Einführung in Data Science
Überwachtes Lernen – Block 3

PVA 3 – Programm

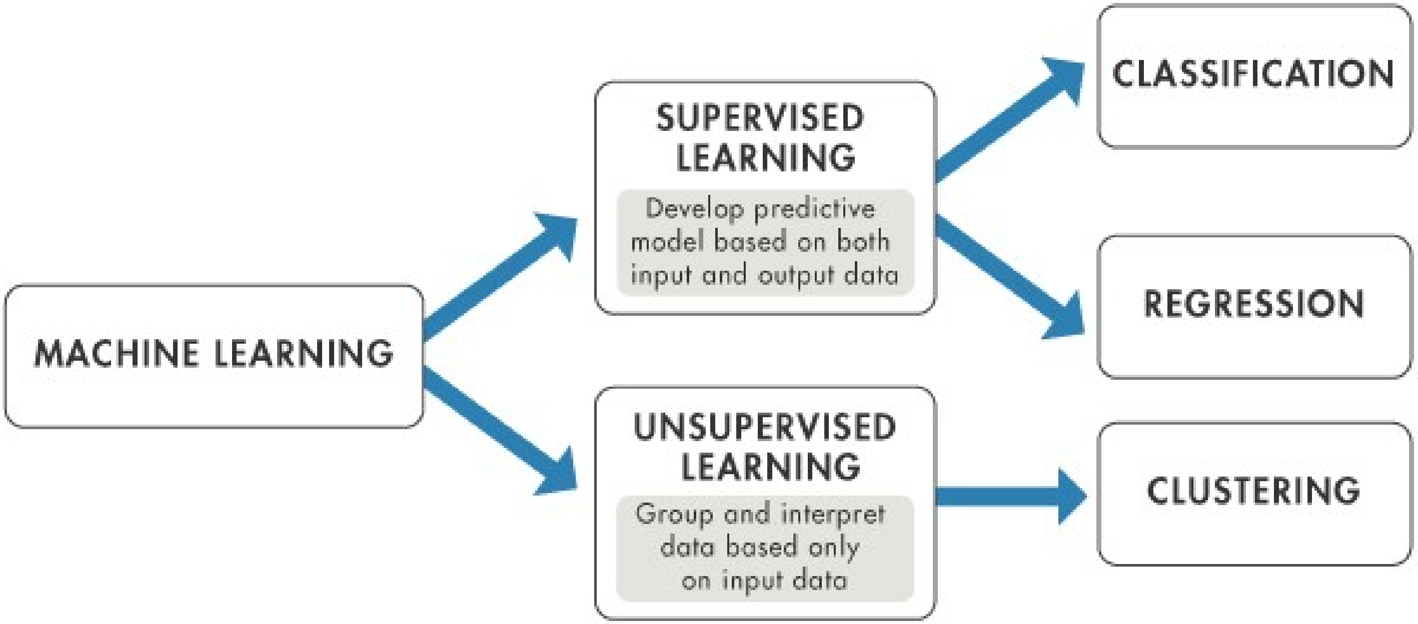
Thema	Form	Zeit
Besprechung der Semesterarbeit	Diskussion	13:45 – 14:00
Besprechung Vorbereitung	Diskussion	14:00 – 14:15
Supervised Learning	Vorlesung, Diskussion	14:15 – 15:00
Pause		15:00 – 15:15
Workshop 2	Gruppenarbeit	15:15 – 16:30
Workshop 1	Gruppenarbeit	16:30 – 17:00

Machine Learning

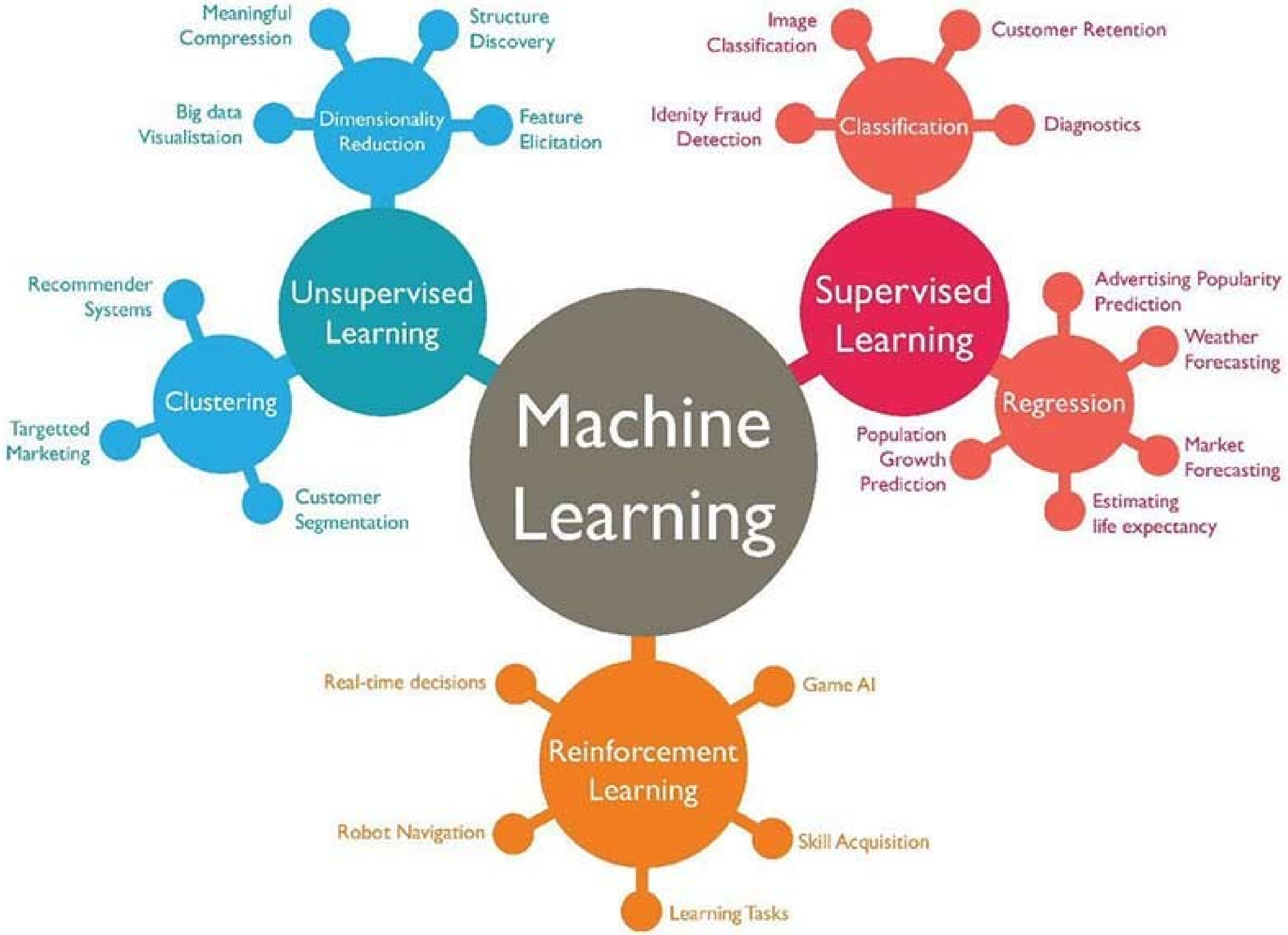
Definition

- Arthur Samuel (1959). ML: Field of study that gives computers the ability to learn without being explicitly programmed

Supervised Learning vs. Unsupervised Learning



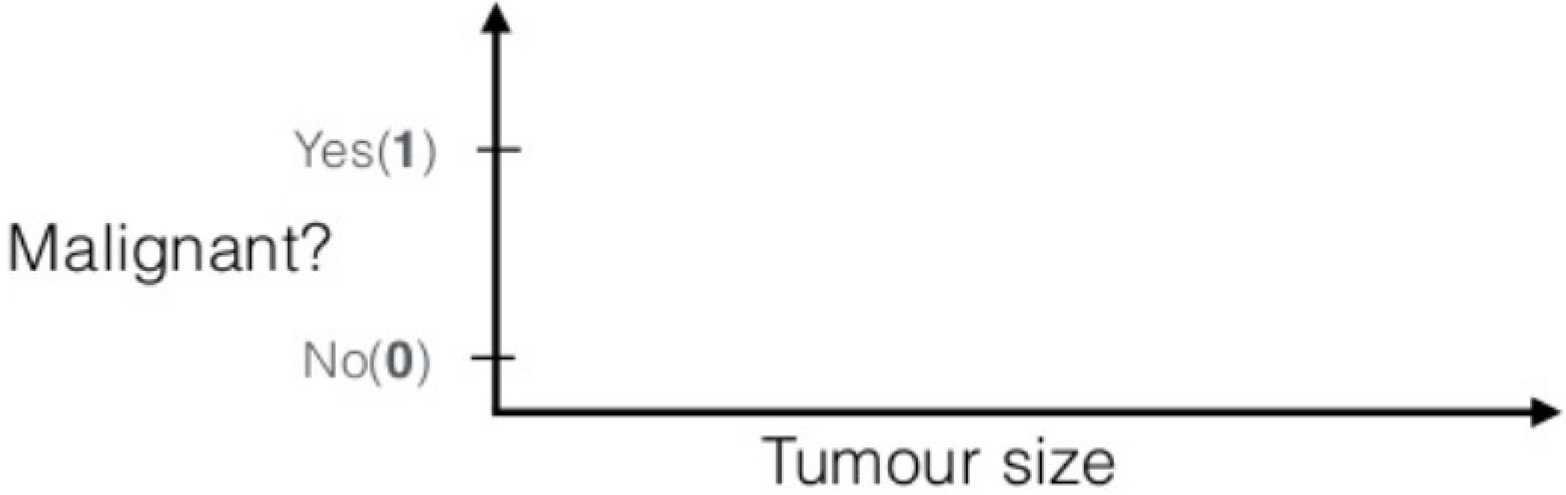
Supervised Learning vs. Unsupervised Learning



