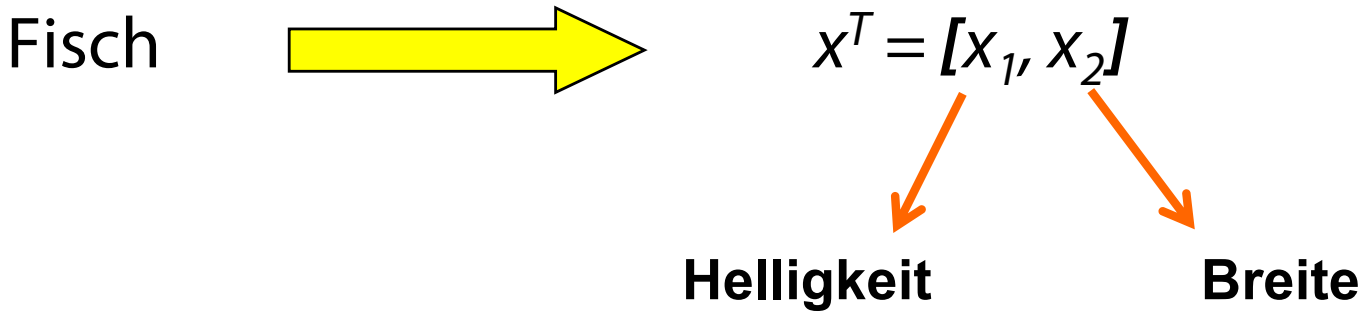
Schwellwert-Entscheidungsgrenze und induzierte Kosten

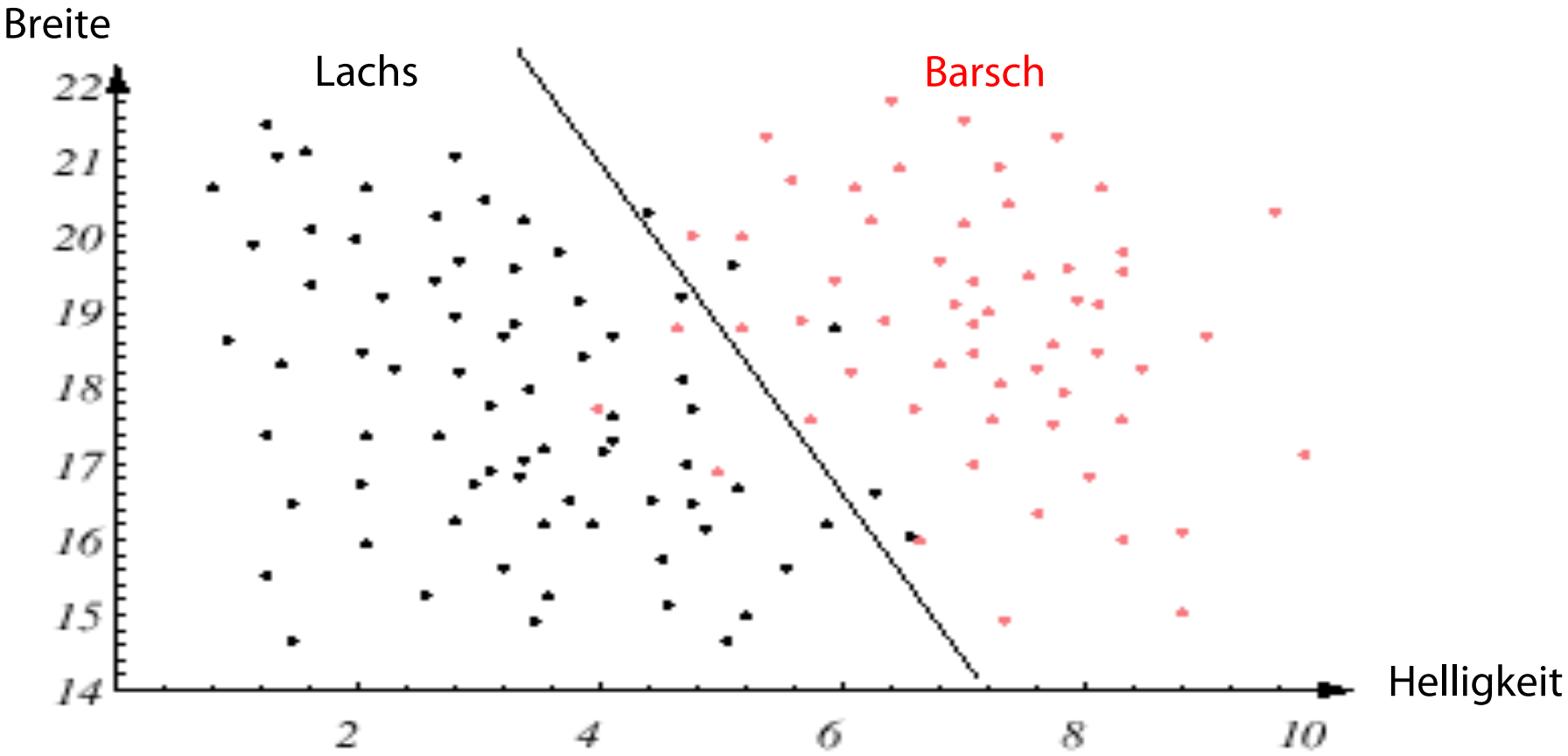
– Schwellwert-Entscheidungsgrenze in Richtung mittlerer Helligkeitswerte minimiert die Kosten (der Fehlklassifikation)



