



BEUTH HOCHSCHULE FÜR TECHNIK BERLIN
University of Applied Sciences

Einführung in Maschinelles Lernen

Bachelor Medieninformatik
Wintersemester 2019/20

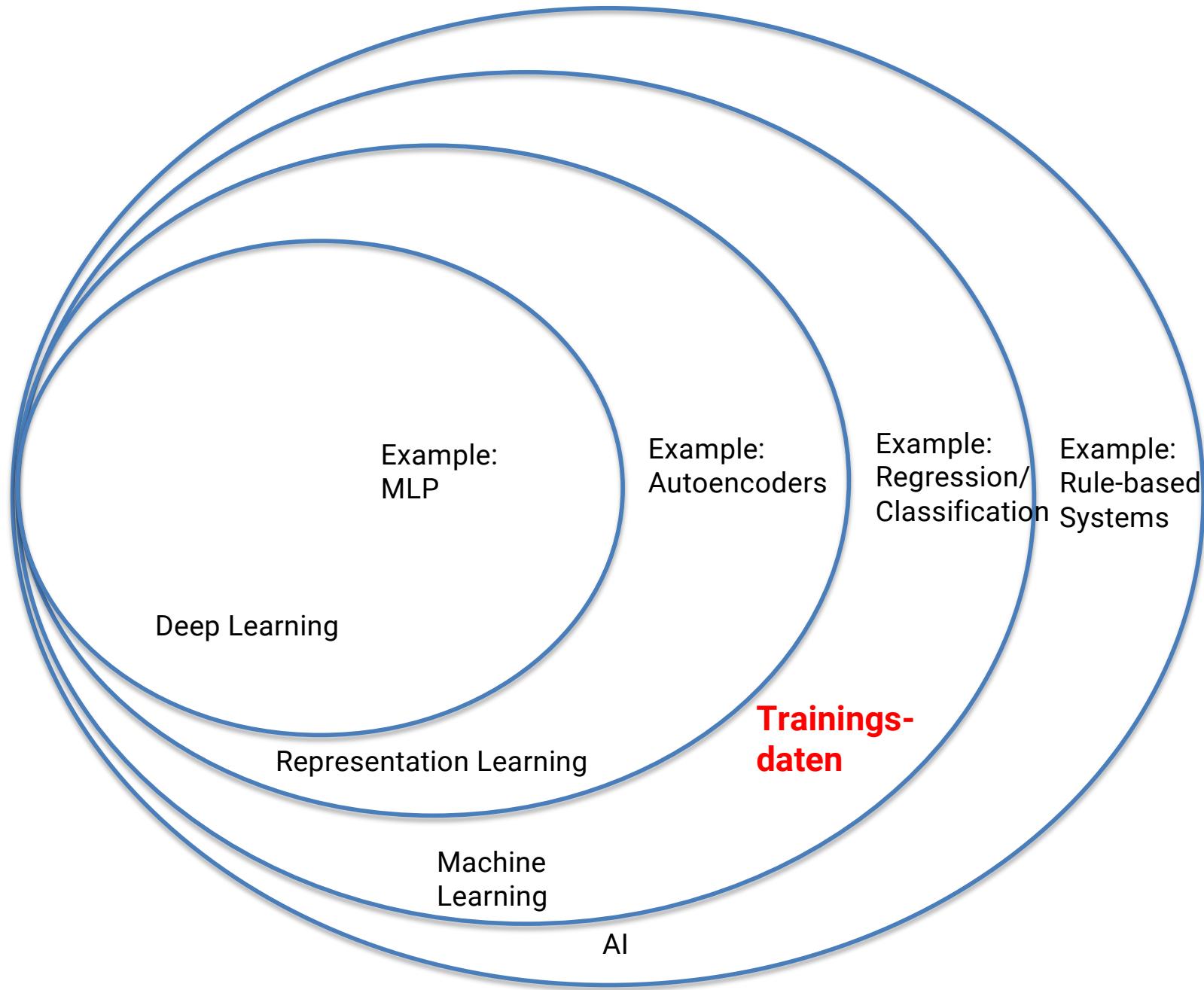
Prof. Dr.-Ing. Kristian Hildebrand
khildebrand@beuth-hochschule.de

Lernziele

- **Was ist Maschinelles Lernen?**
 - Motivation und Anwendungsgebiete
 - Supervised vs. Unsupervised Learning
 - Features
 - Repräsentation von Daten
- **Unsupervised Learning Techniken**
 - Clustering am Beispiel von k-means
- **Supervised Learning**
 - K-Nächste Nachbarn Suche
 - Support Vector Maschines
 - Neural Networks

Was ist Maschinelles Lernen?

Was ist Maschinelles Lernen?



Sinnvolle Ressourcen

- <https://cs231n.github.io/> (stanford course)
- <http://www.deeplearningbook.org/> (Bibliothek)
- <https://blog.openai.com>

Machine Learning als Software 2.0

- Einfacher Daten zu sammeln als ein Programm zu schreiben
- Job künftiger Programmierer beinhaltet **sammeln, säubern, manipulieren, kategorisieren, analysieren and visualisieren Daten** die in Neuronale Netze eingespeist werden.
- Software 2.0 ersetzt 1.0 nicht, übernimmt aber in vielen Bereichen wesentliche Teile:
 - Visual recognition
 - Speech recognition
 - Speech Synthesis
 - Machine Translation
 - Robotics
 - Games

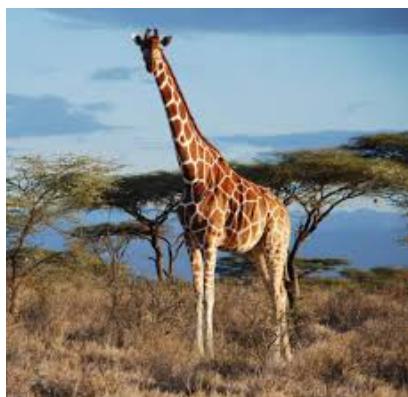


Was ist Maschines Lernen?