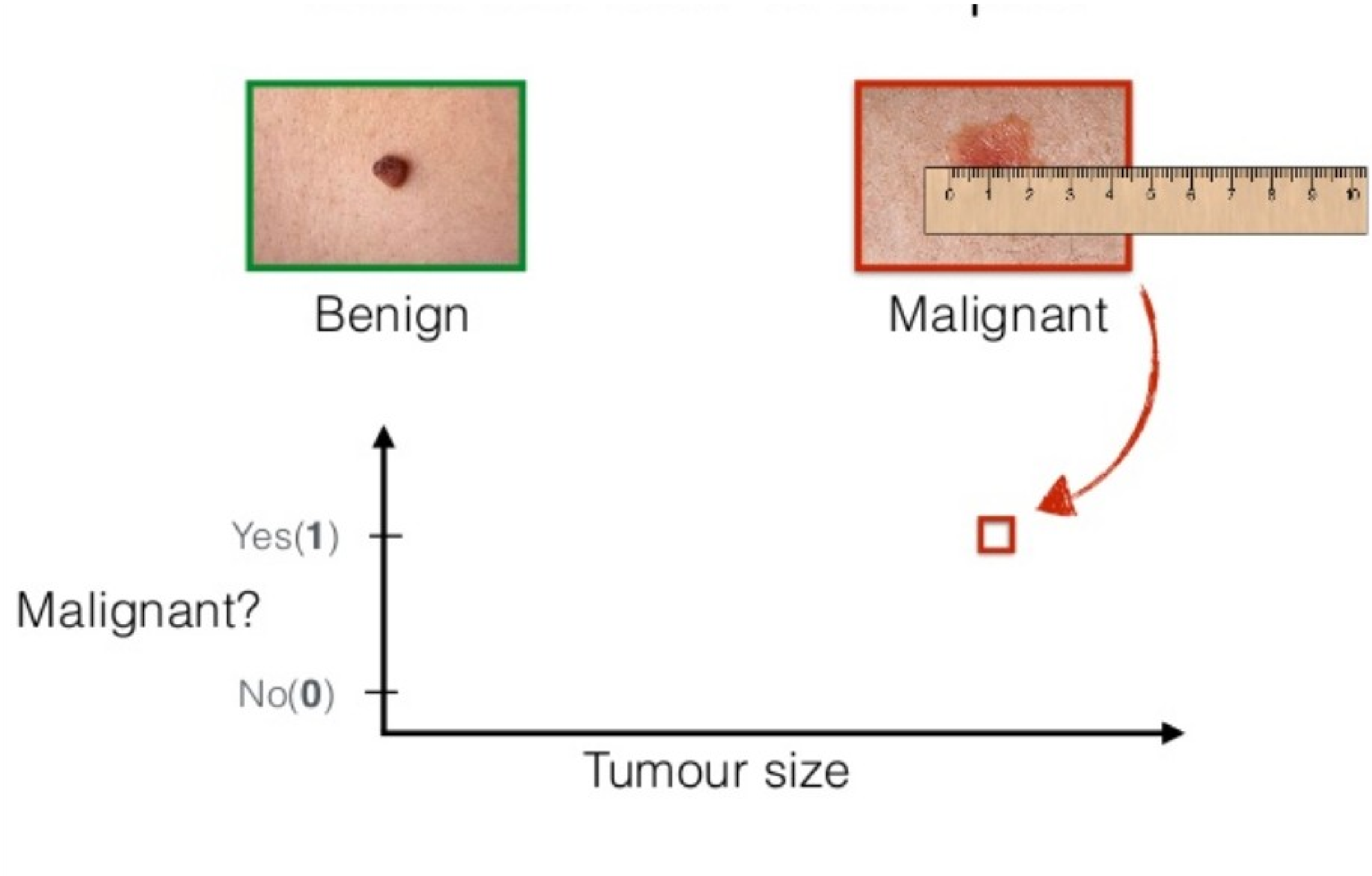
Beispiel Classification - Tumor

Benign

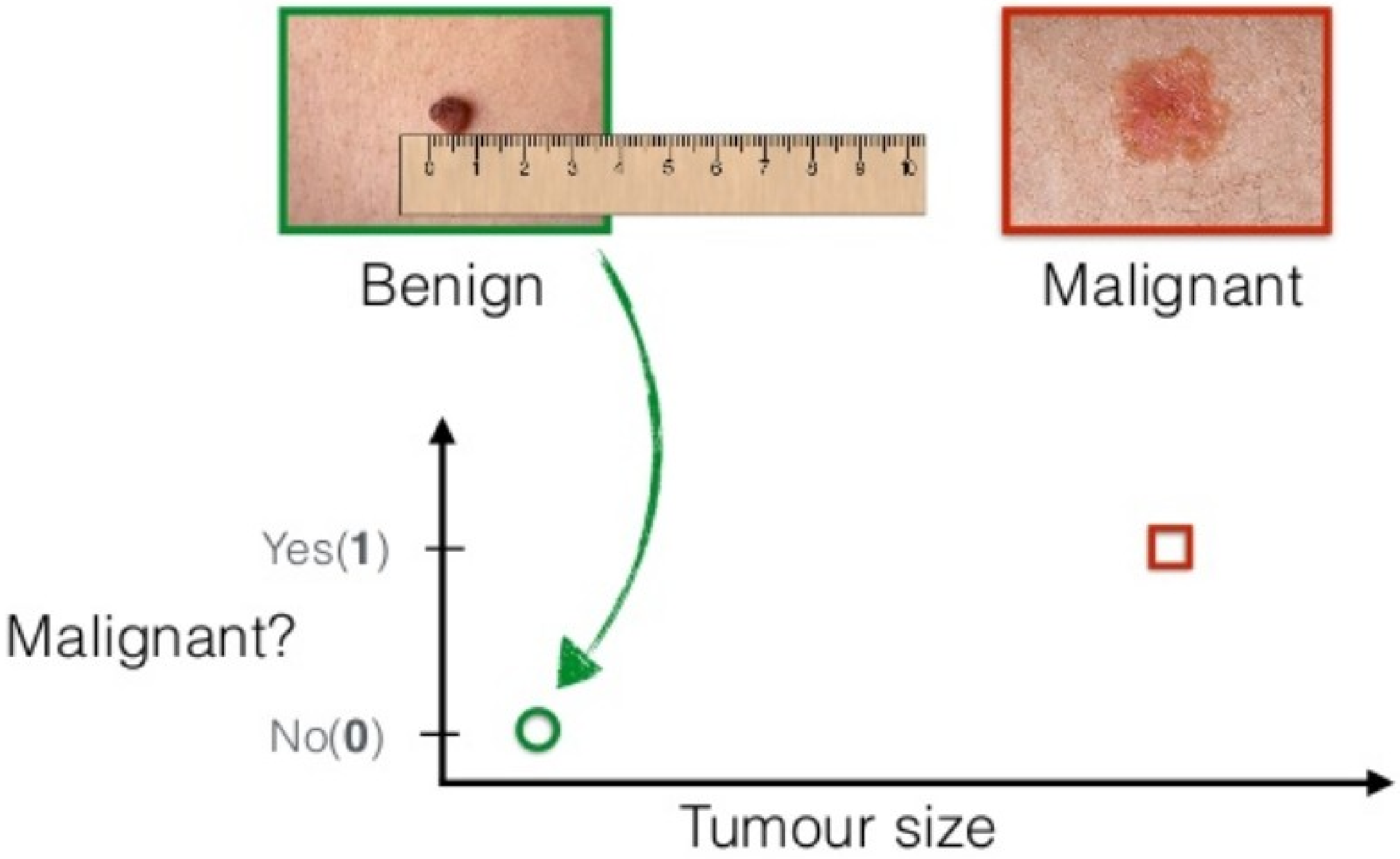
Malignant



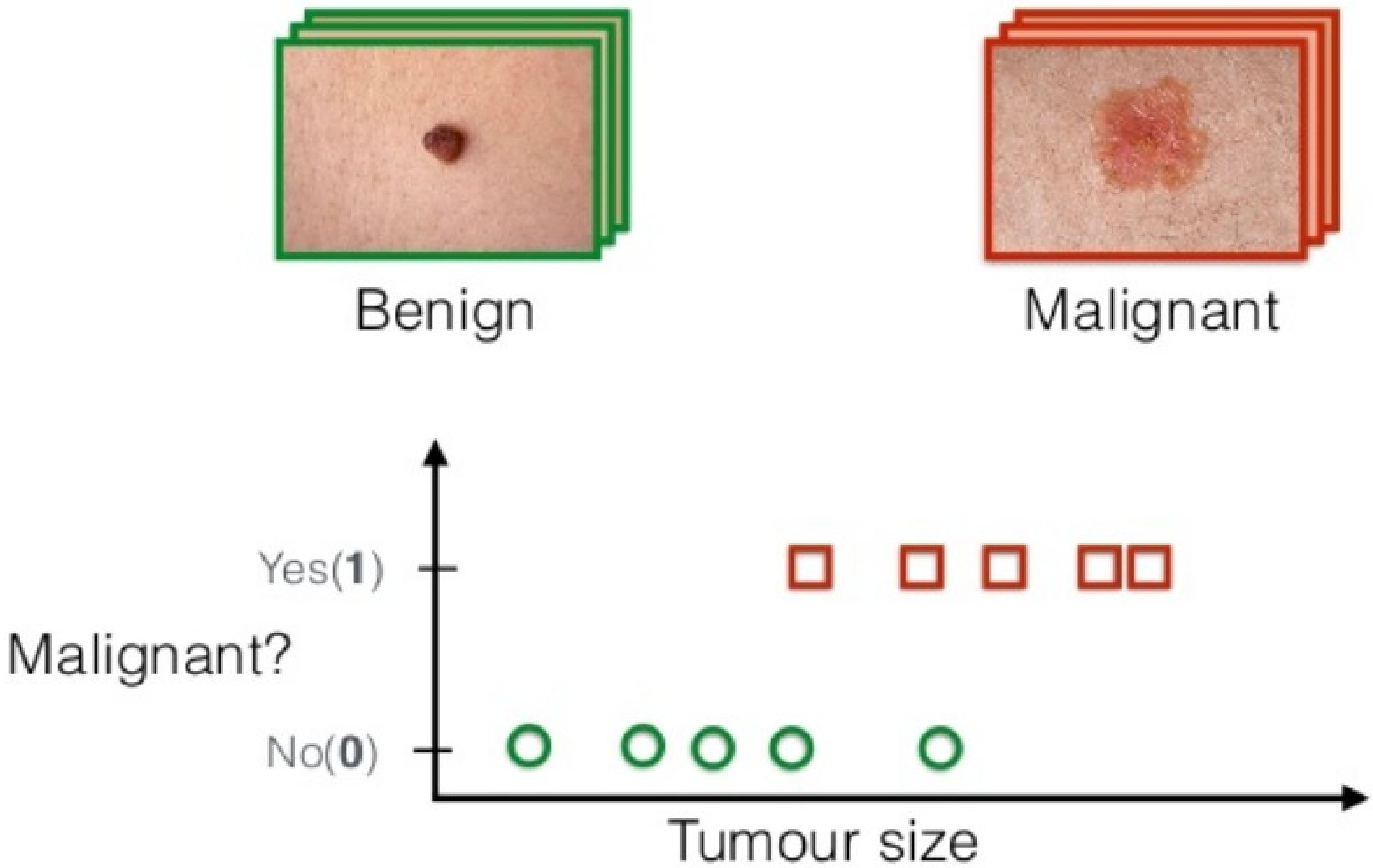
Beispiel Classification - Tumor



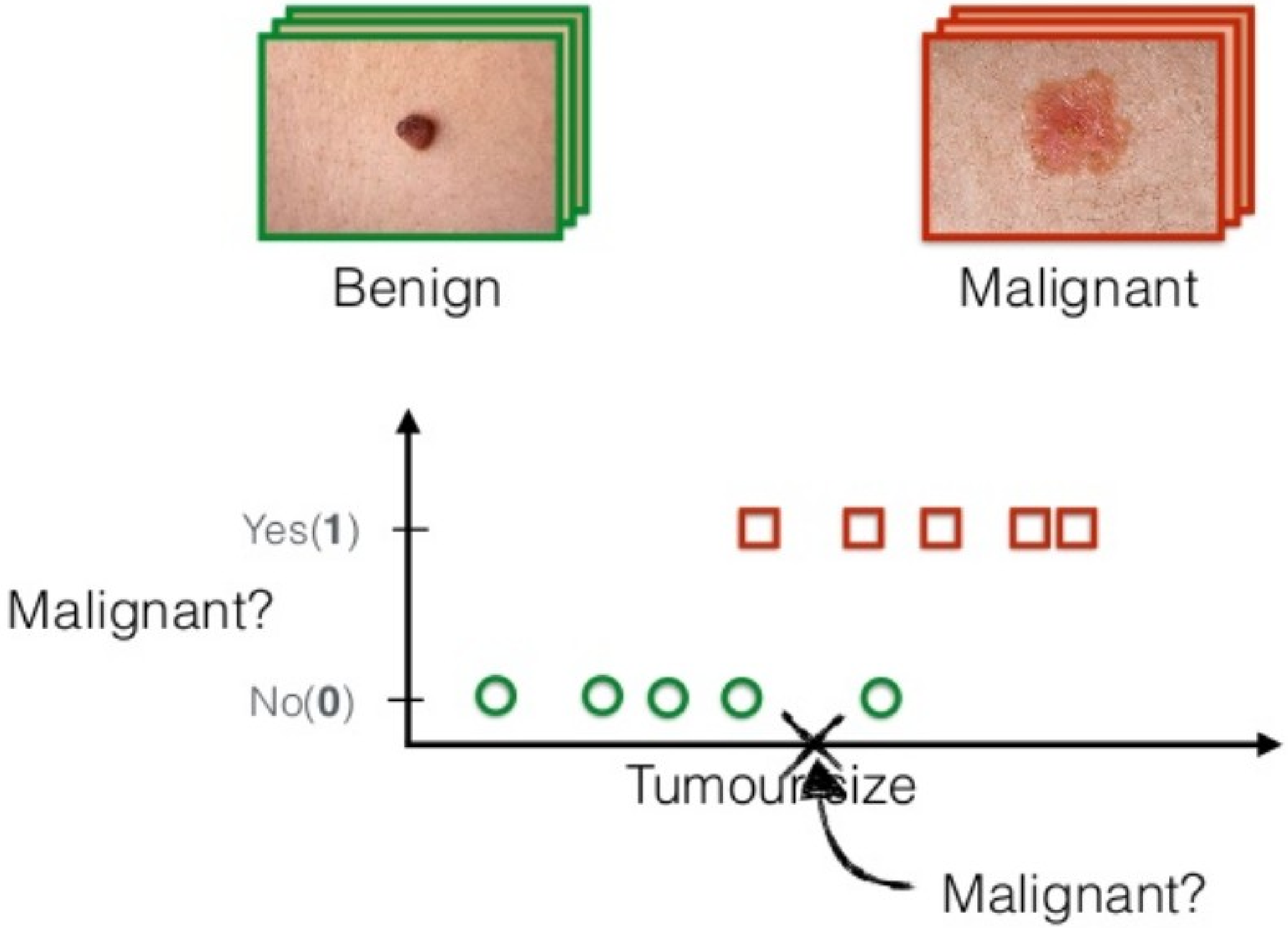
Beispiel Classification - Tumor