Untersuchung in der sog. Entscheidungstheorie

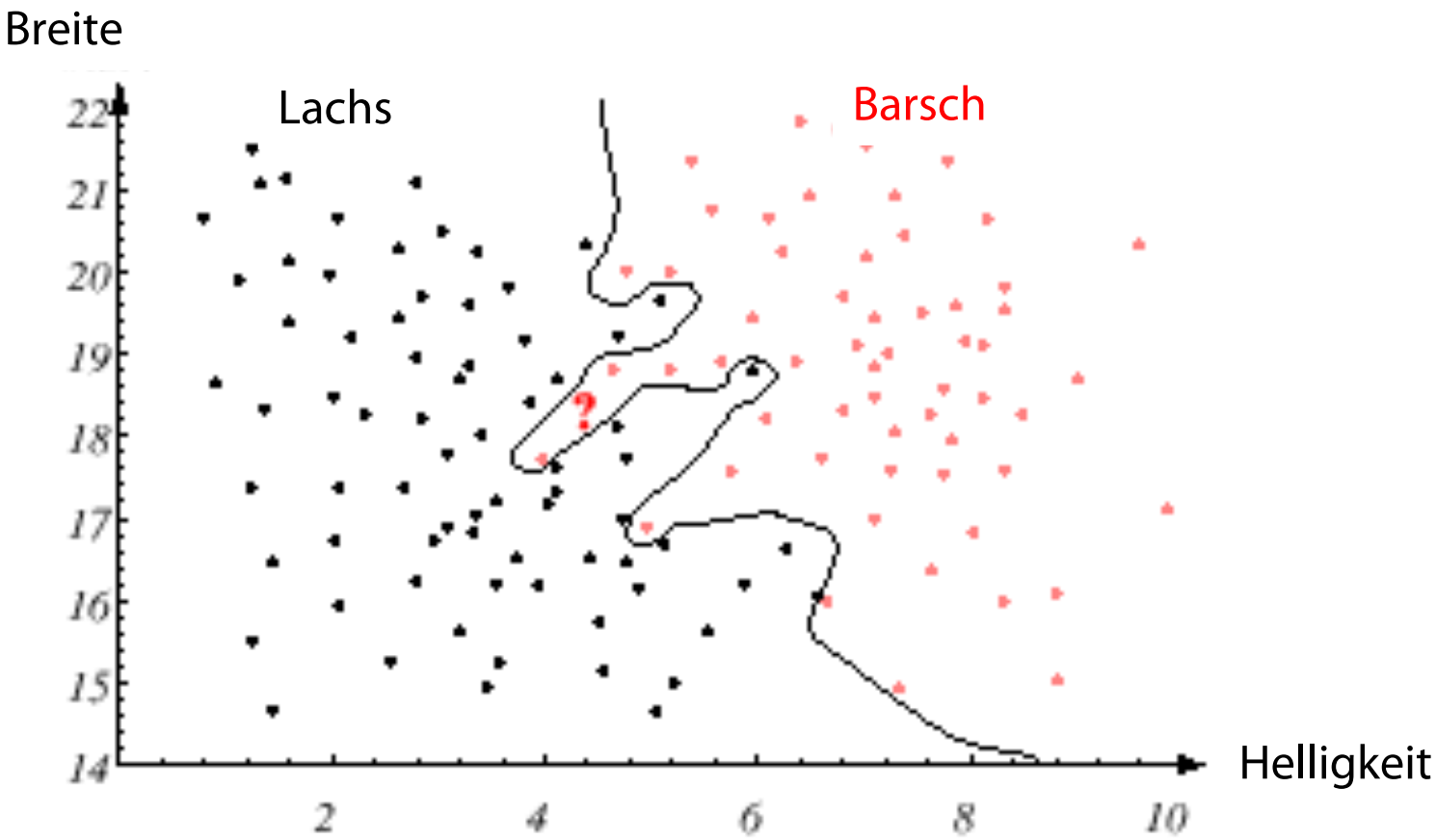
Passe Helligkeit an
und verwende zusätzlich die Breite des Fisches



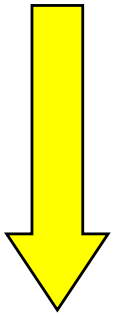

Breite
Lachs
Barsch
22
21
20
19
18
17
16
15
14
2
4
6
8
10
Helligkeit

- Weitere Merkmale, die nicht direkt zu Helligkeit und Breite in Beziehung stehen, könnten hinzukommen
  - Vorsicht aber vor Reduktion durch "verrauschte Merkmale"

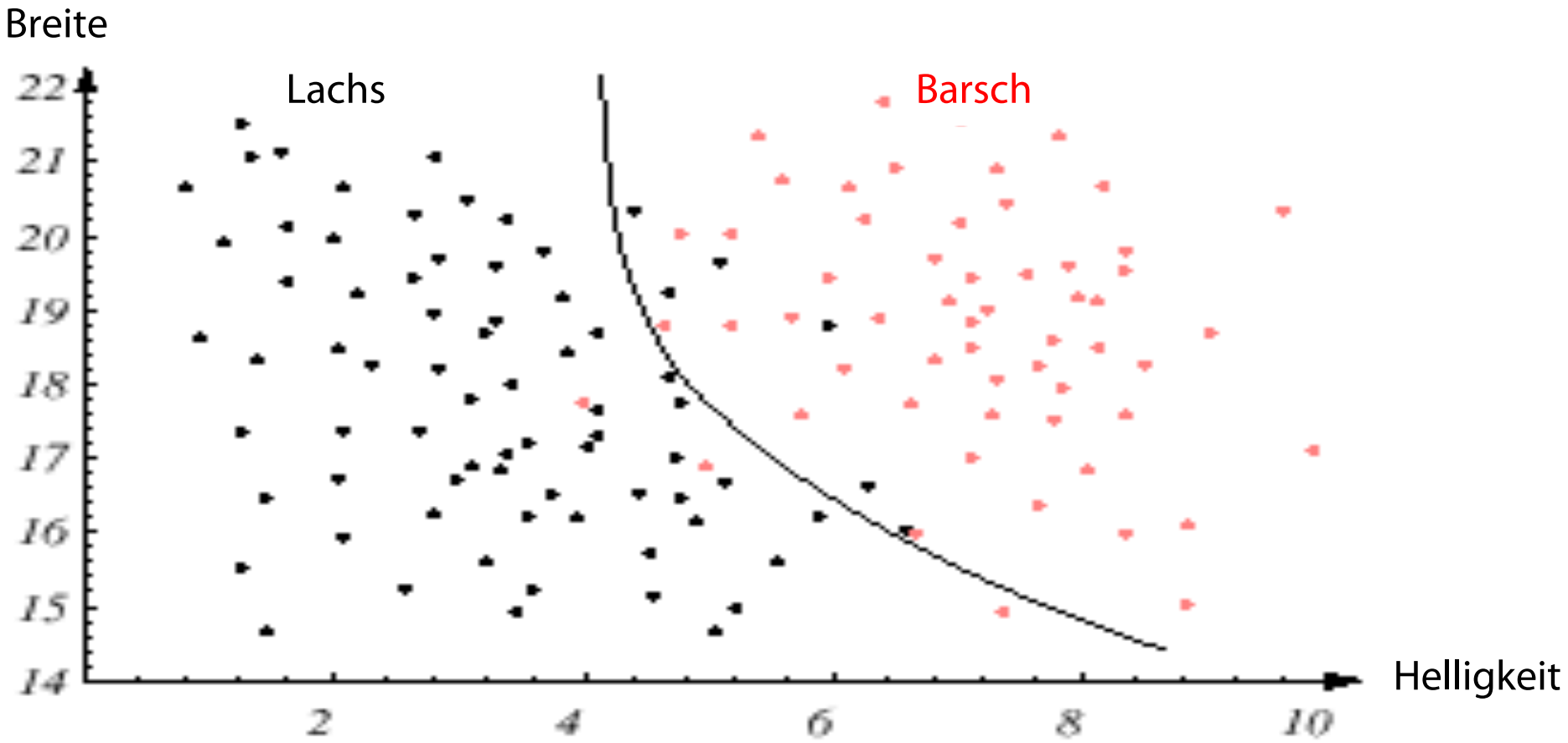
- Wünschenswerterweise ergibt die beste Entscheidungsgrenze eine optimale Performanz (im Sinne einer Verlustminimierung durch Falschklassifikation)


Breite
Lachs
Barsch
?
22
21
20
19
18
17
16
15
14
2
4
6
8
10
Helligkeit

Vorfreunde über Klassifikationsleistung auf Testdaten kann verfrüht sein. Wichtig ist die Leistung auf neuen Daten!



Generalisierungsfähigkeit zählt!