Lernen von Beispielen

Non-Acerous



Lernen eines Models auf Basis von Beispieldaten

Features und Labels

Was sind Features und wie transformieren wir Daten in Features?
(Musik, Text, Bild)

Features und Labels

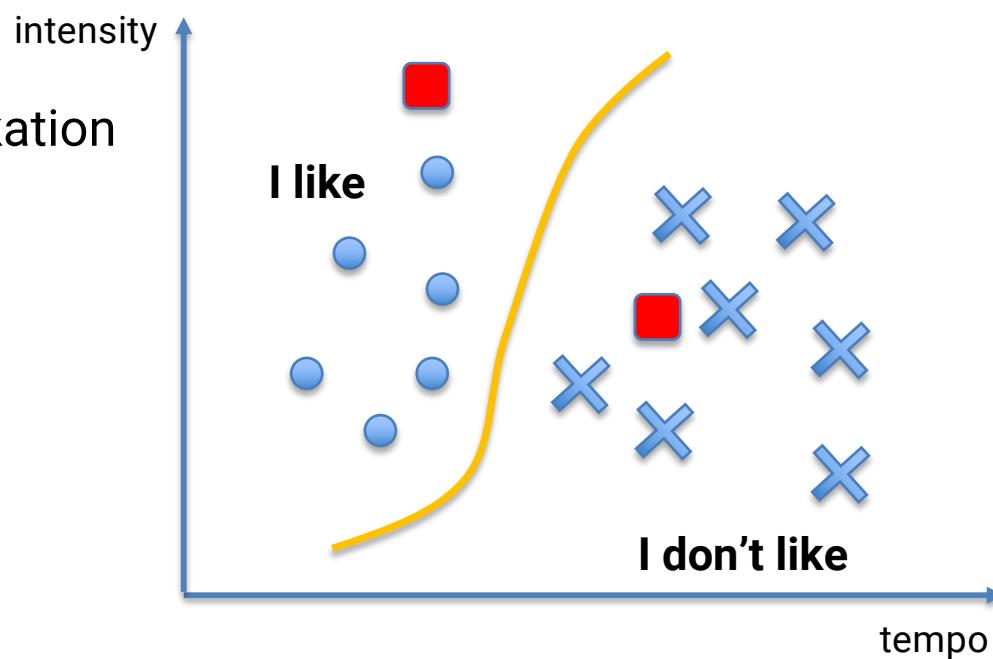


Features



Klassifikation

- intensity
- male/female singer
- tempo
- genre
- ...



Datentypen und Repräsentation

Wie repräsentieren wir Feature-Daten im Rechner?

Datentypen

- Numerische Daten, Messungen
 - floating point vectors $v = [0.1, 0.3, 0.4]$

Sample	Category	Numerical	Sample	Human	Penguin	Octopus	Alien
1	Human	1	1	1	0	0	0
2	Human	1	2	1	0	0	0
3	Penguin	2	3	0	1	0	0
4	Octopus	3	4	0	0	1	0
5	Alien	4	5	0	0	0	1
6	Octopus	3	6	0	0	1	0
7	Alien	4	7	0	0	0	1

- Kategorien
 - set $s = \{\text{bus, car, cat, dog...}\} \Rightarrow s = \{1,2,3,4...\}$

- Text
 - Häufigkeit der Worte im Text (histogram of word occurrences) – Bag of Words
 - Wörterbuch (dictionary)
 - word2vec

the dog is on the table



- Bild
 - Bildpixel als Datenvektor
 - Bildmerkmale als Datenvektor

Beispiele

Bag of words / Representation von Text

- Wie sieht der Eingabevektor aus?
- **Idee: Text-to-vector**

“John likes to watch movies. Mary likes movies too.”

“John also likes to watch football games.”

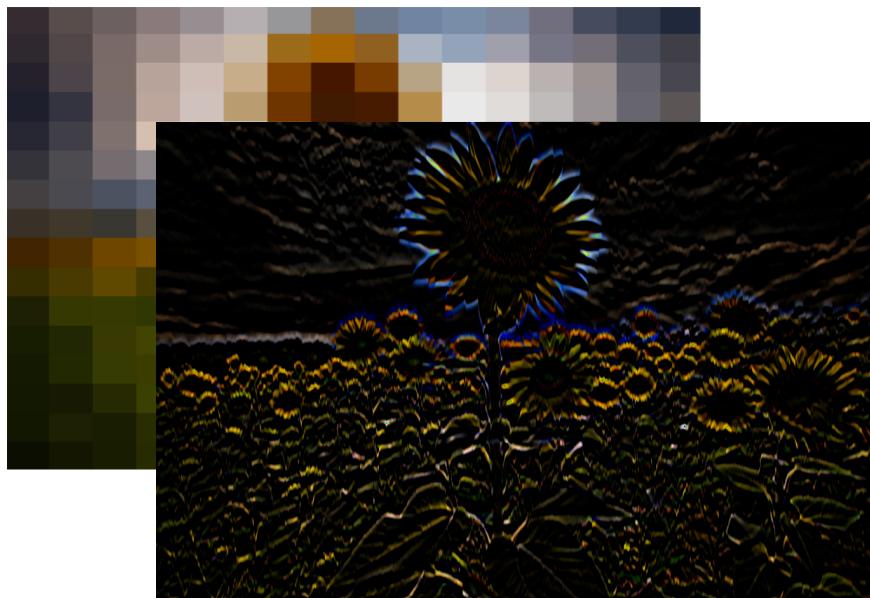
Bag of words

- **Eigenschaften:**
 - Reihenfolge der Wörter ist egal
 - unterschiedliche Länge des Textes gibt einen anderen Inputvektor (ABER: nicht in der Länge, nur in der Anzahl der Einträge im Histogramm)
 - Wortzusammenhänge können darüber zunächst nicht erkannt werden