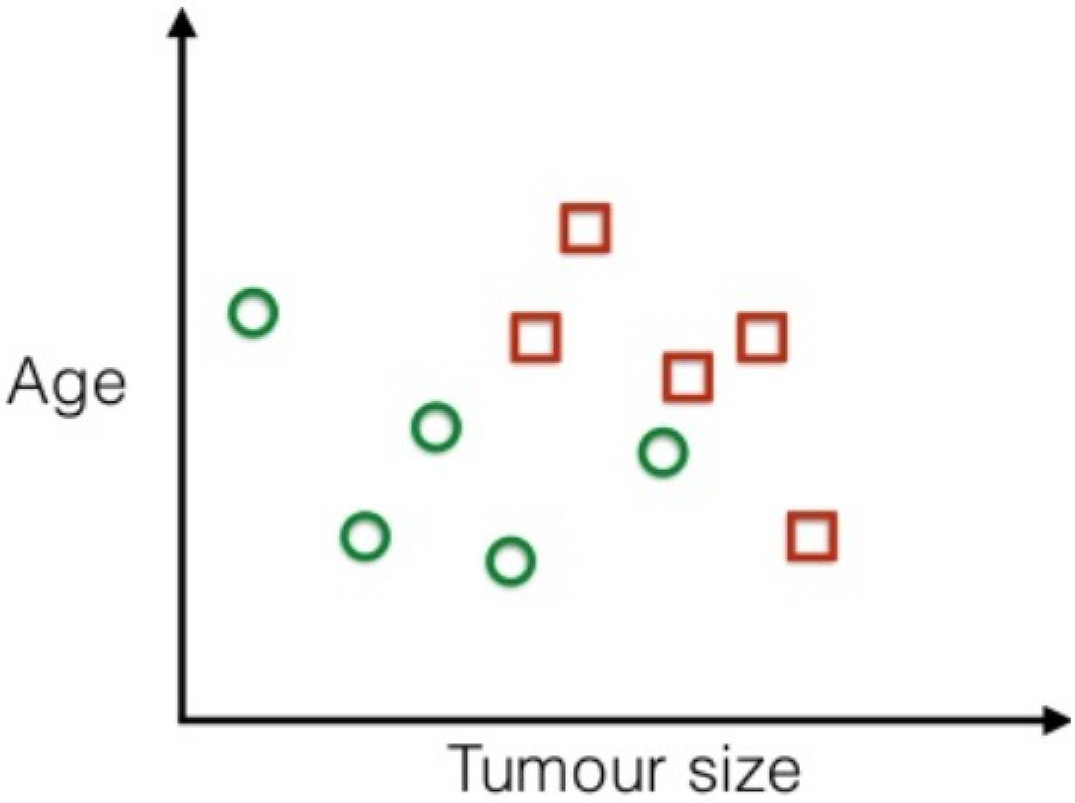
Beispiel Classification - Tumor



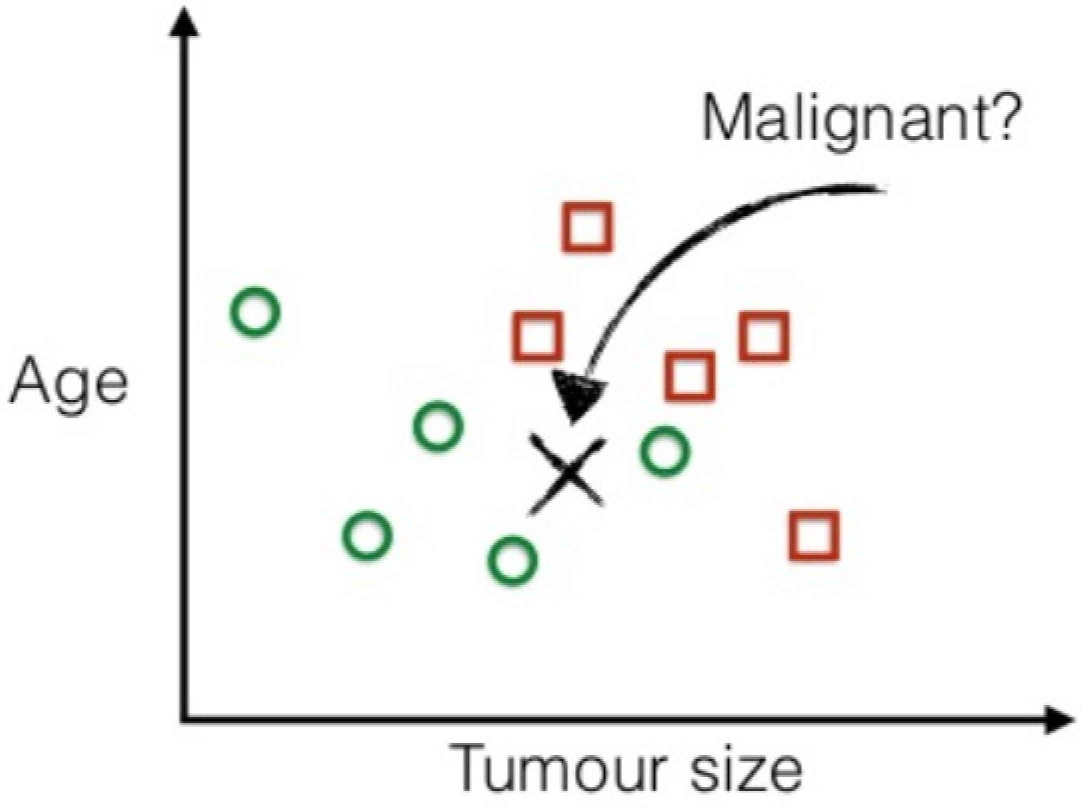
Beispiel Classification - Tumor



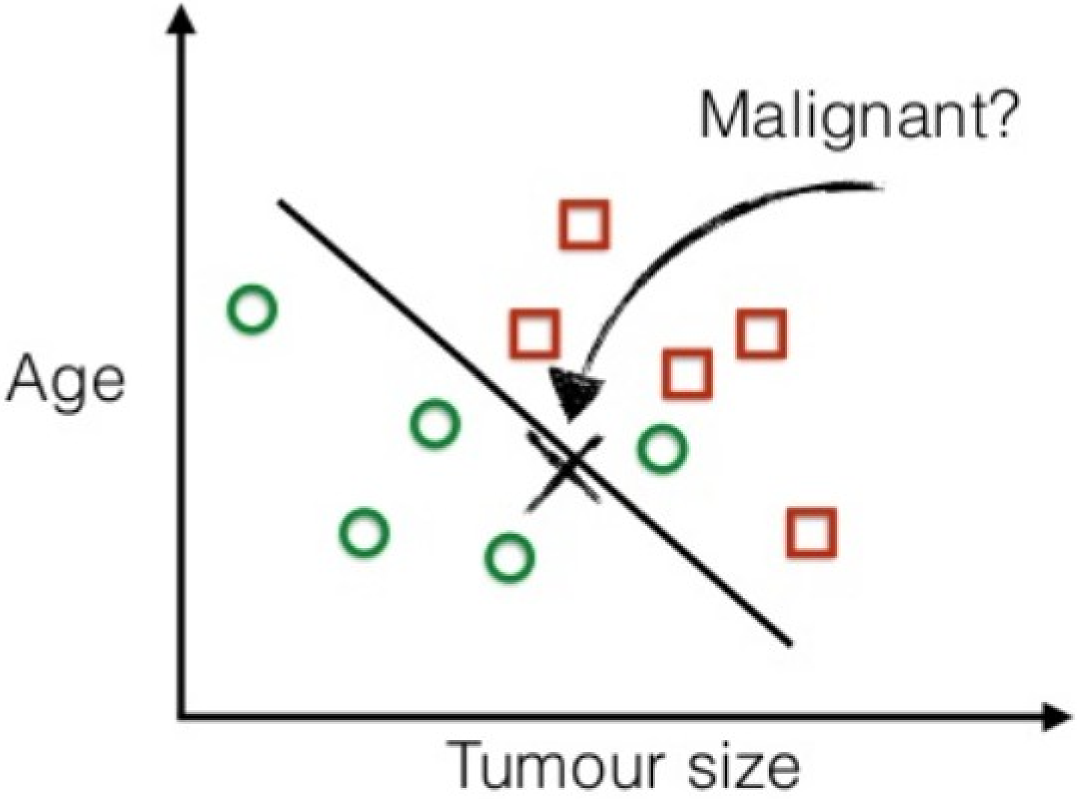
Beispiel mit mehreren Features



Beispiel mit mehreren Features