Breite
Lachs
Barsch
22
21
20
19
18
17
16
15
14
2
4
6
8
10
Helligkeit

Klassifikatorentwicklung

- Relevante Merkmale automatisch bestimmbar?
  - Deep Learning

- Dynamisch Angepassung des Klassifikators möglich?
  - Ohne spezielle Groundtruth Daten: Transduktives Lernen
  - Mit Belohnungssystem: Reinforcement-Lernen

- Übertragung eines Klassifikators auf neue Anwendung?
  - Transfer-Lernen

Bewertung eines Klassifikators

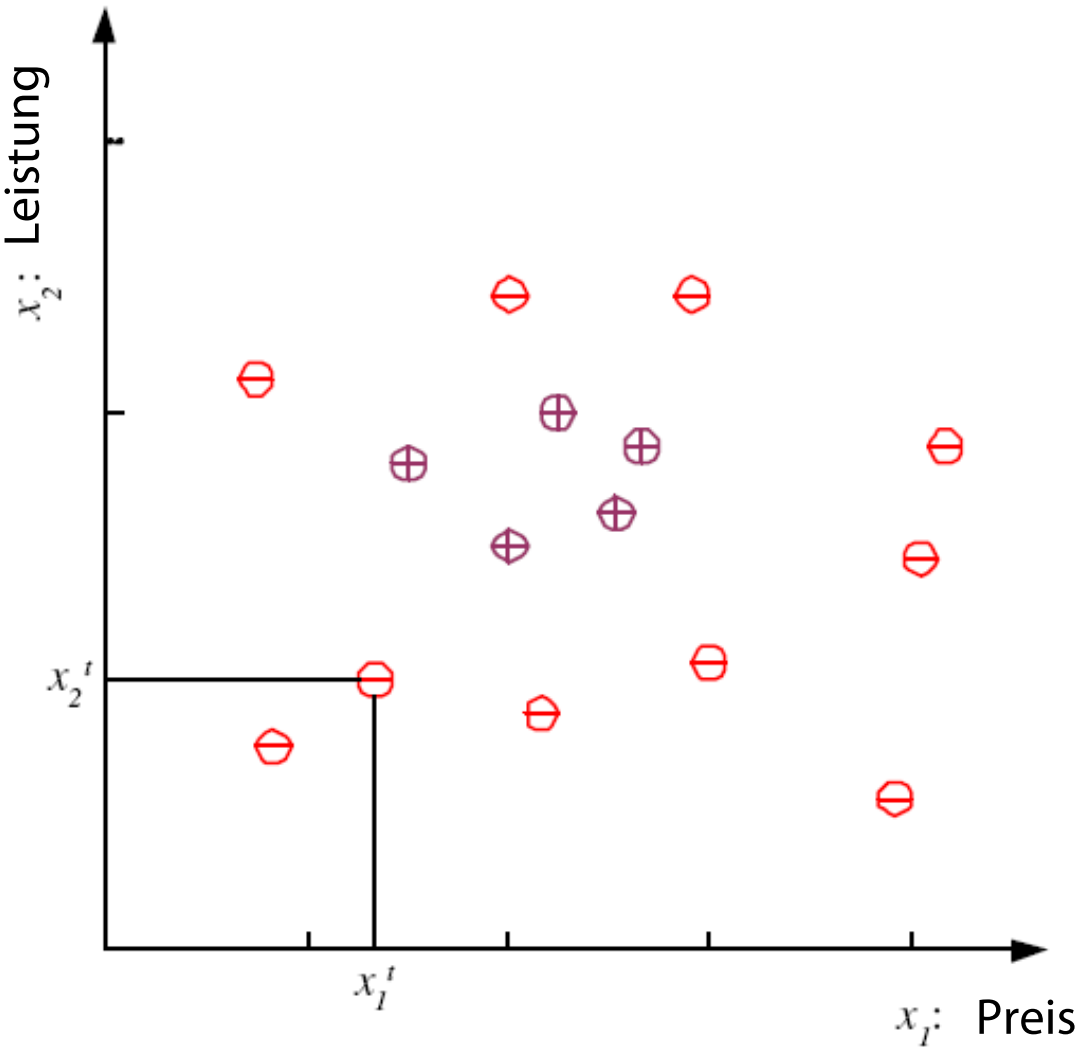
- Klasse C eines “Familienautos”
  - Vorhersage: Ist Auto x ein Familienauto?
  - Wissensextraction:
    Was erwarten Menschen von einem Familienauto?
- Ausgabe:

  Positive (+) und negative (–) Beispiele
- Repräsentation der Eingabe:

  x_1: Preis, x_2 : Leistung

Frage: Wie gut funktioniert ein bestimmter Klassifikator?

Trainingsmenge $\mathcal{X}$



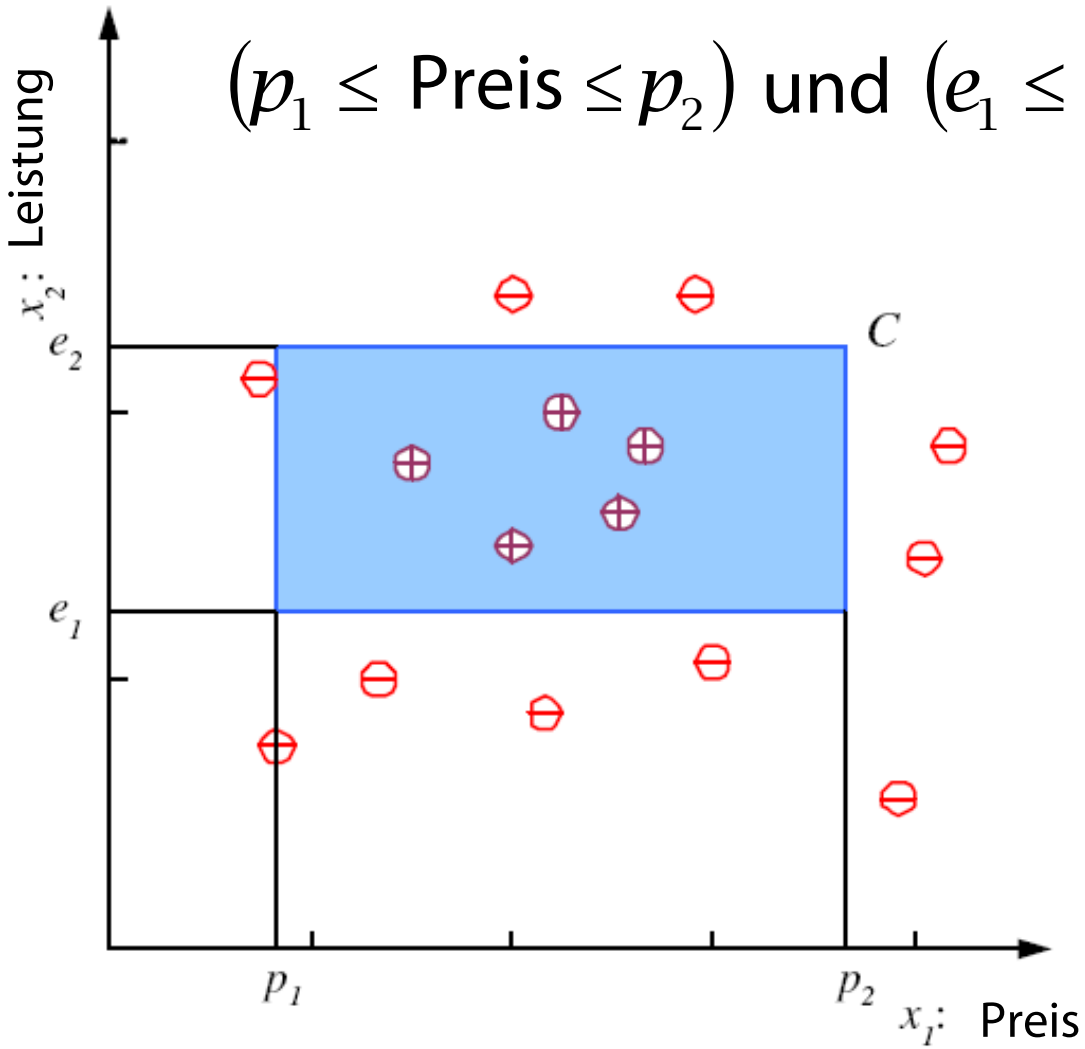
$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^N$$