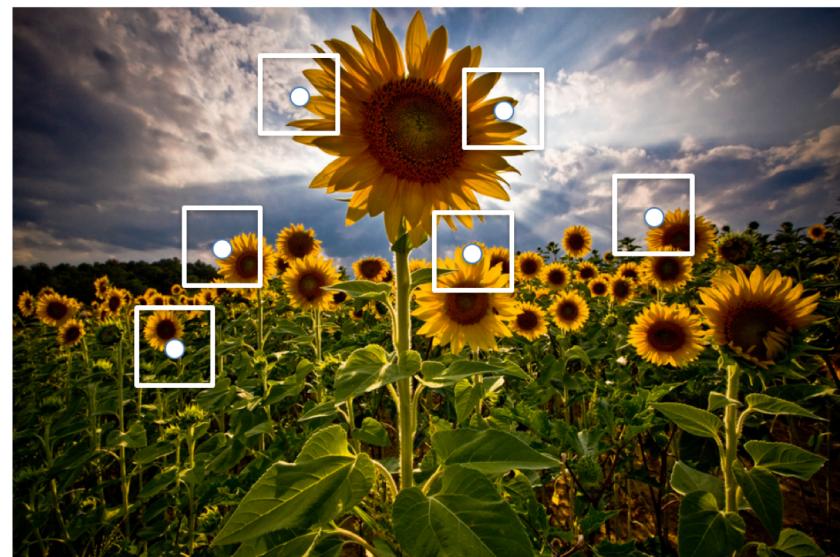
Bildmerkmale / Deskriptoren



Globaler Deskriptor

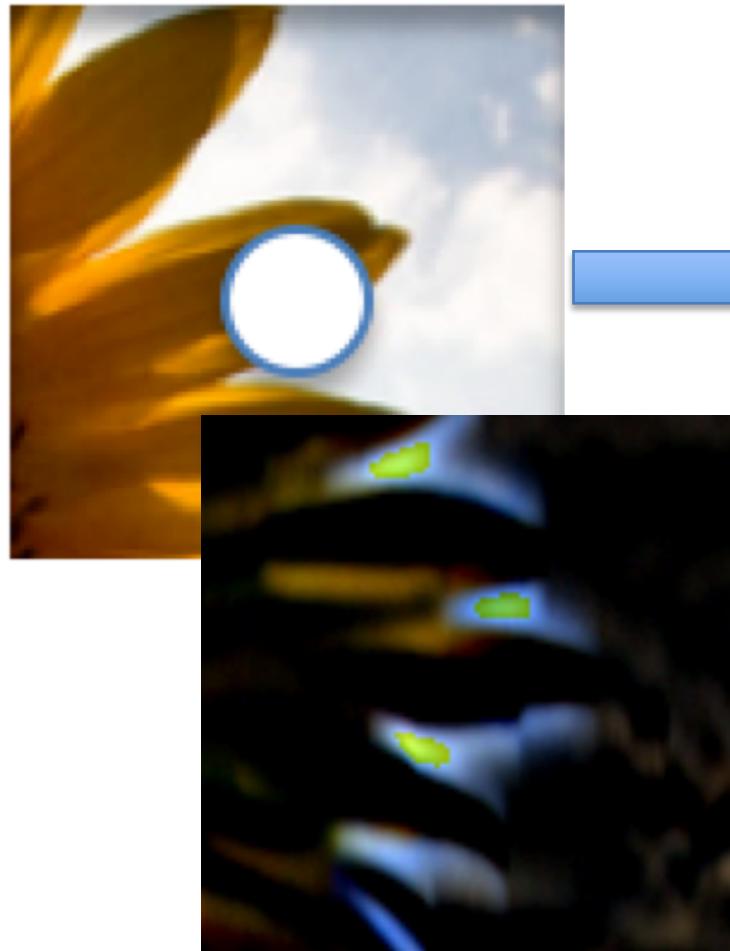


Lokaler Deskriptor

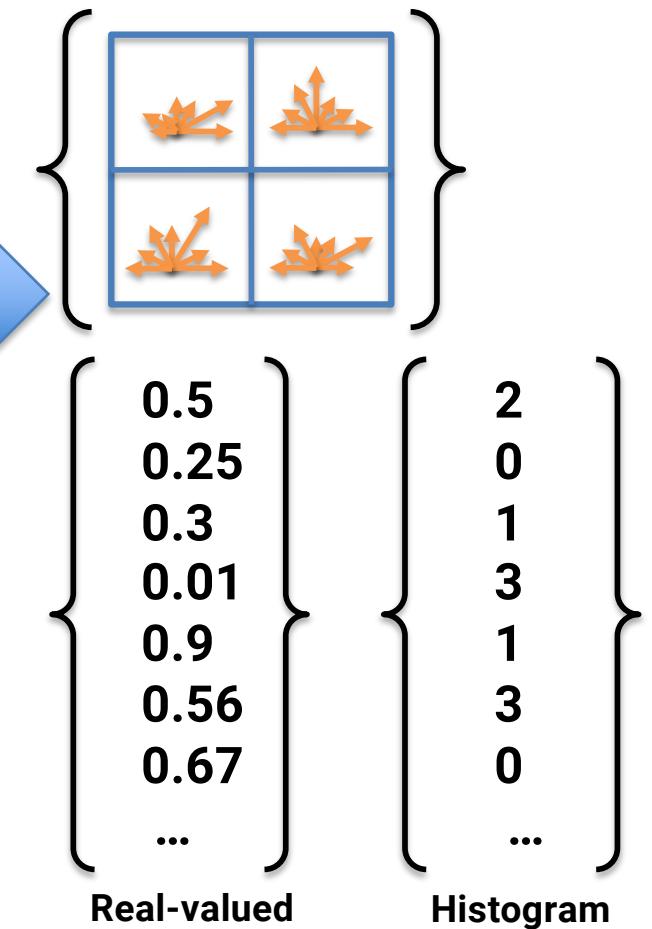


Bildmerkmale

Was ist der Vorteil eines lokalen Bildmerkmals?



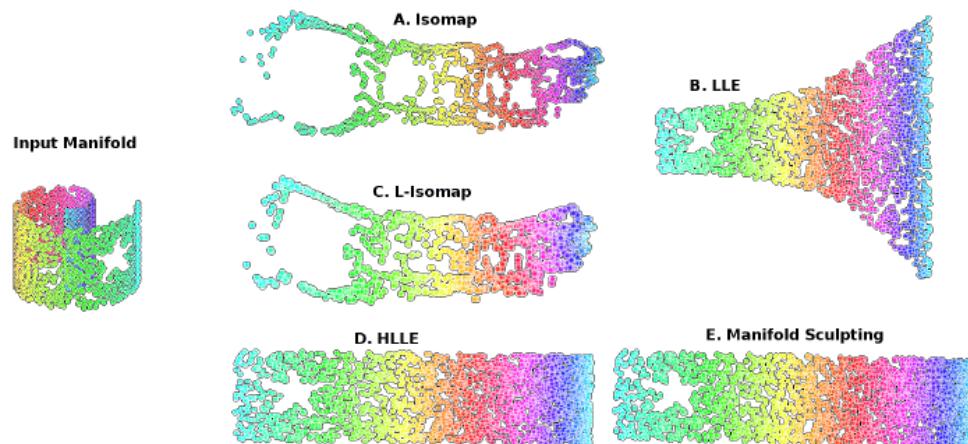
Merkmal-
beschreibung



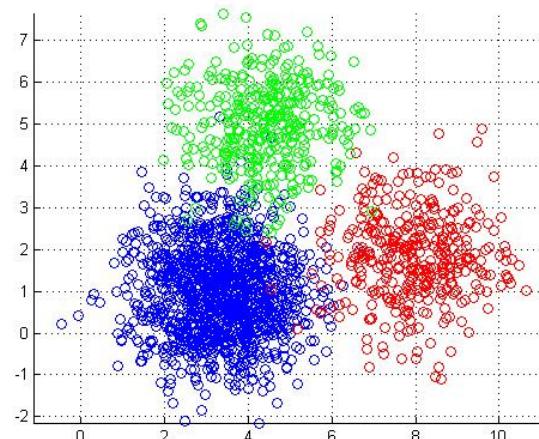
Machine Learning

Unsupervised Learning

- Dimensionality reduction
 - Principal Component Analysis (PCA)