Beispiel mit mehreren Features

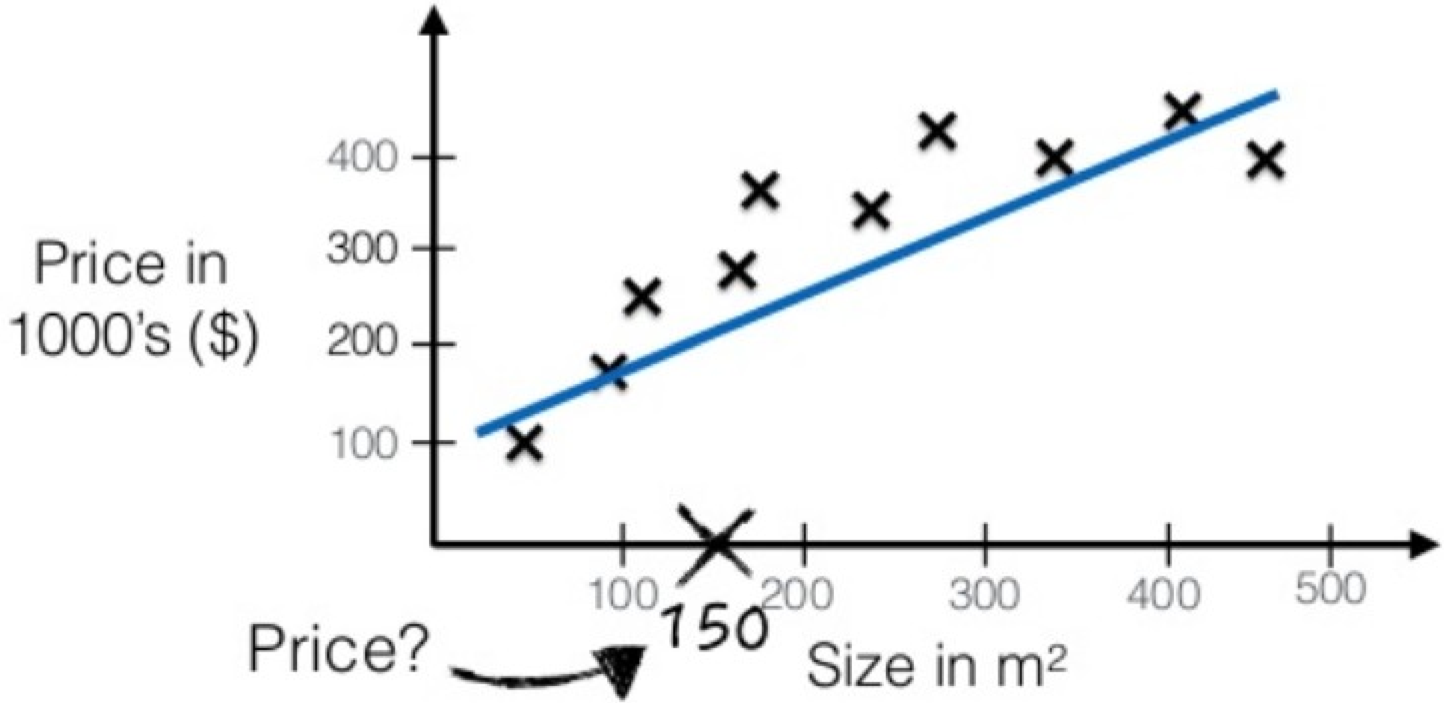


Noch mehr Features

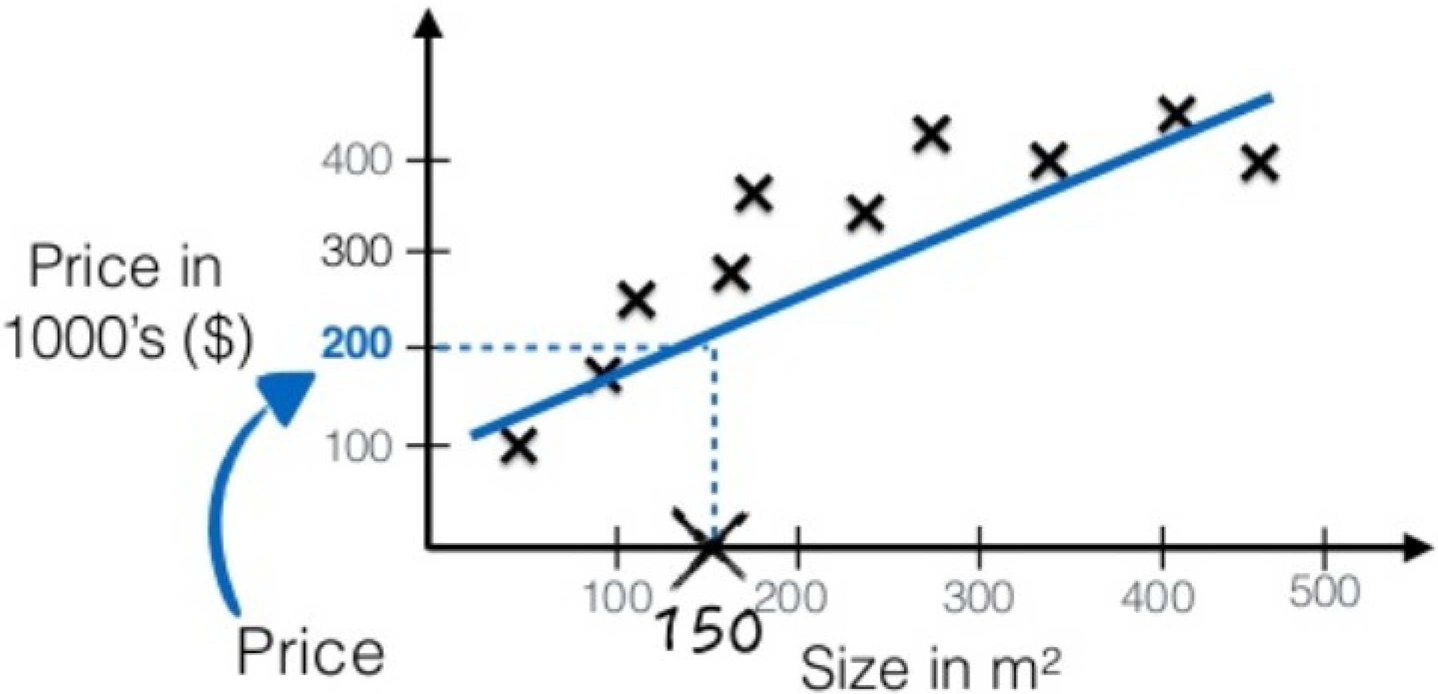
- Einheitlichkeit der Zellgrössen
- Einheitlichkeit der Zellformen
- Dichte
- …
- Lern-Algorithmen können mit theoretisch unendlich vielen Features umgehen