$$r = \begin{cases} 1 \text{ if } \boldsymbol{x} & \text{ist positiv} \\ 0 \text{ if } \boldsymbol{x} & \text{ist negativ} \end{cases}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

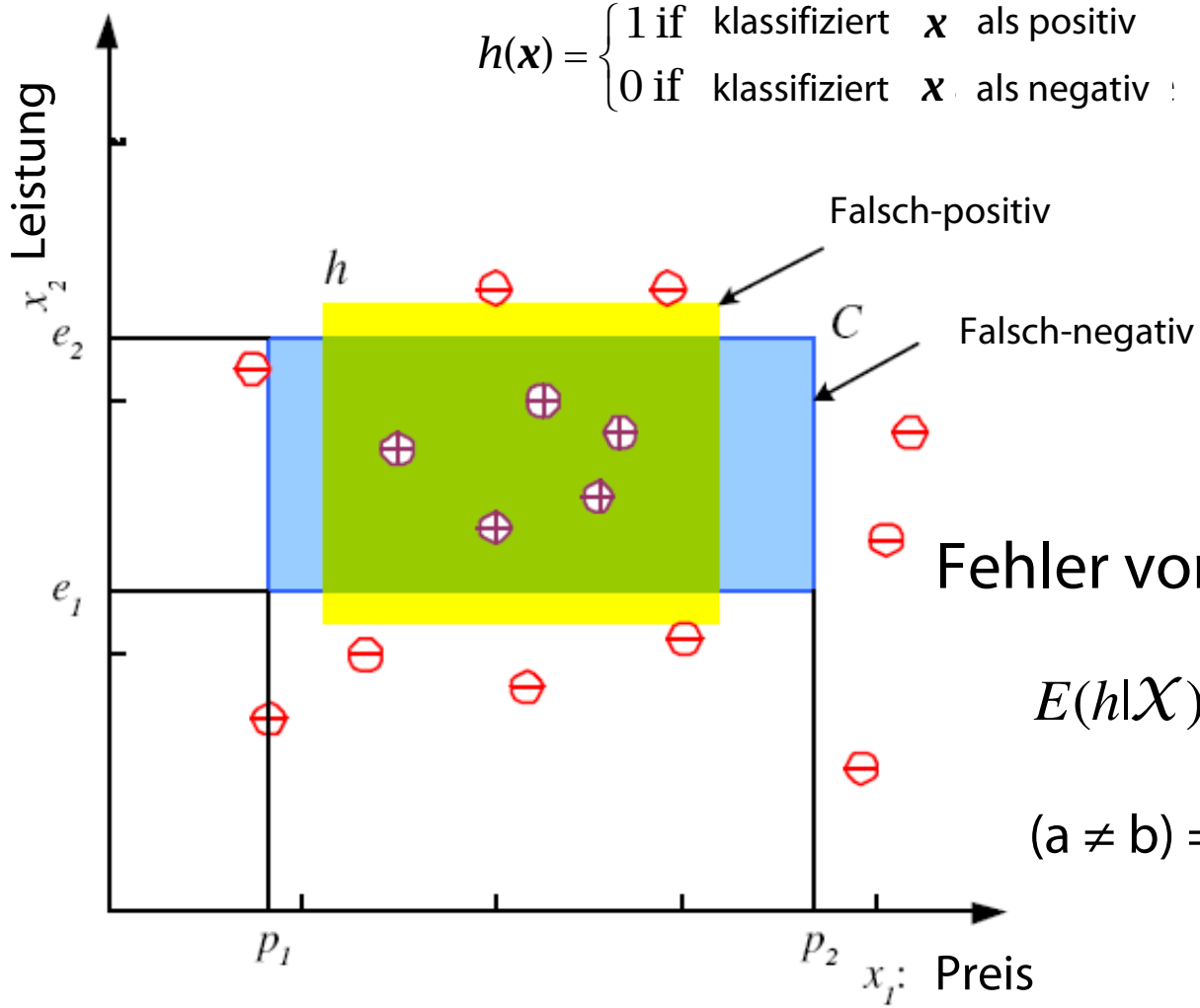
Richtige Klasse C

$(p_1 \leq \text{Preis} \leq p_2)$ und $(e_1 \leq \quad \text{Leistung} \quad \leq e_2)$



Hypothesenklasse $\mathcal{H}$ (gelb)

$$h(\boldsymbol{x}) = \begin{cases} 1 \text{ if } & \text{klassifiziert } \boldsymbol{x} \text{ als positiv} \\ 0 \text{ if } & \text{klassifiziert } \boldsymbol{x} \text{ als negativ} \end{cases}$$