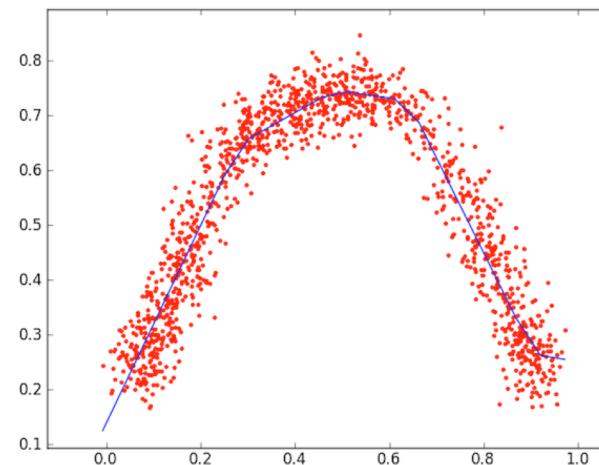
- Clustering
 - K-Means
 - Mean Shift



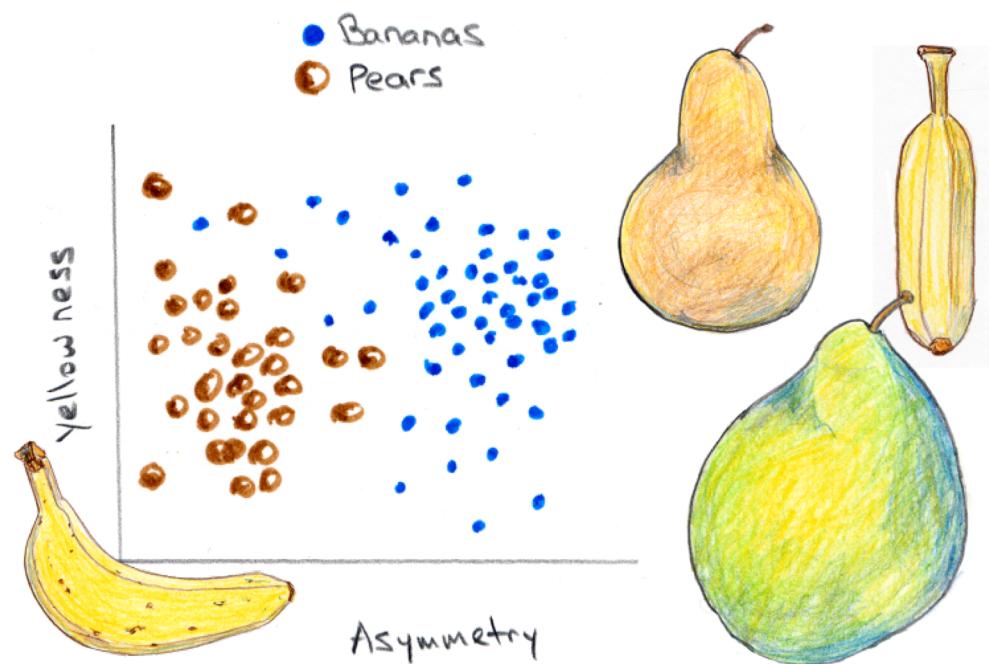
Machine Learning

- **Regression**
 - Least Squares
 - Neural Networks

Supervised Learning

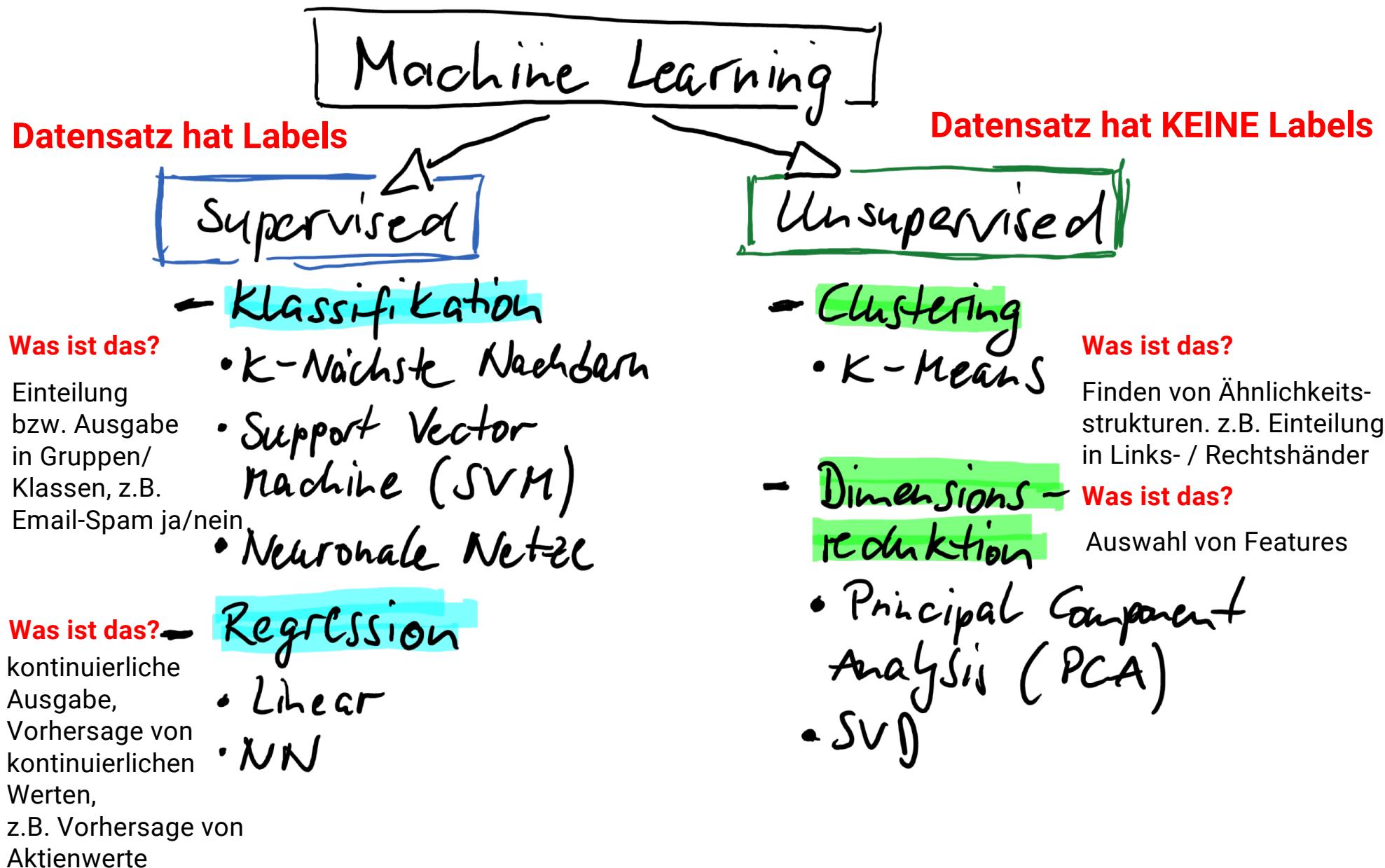


- **Classification**
 - Support Vector Machines
 - Neural Networks

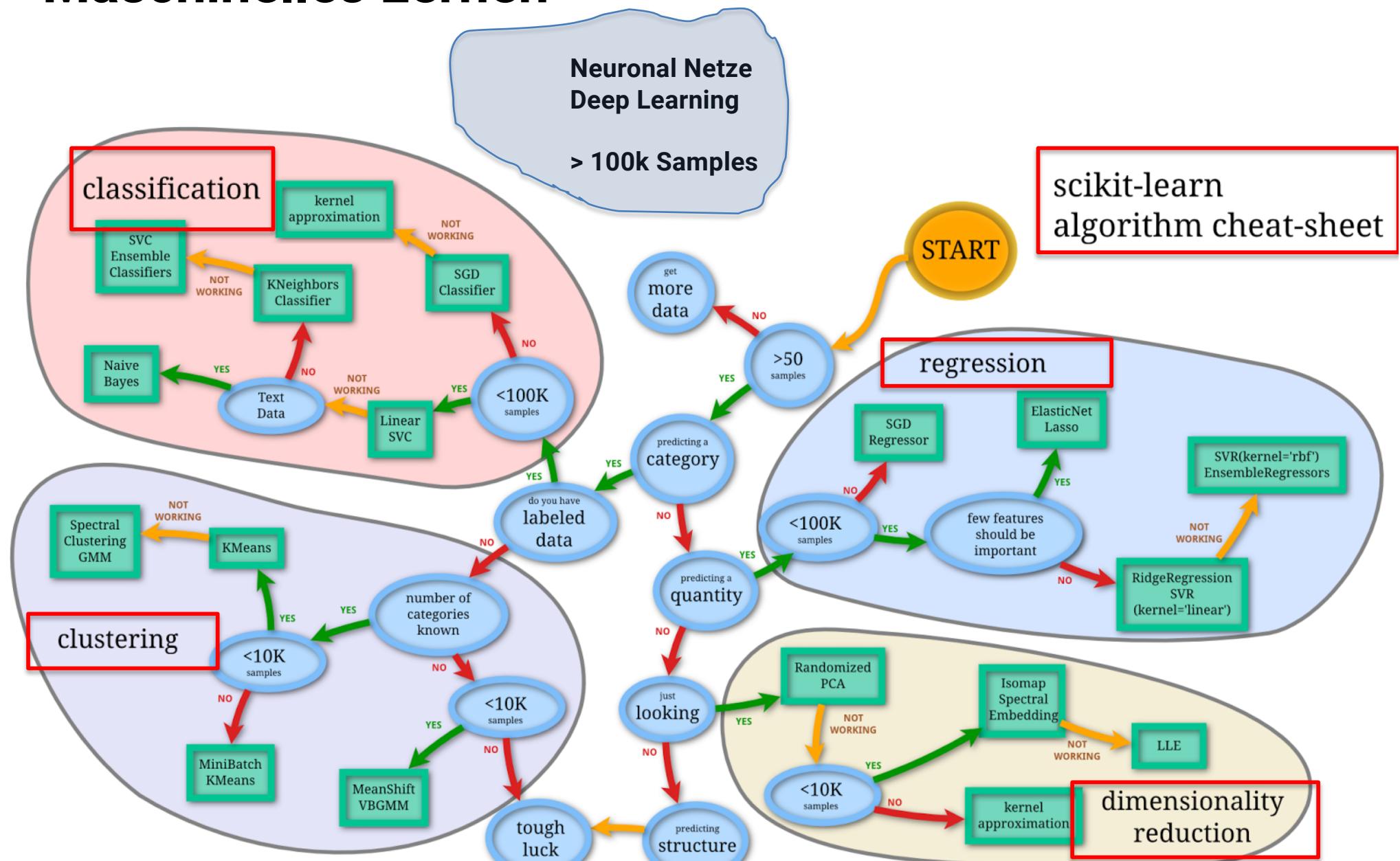


Supervised vs. Unsupervised

Beispiel:
Iris – Dataset

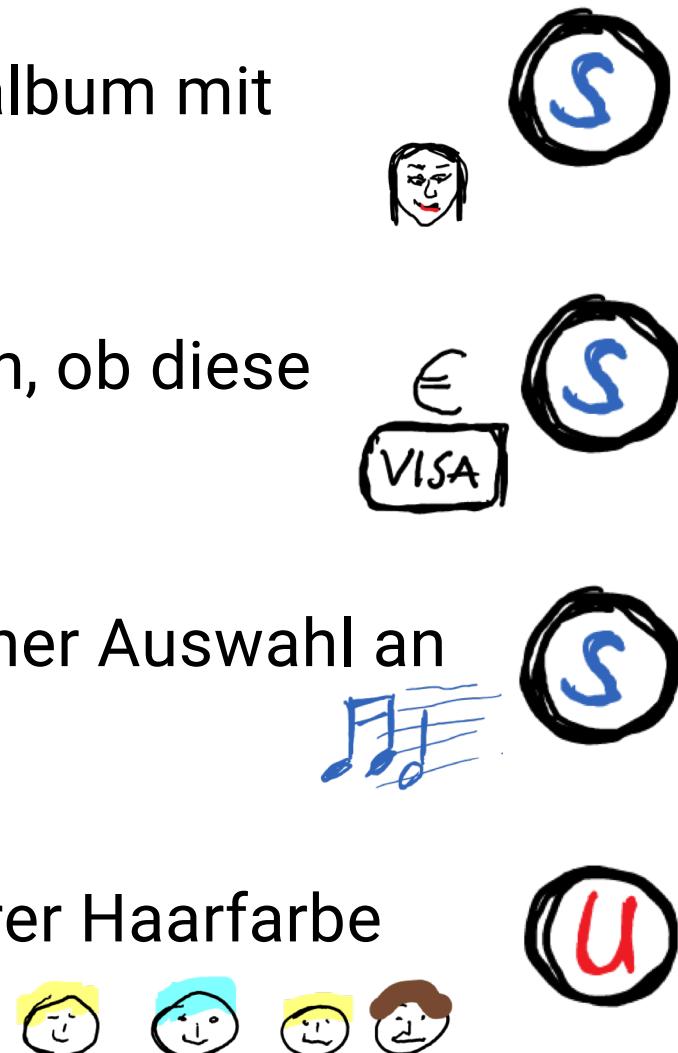


Maschinelles Lernen



Supervised vs. Unsupervised

- Erkennen einer Person in einem Fotoalbum mit gelabelten Fotos
- Analyse von Kreditkartenabbuchungen, ob diese Betrug sind oder nicht
- Vorschlag für einen Song, gegeben einer Auswahl an Songs, die eine Person gut findet
- Gruppierung von Personen anhand Ihrer Haarfarbe