  → Hochdimensionale Räume

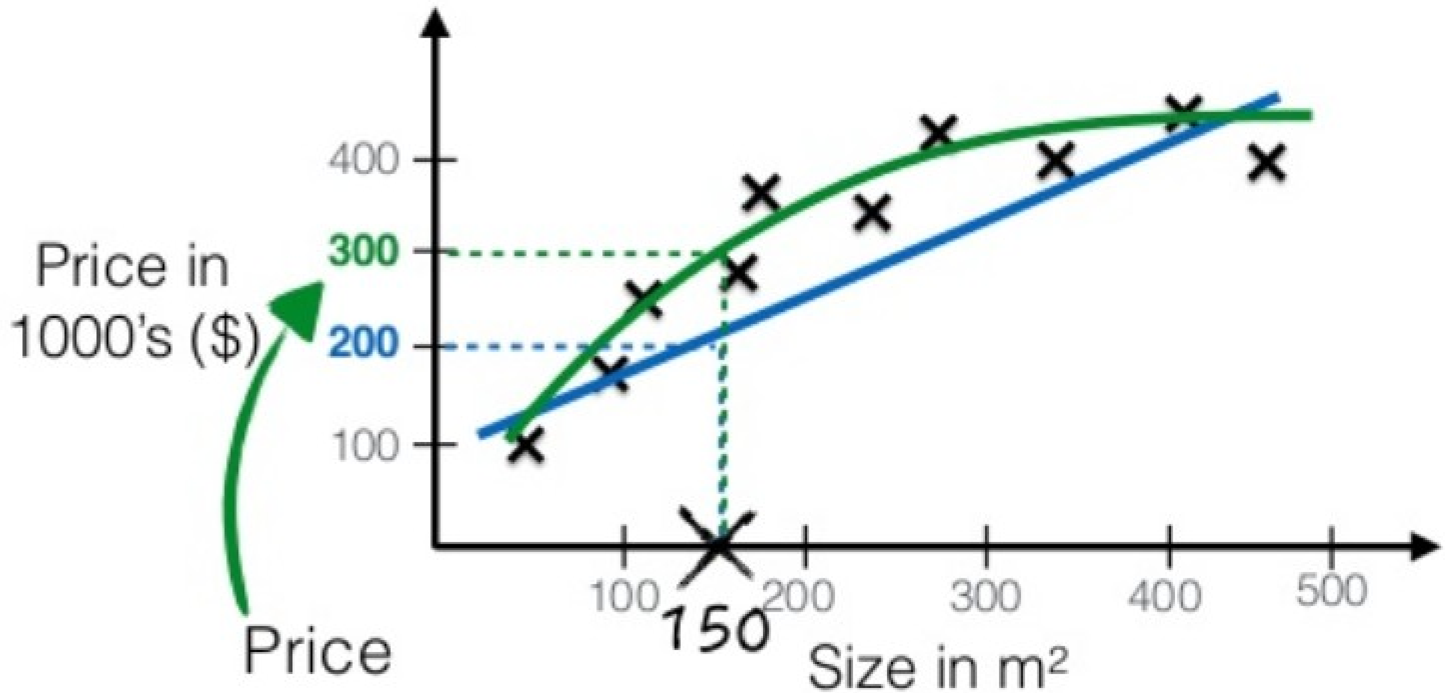
Beispiel Regression - Hauspreis



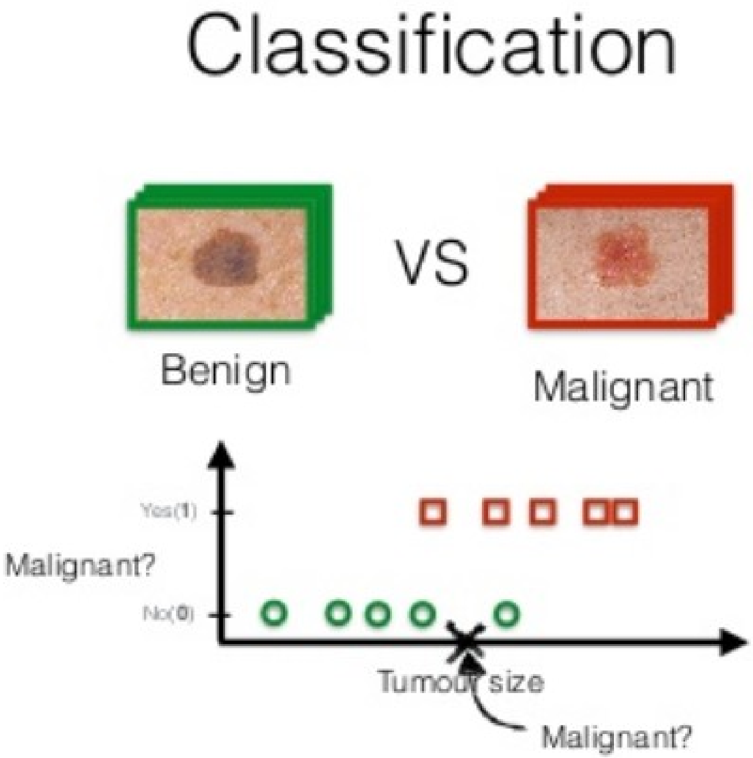
Beispiel Regression - Hauspreis



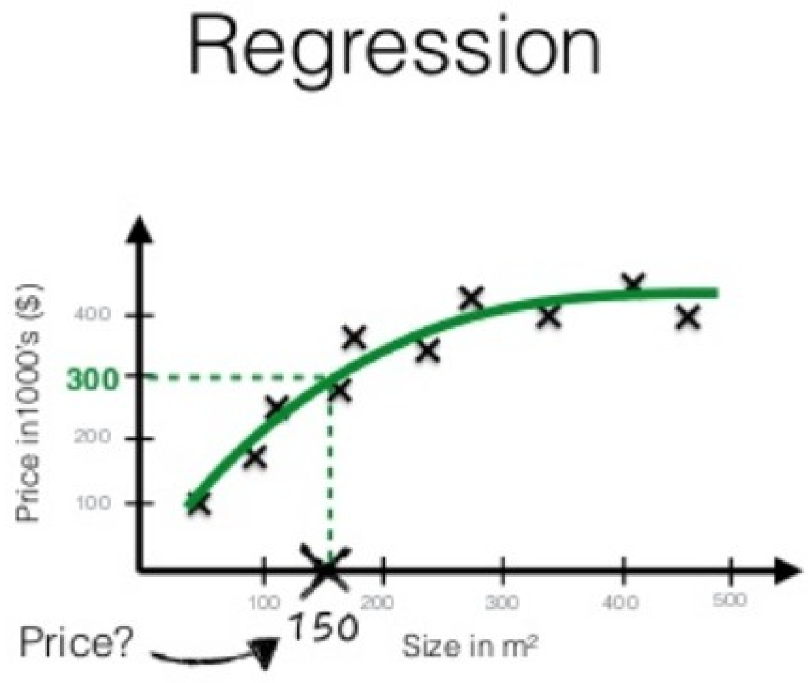
Beispiel Regression - Hauspreis



Classification vs Regression

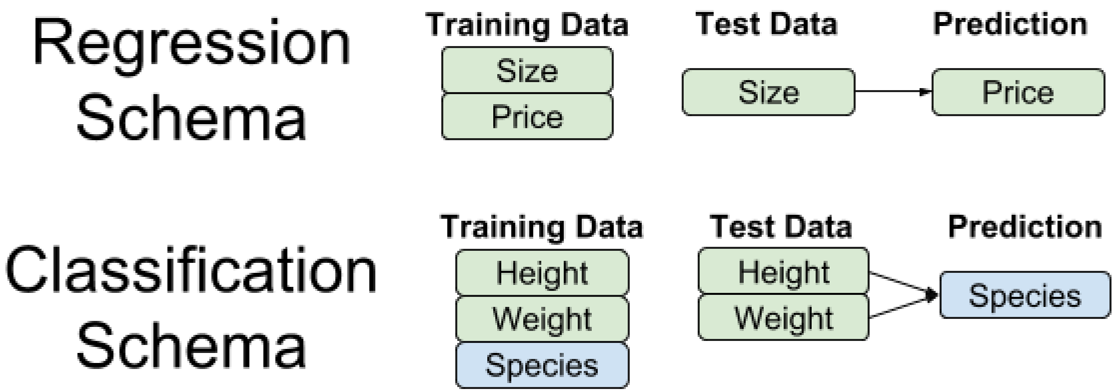


Die Ausgangsvariable nimmt Klassenbezeichnungen.