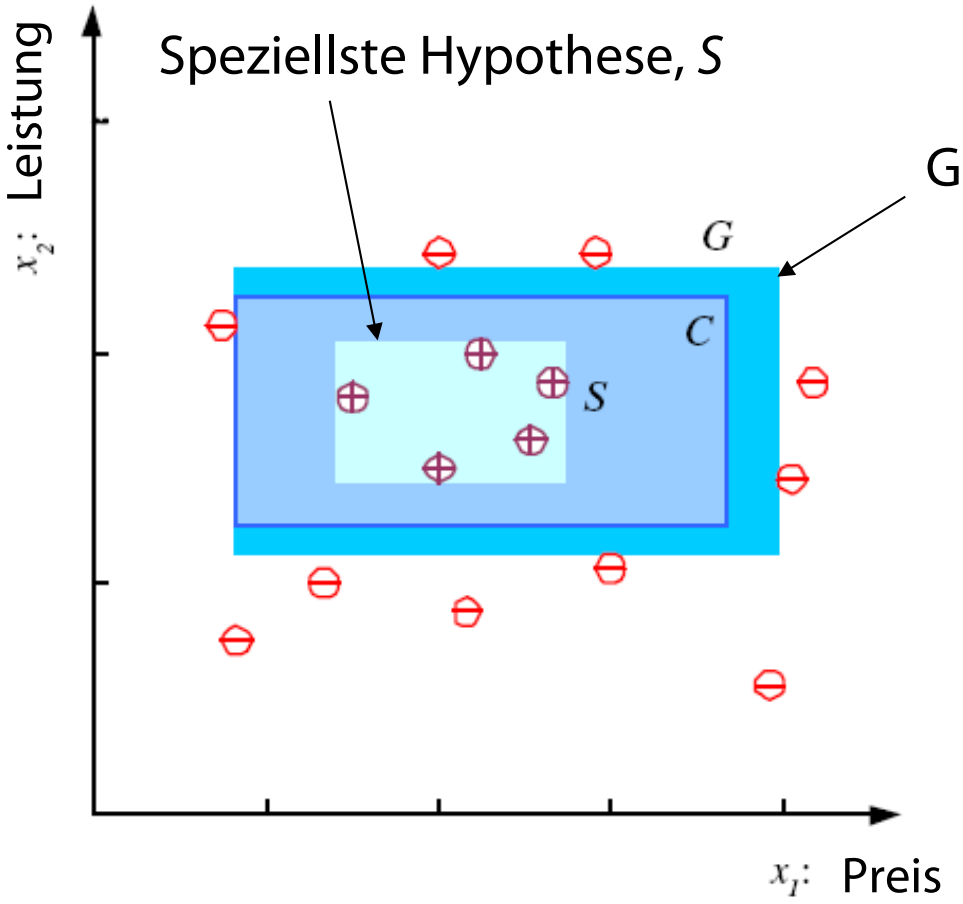


Fehler von h bzgl. $\mathcal{H}$

$$E(h|\mathcal{X}) = (1/N)\sum_{t=1}^{N}\left(h\left(\mathbf{x}^t\right) \neq r^t\right)$$

(a ≠ b) = 1 if ≠, 0 sonst

S, G, and der Versionsraum (Version Space)



$h \in$ H, zwischen S and G ist konsistent und definiert den Versionsraum

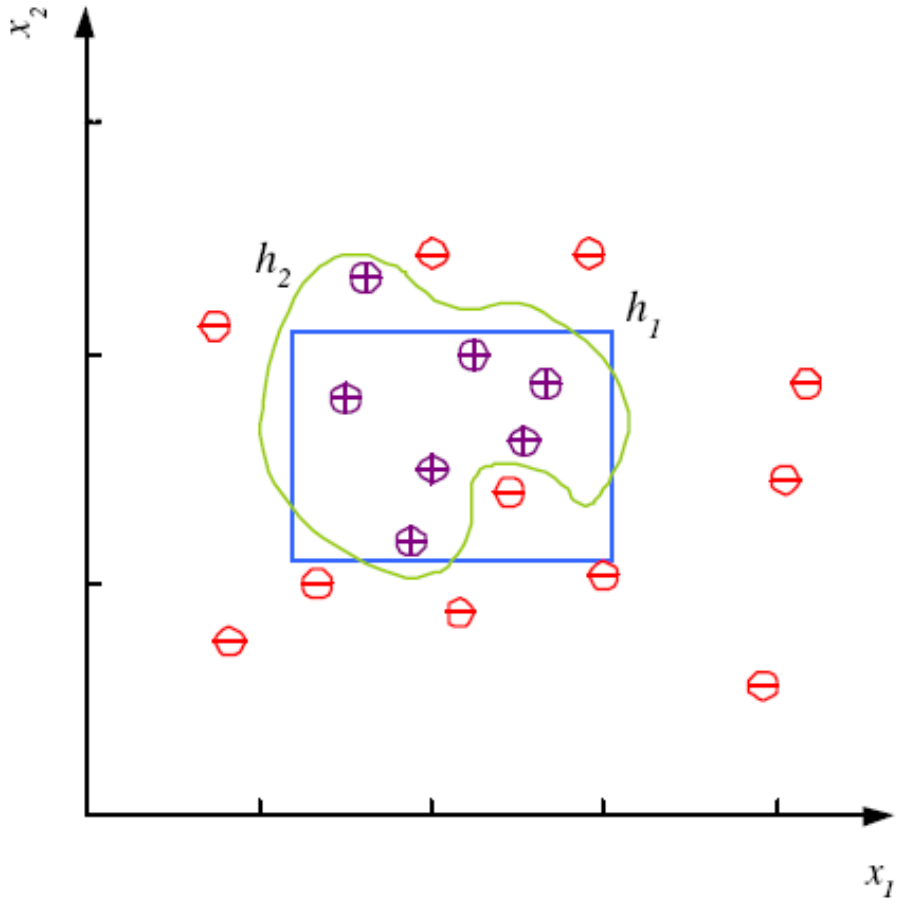
Mitchell, Tom M., Generalization as search. *Artificial Intelligence*. **18** (2): 203–226, **1982**

Rauschen und Modellkomplexität

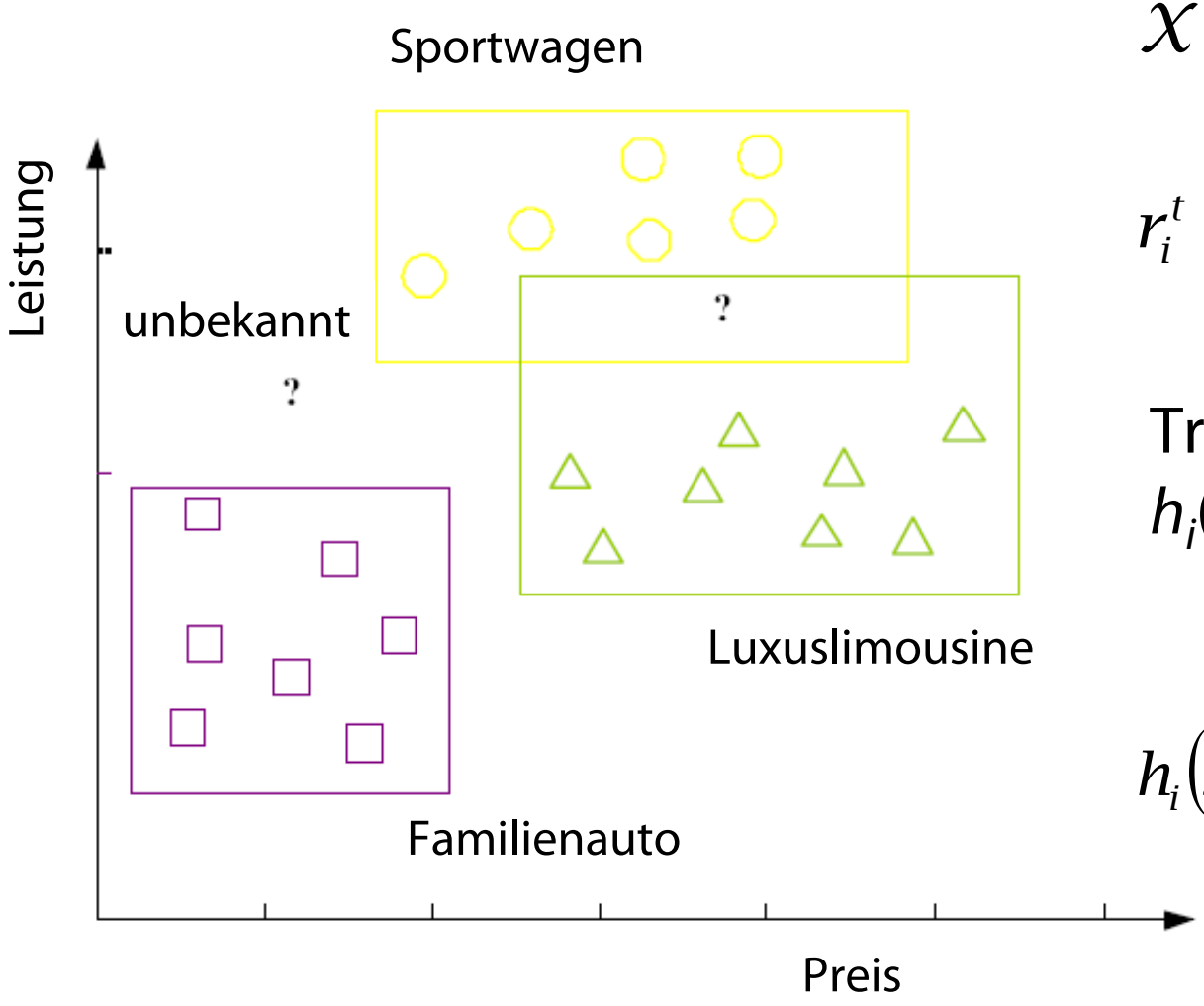
Verwende einfaches Modell:

- Einfacher zu verwenden
  (weniger Berechnungsschnitte)
- Leichter zu trainieren
  (weniger Daten zu speichern)
- Leichter zu erklären
  (besser interpretierbar)
- Bessere Generalisierung
  (Occam's Razor)

Modellkomplexität:
"Größe" der Beschreibung



Clustering: Verschiedene Klassen, C_i i=1,...,K



$$\mathcal{X} = \{\boldsymbol{x}^t, r^t\}_{t=1}^N$$

$$r_i^t = \begin{cases} 1 \text{ if } \boldsymbol{x}^t \in C_i \\ 0 \text{ if } \boldsymbol{x}^t \in C_j, j \neq i \end{cases}$$

Trainiere Hypothesen $h_i(\boldsymbol{x})$, $i = 1,...,K$:

$$h_i\left(\boldsymbol{x}^t\right) = \begin{cases} 1 \text{ if } \boldsymbol{x}^t \in C_i \\ 0 \text{ if } \boldsymbol{x}^t \in C_j, j \neq i \end{cases}$$

Regression: Verschiedene Modellklassen

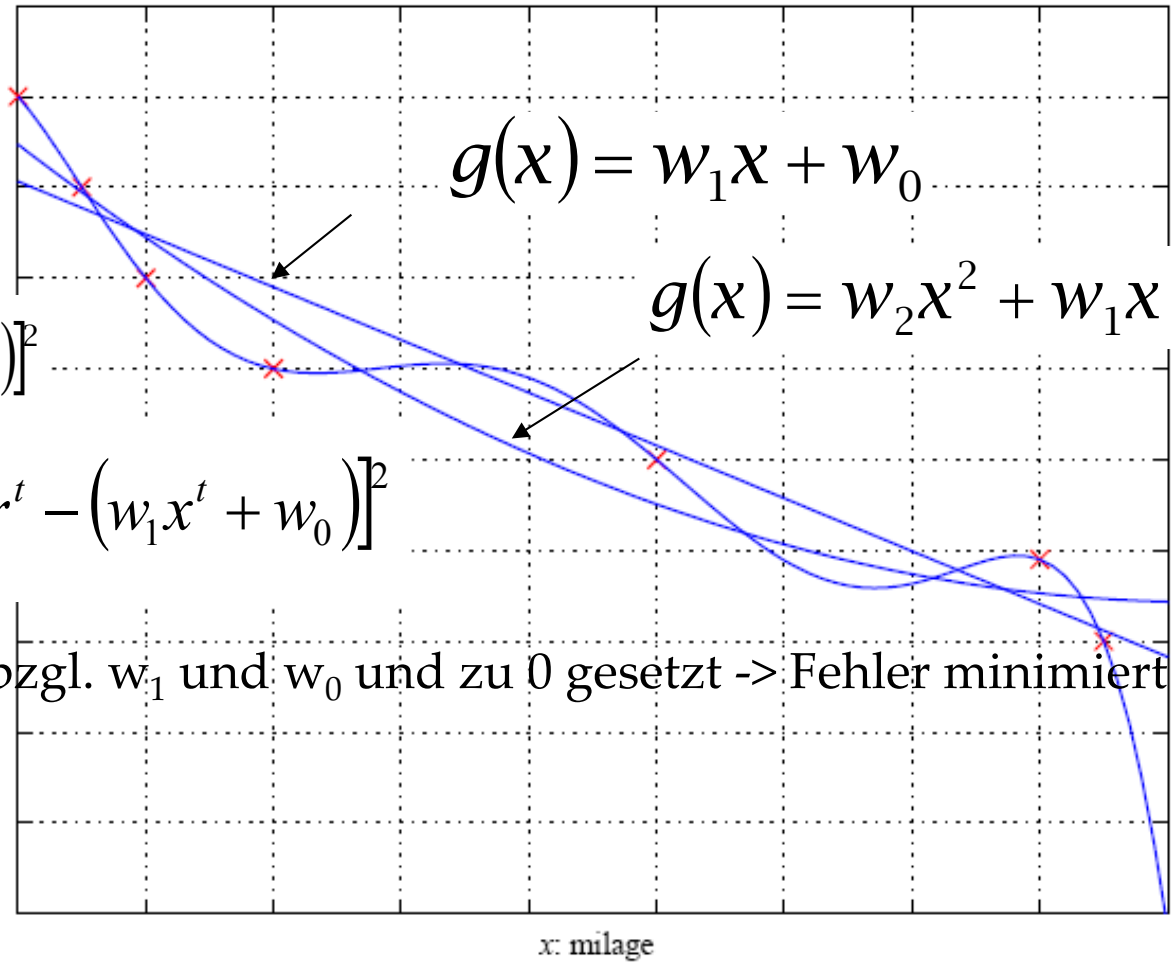
$$\mathcal{X} = \left\{x^t, r^t\right\}_{t=1}^{N}$$

$$r^t \in \Re$$

$$r^t = f\left(x^t\right)$$

$$E(g \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - g\left(x^t\right)\right]^2$$

$$E\left(w_1, w_0 \mid \mathcal{X}\right) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - \left(w_1 x^t + w_0\right)\right]^2$$



Partielle Ableitungen von E bzgl. w_1 und w_0 und zu 0 gesetzt -> Fehler minimiert

$$w_1 = \frac{\sum_t x^t r^t - \bar{x}\bar{r}N}{\sum_t (x^t)^2 - N\bar{x}^2}$$

$$w_0 = \bar{r} - w_1\bar{x}$$

Überwachtes Lernen

- Beispiel: Regression
  - Gegeben Datenpunkte, bestimme Parameter, so dass für gegebene x-Werte, die y-Werte geschätzt werden können