Datenvorverarbeitung / Normalisierung

Wie ist die Hosengröße von Troubadix?

70 kg

5,95 ft



Berechnung über Größe + Gewicht
L oder S?

65 kg

5,41 ft

S

85 kg

6,1 ft

L



Größe + Gewicht

Feature Scaling

Normalisierung

- Max-/Min-Skalierung $x' = \frac{x - x_{min}}{x_{max} - x_{min}}$

Unsupervised Learning am Beispiel von k-means

Unsupervised Learning

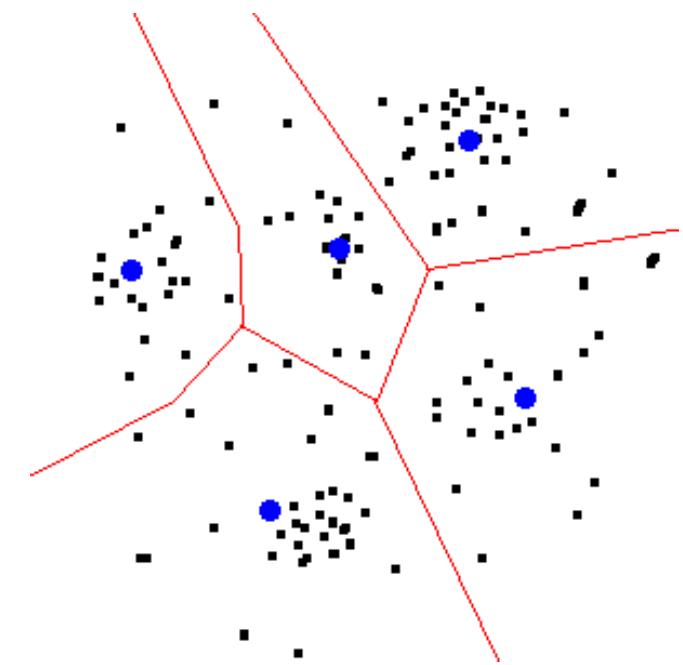
- Gegeben Datensatz ohne Labels (eine Klasse)
 - Nützliche Informationen extrahieren (Struktur des Datensatzes)

k-means

- wichtige Clustering-Methode mit häufigem Einsatz in vielen Gebieten der Informatik
- Ziel: Aufteilung der **N Datenpunkte** in **k Cluster $S = \{S_1, S_2, \dots S_k\}$**
- Jeder Datenpunkt gehört zu dem Cluster-Zentrum mit kürzestem Abstand