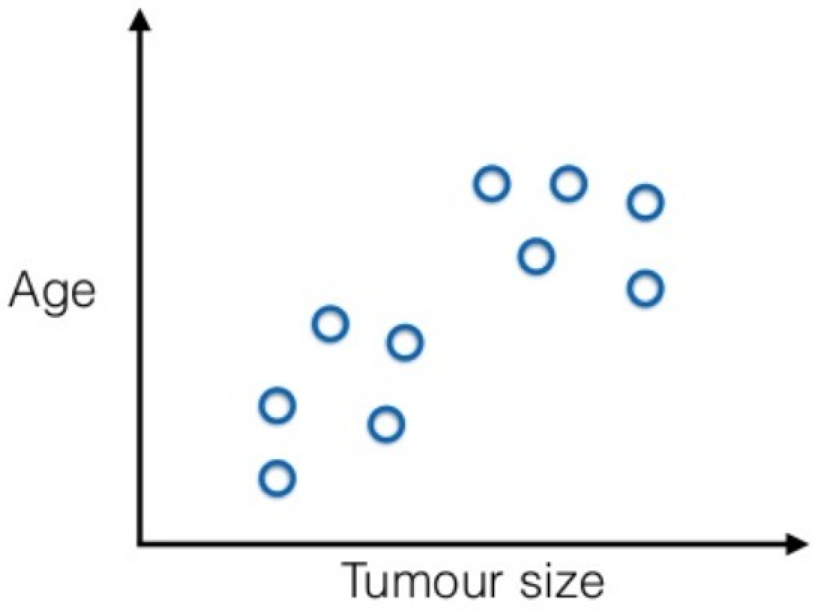
Die Ausgangsvariable nimmt kontinuierliche Werte an.

Classification vs Regression



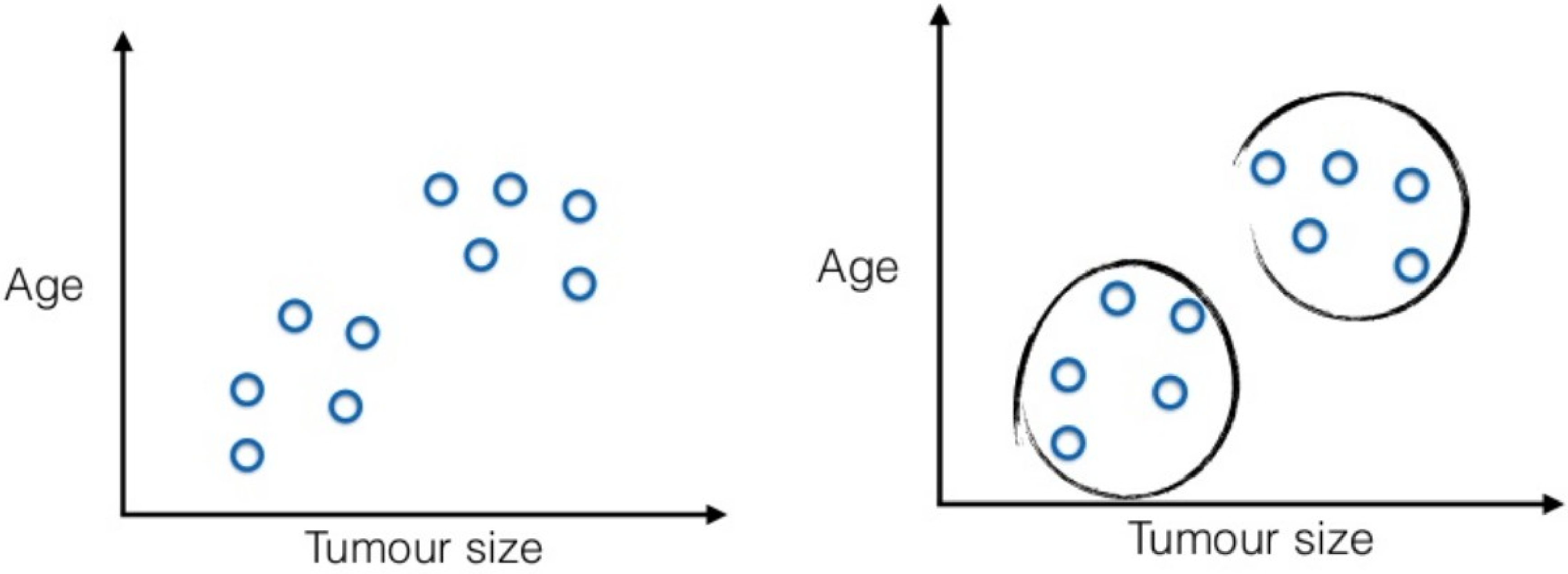
Unsupervised Learning

- Die bekannten Daten haben keine Labels (im Unterschied zum Supervised Learning)
- Wir wissen auch nicht, was die Datenpunkten bedeuten
- Die Aufgabe besteht darin, in den Daten Muster (Clusters) zu finden