- Beispiel: Klassifikation
  - Gegebene Datenpunkte jeweils mit Klassifikationswert, bestimme Klassifikationswert für Datenpunkte ohne diesen (binärer oder mehrwertiger Klassifikator)

Modellauswahl & Generalization

- Lernen kann als Optimierungsproblem angesehen werden:
  - Berechne Modell, so dass Fehlerfunktion minimiert
- Lernen ist ein schlecht gestelltes Problem;
  In der Regel reichen die Daten nicht, eine eindeutige Lösung des Optimierungsproblems zu finden
- Es werden Vorannahmen benötigt (inductive bias), Annahmen bzgl. H
- Generalisierung: Wie gut arbeitet das Modell auf neuen Daten?
- Überanpassung (Overfitting): H komplexer als C oder f
- Unteranpassung (Underfitting): H weniger komplex als C oder f

Drei-Wege-Austauschbeziehung

(Dietterich, 2003):

1. Komplexität von $\mathcal{H}$, $c(\mathcal{H})$,
2. Trainingsmengengröße N,
3. Generalisierungsfehler, E, auf neuen Daten

- Wenn $N\uparrow$, $E\downarrow$
- Wenn $c(\mathcal{H})\uparrow$, gilt zuerst $E\downarrow$ und dann $E\uparrow$

Kreuzvalidierung

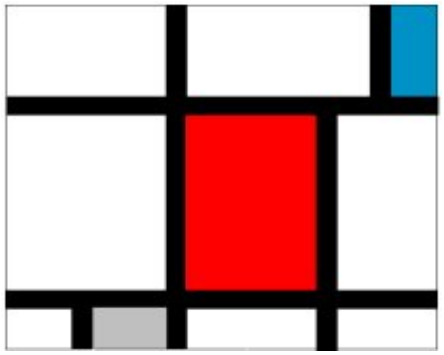
- Um den Generalisierungsfehler abzuschätzen, brauchen wir Daten, mit denen nicht trainiert wurde
- Aufteilung der Daten:
  - Trainingsmenge (50%)
  - Validierungsmenge (25%)
  - Testmenge (z.B. für Publikation) (25%)
- Neuabtastung, wenn wenige Daten vorhanden

Betrachtungsebenen überwachtes Lernen

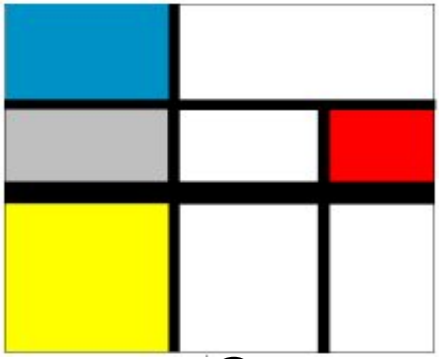
1. Modell: $g(\boldsymbol{x} \mid \theta)$

2. Fehlerfunktion (Verlustfunktion): $E(\theta \mid \mathcal{X}) = \sum_t L(r^t, g(\boldsymbol{x}^t \mid \theta))$

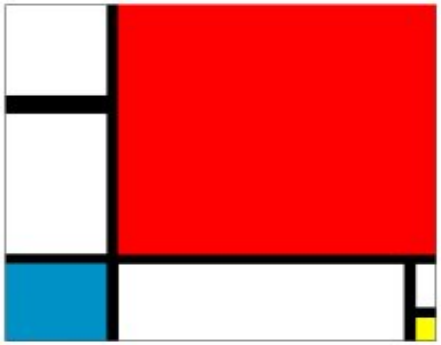
3. Optimierungs verfahren: $\theta^* = \arg\min_\theta E(\theta \mid \mathcal{X})$



1

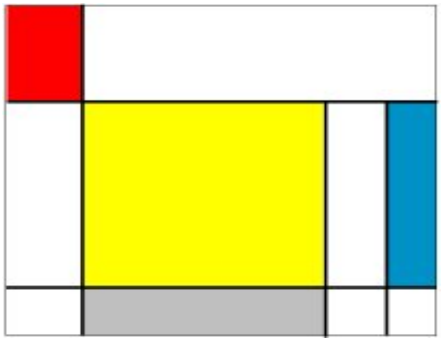


2

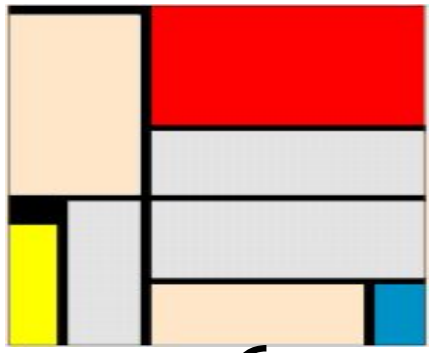


3

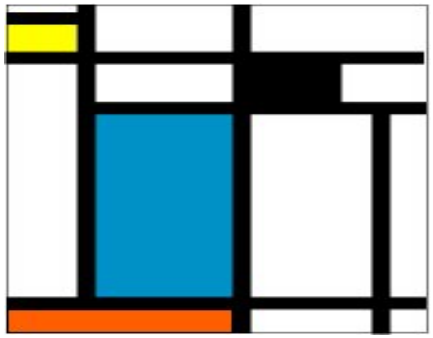
4



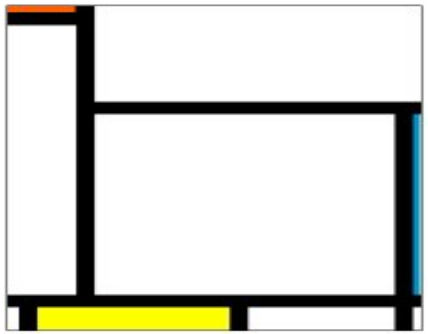
5



6



7



8 ?

Daten in Tabellarischer Form

Number	Lines	Line types	Rectangles	Colours	Mondrian?
1	6	1	10	4	*No*
2	4	2	8	5	*No*
3	5	2	7	4	Yes
4	5	1	8	4	Yes
5	5	1	10	5	*No*
6	6	1	8	6	Yes
7	7	1	14	5	*No*

Anfrage

Number	Lines	Line types	Rectangles	Colours	Mondrian?
8	7	2	9	4	

Analyse von Daten

- Betrachtung einer Spalte x mit n Werten
- Bestimmung des Mittelwerts: $\overline{x} = 1/n \cdot \Sigma_{i=1}^{n} x_i$
- Große und kleine Werte können sich aufheben
- Mittlere Abweichung vom Mittelwert betrachten (Varianz)
- Bestimmung der Varianz: $var = 1/n \cdot \Sigma_{i=1}^{n} (\overline{x}-x_i)^2$
- Meist betrachtet wird die sog. Standardabweichung: $\sigma = \sqrt{var}$