 - Zentrum/Schwerpunkt/Durchschnitt eines Clusters ist das Clusterzentrum

$$\arg \min_s \sum_{i=1}^k \sum_{x \in s_i} \|x - \mu_i\|^2$$

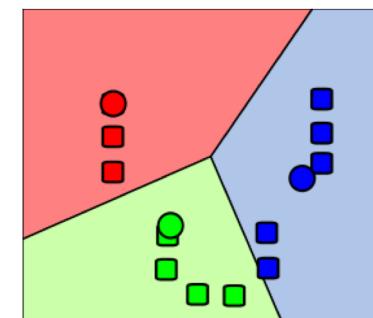
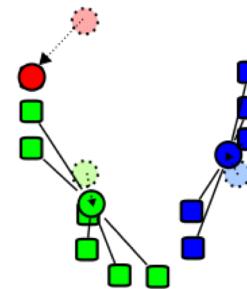
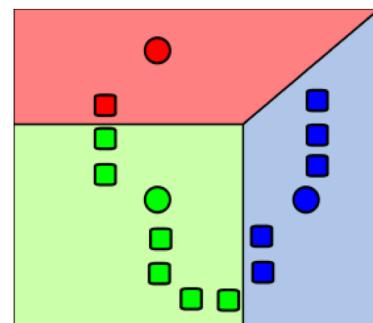
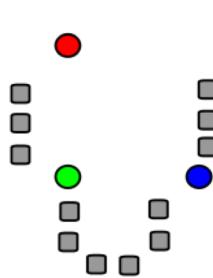
- auch unter **Lloyd's Algorithmus bekannt**



<http://mnemstudio.org/>

k-means Algorithmus

- **Schritte:**
 1. **Initialisierung:** Zufällige Zuweisung von k Datenpunkten als Cluster-Zentrum
 2. **Zuweisungs-Schritt:** Weise jedem Datenpunkt das Cluster mit dem nächstgelegenen Cluster-Zentrum zu
 3. **Optimierungsschritt:** Berechne das neue Clusterzentrum aus allen Datenpunkten einer Clusters
- **Iteriere Schritt 2 /3** bis sich die Clusterzentren nicht mehr ändern