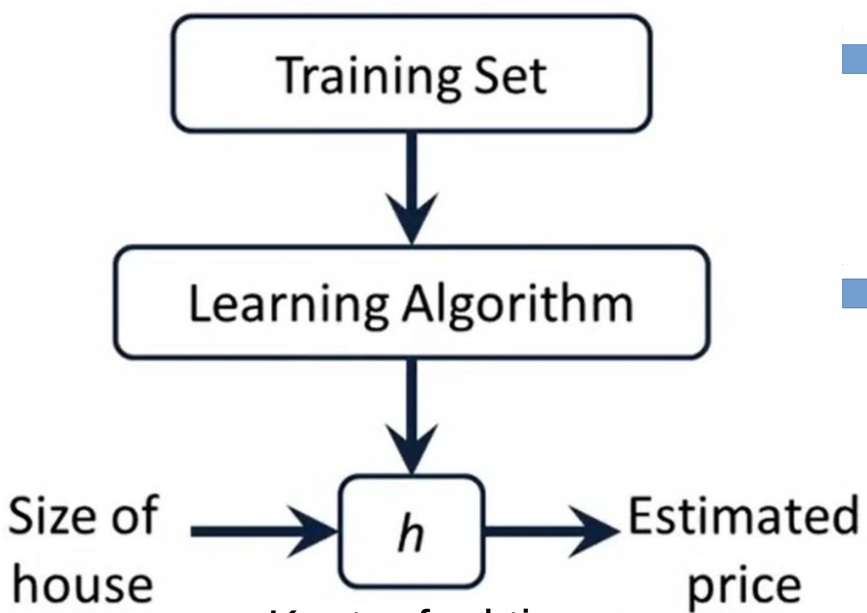


Unsupervised Learning

- Die bekannten Daten haben keine Labels (im Unterschied zum Supervised Learning)
- Wir wissen auch nicht, was die Datenpunkten bedeuten
- Die Aufgabe besteht darin, in den Daten Muster (Clusters) zu finden

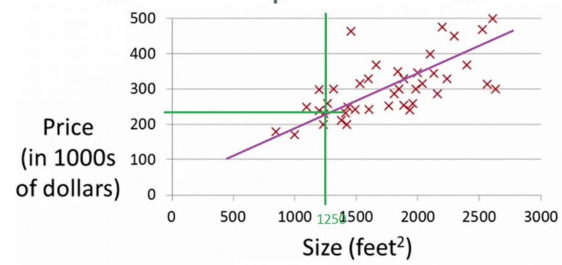


Linear Regression

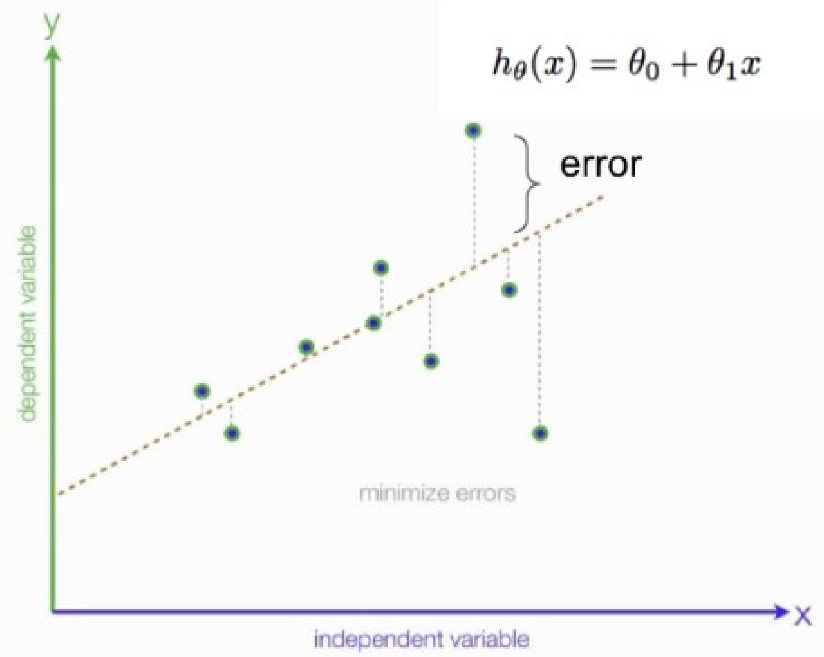


Kostenfunktion

Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
...	...



Kostenfunktion



Hypothesis:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

Parameters:

$$\theta_0, \theta_1$$

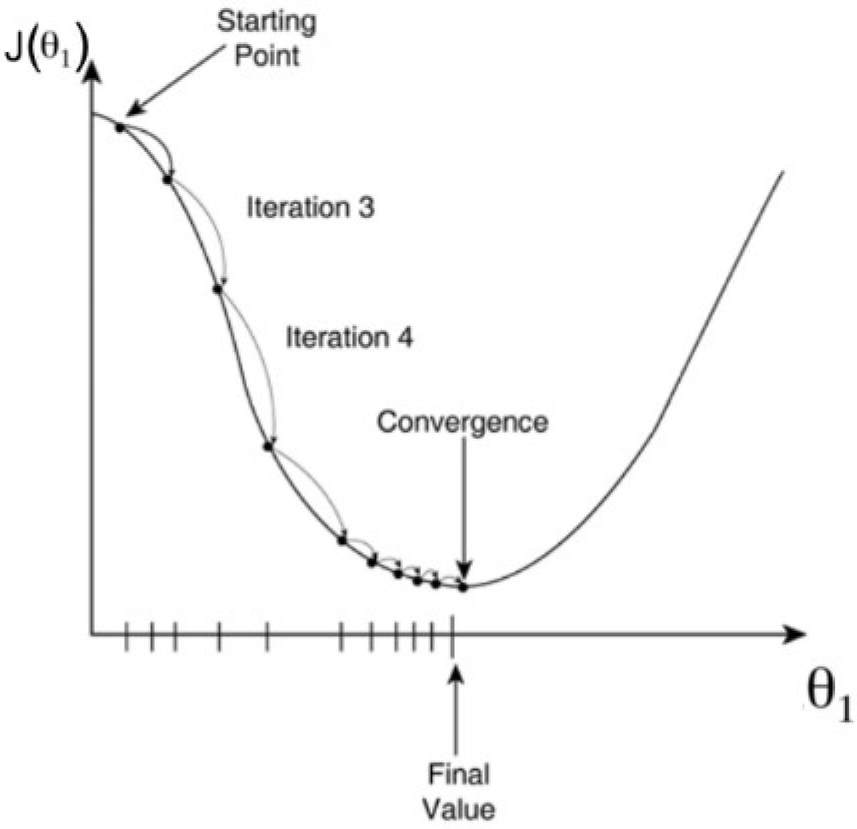
Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Goal:

$$\underset{\theta_0, \theta_1}{minimize} J(\theta_0, \theta_1)$$

Gradient Descent



Cost Function – "One Half Mean Squared Error":

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

Objective:

$$\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$$

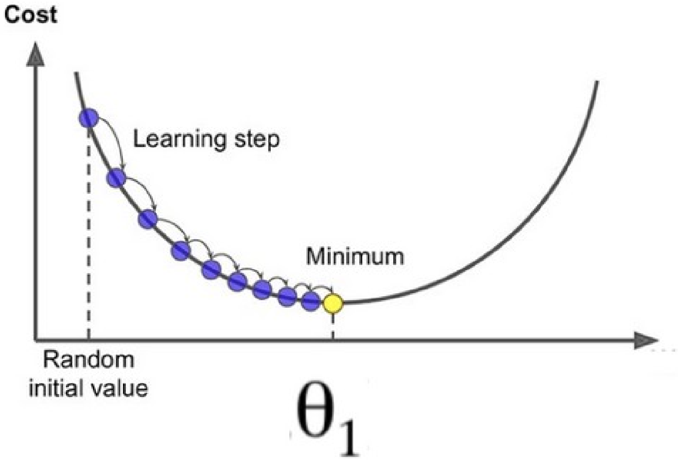
Derivatives:

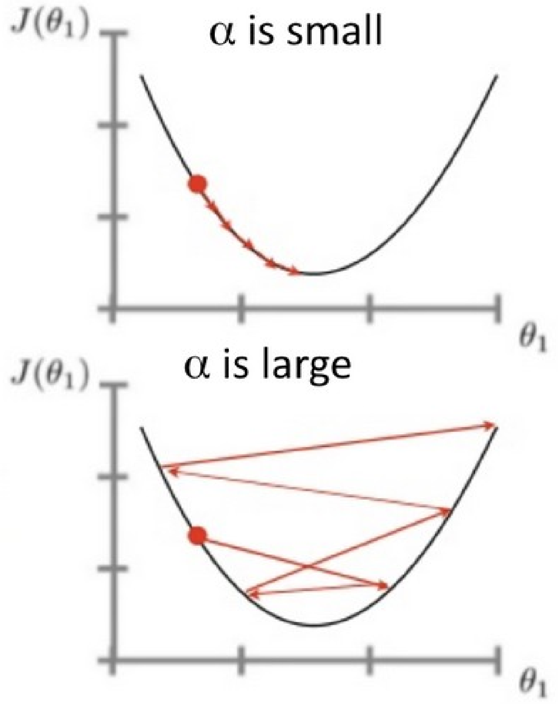
$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x^{(i)}$$