Halte Daten in normalisierter Form vor

Eine Möglichkeit zur Normalisierung:

$$x_t' \equiv \frac{x_t - \bar{x}_t}{\sigma_t}$$

Gemittelte Abweichung vom Mittel

Normalisierte Trainingsdaten

Number	Lines	Line types	Rectangles	Colours	Mondrian?
1	0.632	-0.632	0.327	-1.021	*No*
2	-1.581	1.581	-0.588	0.408	*No*
3	-0.474	1.581	-1.046	-1.021	Yes
4	-0.474	-0.632	-0.588	-1.021	Yes
5	-0.474	-0.632	0.327	0.408	*No*
6	0.632	-0.632	-0.588	1.837	Yes
7	1.739	-0.632	2.157	0.408	*No*

$$d(\mathbf{x}_i,\mathbf{x}_j) = \sqrt{\sum_{t=1}^{T}[x_{it} - x_{jt}]^2}$$

Number	Lines	Line types	Rectangles	Colours	Mondrian?
8	1.739	1.581	-0.131	-1.021	

Normalisierte Trainingsdaten

Number	Lines	Line types	Rectangles	Colours	Mondrian?
1	0.632	-0.632	0.327	-1.021	*No*
2	-1.581	1.581	-0.588	0.408	*No*
3	-0.474	1.581	-1.046	-1.021	Yes
4	-0.474	-0.632	-0.588	-1.021	Yes
5	-0.474	-0.632	0.327	0.408	*No*
6	0.632	-0.632	-0.588	1.837	Yes
7	1.739	-0.632	2.157	0.408	*No*

$$\sqrt{(0 + 4{,}89 + 5{,}23 + 2{,}04)} = 3{,}489$$

Number	Lines	Line types	Rectangles	Colours	Mondrian?
8	1.739	1.581	-0.131	-1.021	

Distanz der Testinstanz von den Trainingsdaten

Beispiel	Distanz zum Test	Mondrian?
1	2.517	*No*
2	3.644	*No*
3	2.395	Yes
4	3.164	Yes
5	3.472	*No*
6	3.808	Yes
7	3.490	*No*

Klassifikation

1-NN	Yes
3-NN	Yes
5-NN	*No*
7-NN	*No*

Was verwenden wir bei reellwertiger Zielfuntion als Ausgabe?

- Mittel der k-nächsten Nachbarn

Variante von kNN: Distanzgewichtetes kNN

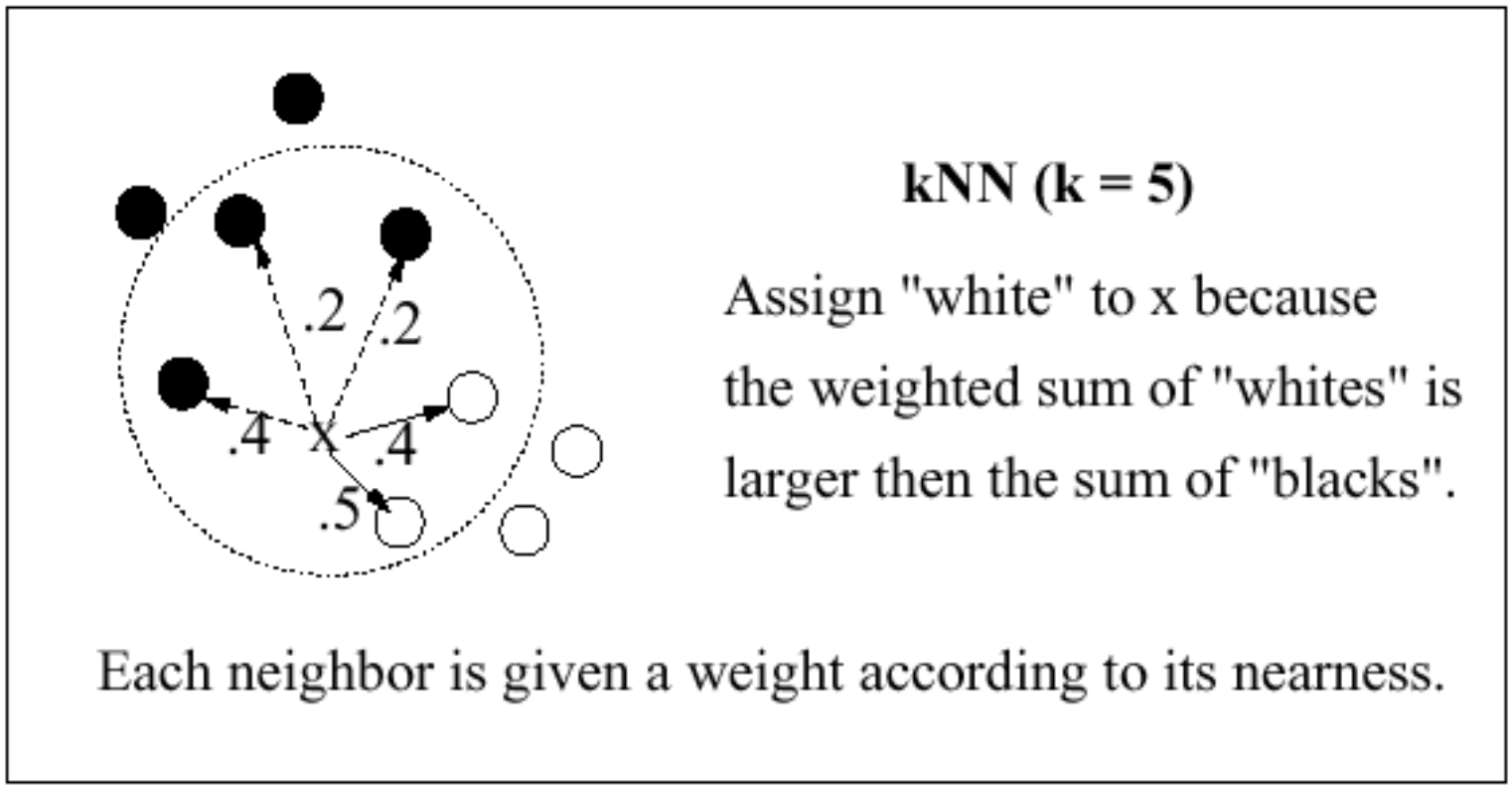
- Nähere Nachbarn haben mehr Einfluss

$$f(\mathbf{x}_q) := \frac{\sum_{i=1}^{k} w_i f(\mathbf{x}_i)}{\sum_{i=1}^{k} w_i} \quad \text{where } w_i = \frac{1}{d(\mathbf{x}_q, \mathbf{x}_i)^2}$$

Dudani, S.A. The distance-weighted k-nearest-neighbor rule. IEEE Trans. Syst. Man Cybern., SMC-6:325–327, **1976**

Variante von kNN: Distanzgewichtetes kNN

k-NN using a weighted-sum voting scheme



Dann könnten wir statt nur k gleich alle Trainingsinstanzen (= Beispiele) nehmen

kNN: Zusammenfassung

- Sehr einfacher Ansatz
- Verhält sich auch noch gutartig, wenn Daten nicht einfach separiert werden können
- Rang 7 der 10 wichtigsten Data-Mining-Verfahren