Supervised Learning

Sketch Classification



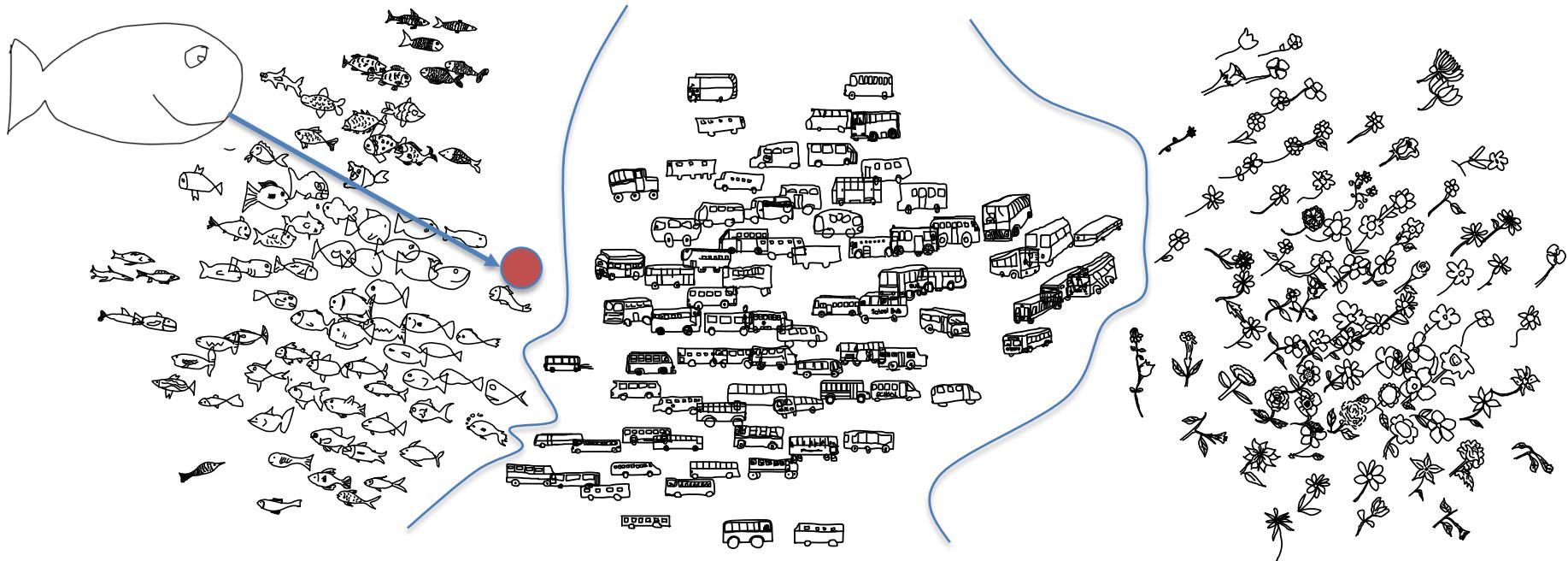
Decision Surface (Klassifikation)

- **Gegeben ist Datensatz mit Label**
 - z.B. 20,000 Handskizzen aus 250 Kategorien

$$X = \left\{ \begin{array}{c} \text{Fisch} \\ \text{Bus} \\ \text{Blume} \end{array} \right\}$$

$Y = \{ \text{Fisch, Bus, Blume} \}$

$Y = \{ [1,0,0], [0,1,0], [0,0,1] \}$

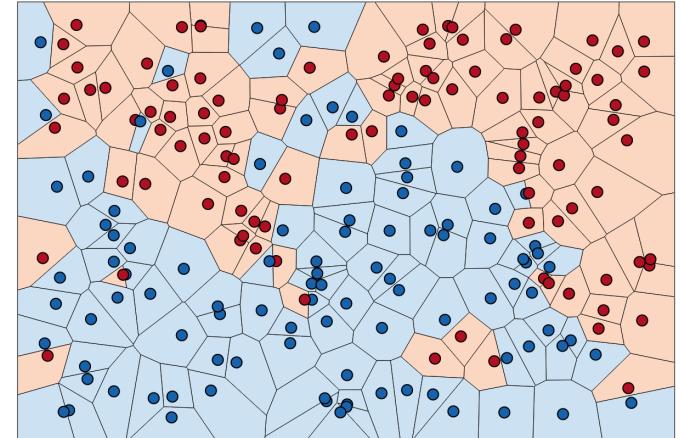


- **Ziel:** Erlernen einer Klassifikation, Finden von Trennungsflächen zwischen den Datenpunkten

K-Nächste Nachbarn

K-Nächste Nachbarn (KNN)

- **vielseitiger und robuster Klassifikator**
 - liefert häufig gute Vergleichswerte
 - sehr einfach zu implementieren
- Gegeben ein **gelabelter Datensatz** (x_i, y_i) fürs Training
- **Ziel:** Funktion $h: X \rightarrow Y$, sodass für unbekannte x eine passende Ausgabe y vorhergesagt wird
- Eigenschaften:
 - **nicht-parametrisch:** keine Annahmen über die Funktion und damit keine Gefahr dir darunterliegenden Daten falsch zu modellieren
 - **Instanz-basiert:** lernt kein Model, merkt sich nur die Trainingsinstanzen für die Vorhersage



<http://scott.fortmann-roe.com/docs/BiasVariance>

KNN - Wie funktioniert es?

- **Idee:** Wenn die Mehrheit meiner k–Nachbarn zu Klasse A gehört, muss x auch zu A gehören
- Ähnlichkeit wird dabei als z.B. euklidische Distanz berechnet:

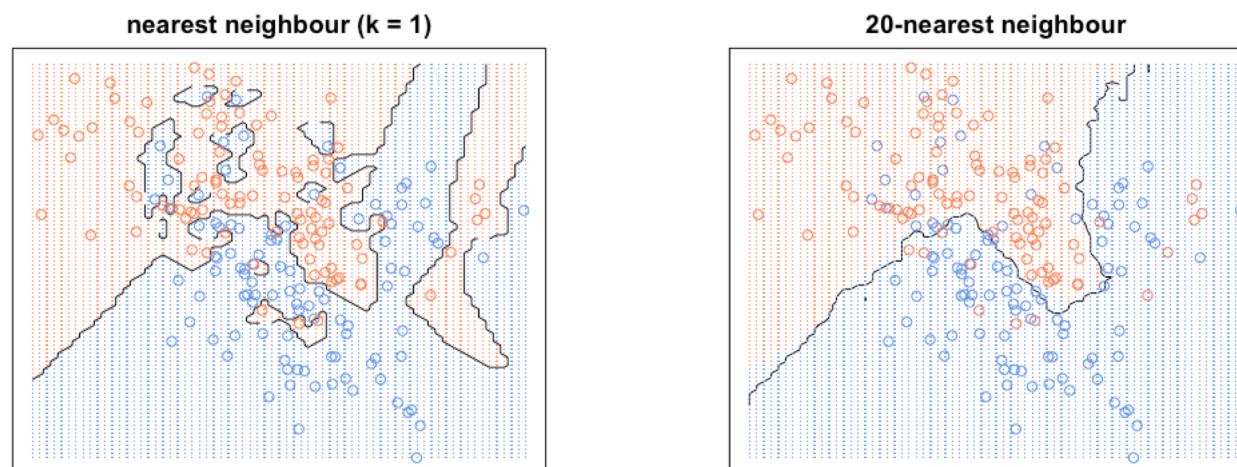
$$d(x, x') = \sqrt{(x_1 - x'_1)^2 + (x_2 - x'_2)^2 + \cdots + (x_n - x'_n)^2}$$

- **Gegeben:** Integer K, unbekannter Datenpunkt x und $d(x,x')$ werden folgende Schritte durchgeführt:
 - berechne $d(x,x')$ für alle Trainingsdaten x'
 - die nächsten K Punkte zu x sind das Vorhersageset A und bestimmen die Klasse

$$P(y = j | X = x) = \frac{1}{K} \sum_{i \in A} I(y^{(i)} = j)$$

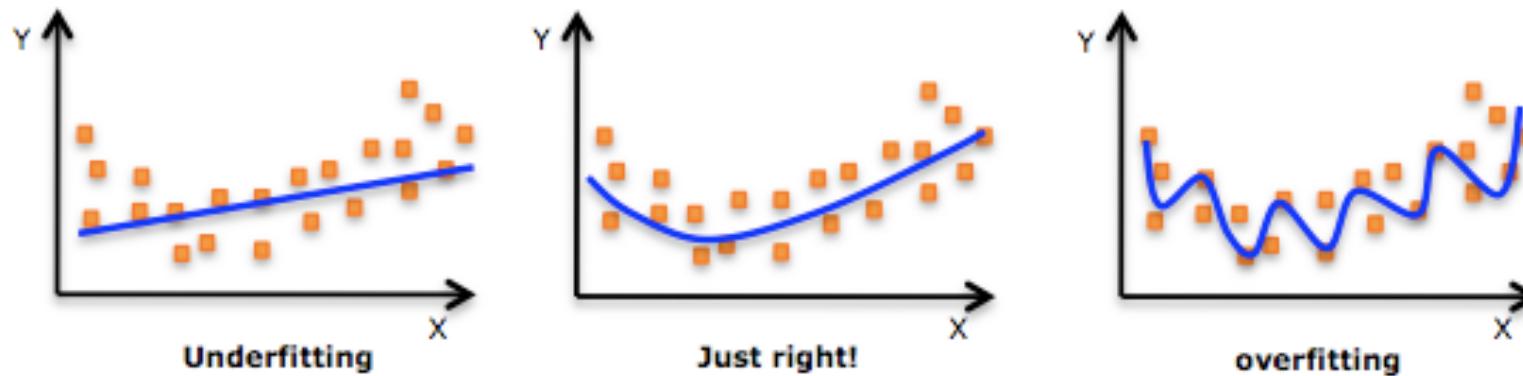
Hyperparameter und KNN

- K ist ein Hyperparameter, d.h. Parameter den man z.B. durch Probieren herausfinden muss
- Hyperparameter sind Parameter, die Training und Vorhersage maßgeblich beeinflussen
 - können nicht von den Daten abgeleitet werden
 - z.B. k in k-means, Lernrate, Anzahl der Hidden Layers in NN, etc
 - müssen durch ‘geschickte’ Auswahl oder Optimierung



Overfitting vs. Underfitting

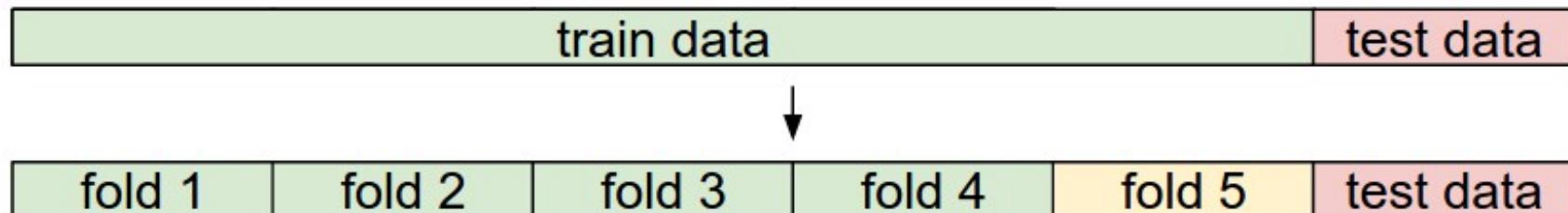
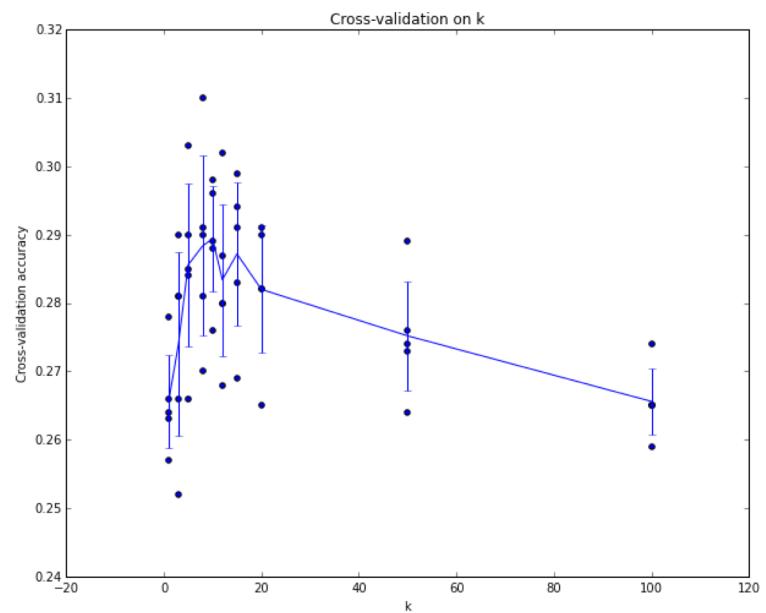
- Überanpassung an das Modell lernt auch des Rauschen in den Daten mit
 - generalisiert damit nicht ausreichend für unbekannte Daten
- Unteranpassung generalisiert zu stark, sodass es viele Fehlklassifikationen gibt



<http://pingax.com/regularization-implementation-r/>

Trainieren, Validieren und Testen

- Interface von Supervised Learning Algorithmen:
 - ***train(train_data, train_labels, params)***
 - ***predict(test_data, params)***
- **Cross Validation / Kreuzvalidierung**
 - Aufteilung des Datensatzes in Trainingsteile und Validierungsteile
 - Iteration über alle Teile und Durchschnitt der Genauigkeit über alle Teil



Maschinelles Lernen

- Ergebnisse hängen u.a. stark von folgenden Faktoren ab:
 - Größe und Qualität des Datensatzes
 - Ggf. gegebene Features
 - Vorverarbeitung der Daten (Normalisierung, Bereinigung etc.)
 - Wahl der Hyperparameter