Learning Rate

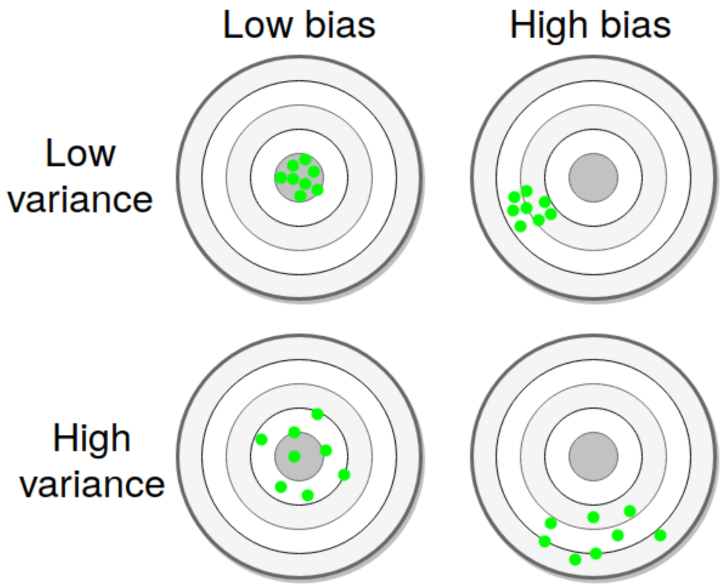
$$\text{repeat until convergence } \{$$
$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$
$$(\text{for } j = 1 \text{ and } j = 0)$$
$$\}$$



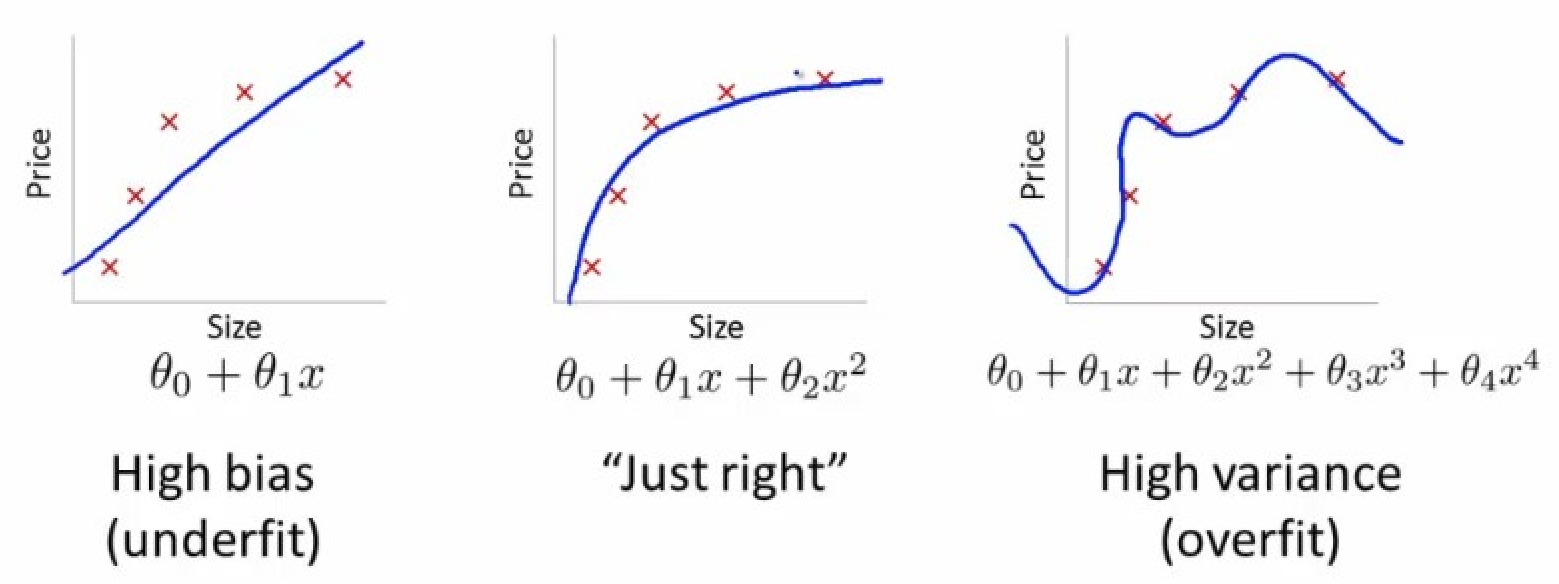


Bias-Variance Dilemma

- Bias: Differenz zwischen der erwarteten Vorhersage des Modells und dem wahren Wert.
- Variance: Wie stark die Vorhersagen für einen bestimmten Punkt zwischen verschiedenen Realisierungen des Modells variieren.



Bias-Variance Tradeoff



Supervised Learning Algorithmen

- Naive Bayes
- Decision Trees
- Random Forest
- Support Vector Machines (SVM)
- K-Nearest Neighbors
- Neural Networks

Naive Bayes

Die bedingte Wahrscheinlichkeit ist die Wahrscheinlichkeit des Auftretens eines Ereignisses unter der Bedingung, dass das Auftreten eines anderen Ereignisses bereits bekannt ist.

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

P (A|B) = bedingte Wahrscheinlichkeit von A, vorausgesetzt B.

P (A ∩ B) = gemeinsame Wahrscheinlichkeit für A und B (Schnittwahrscheinlichkeit)

P (B) = Wahrscheinlichkeit von B (hier Bedingung)

Satz von Bayes

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} = \frac{\frac{P(A \cap B)}{P(A)} \cdot P(A)}{P(B)} = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

Naive Bayes

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Problem: **Die Spieler spielen bei sonnigem Wetter.** Ist diese Aussage korrekt?

Naive Bayes

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

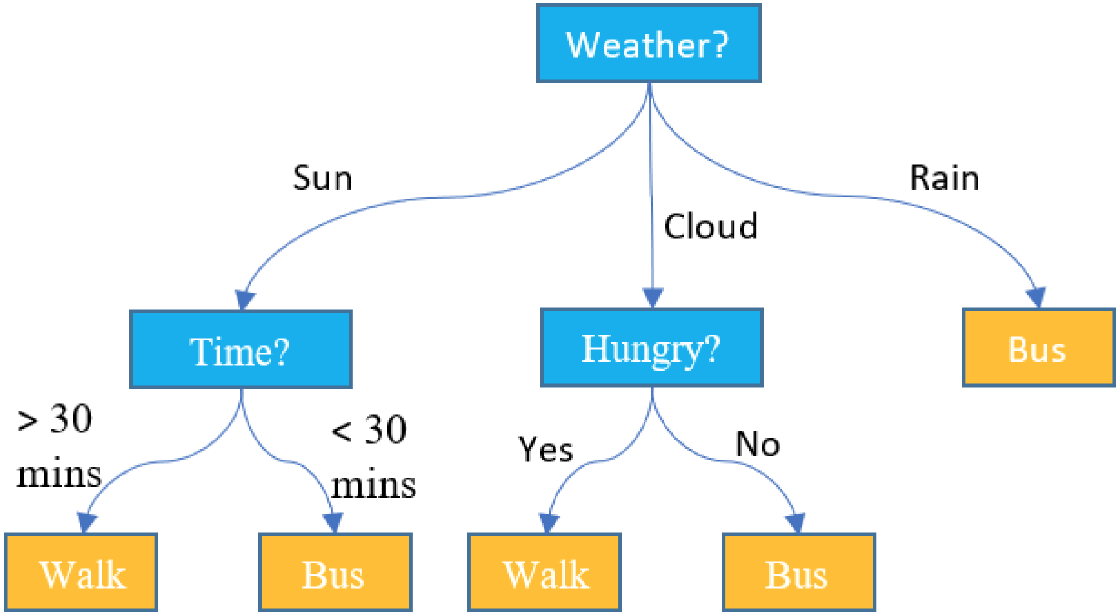
Problem: **Die Spieler spielen bei sonnigem Wetter.** Ist diese Aussage korrekt?

Mehrere Variabel

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Z.B. Workshop: Klassifizieren eine Frucht mit Attributes “rund, süss, rot” (anhand gegebene Beispiele)

Decision Trees

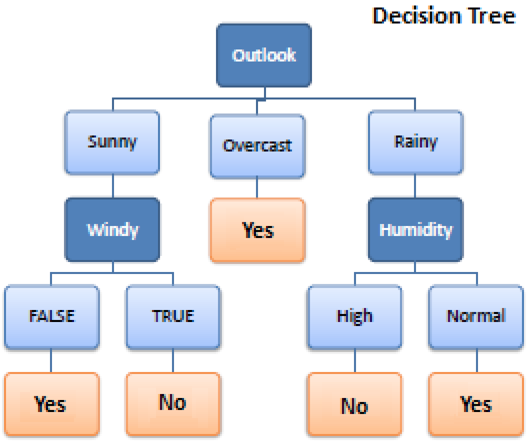


Verschiedene Algorithmen
ID3 → Entropy und Information Gain

Beispiel

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Predictors: Outlook, Temp., Humidity, Windy — Target: Play Golf



R_1: IF (Outlook=Sunny) AND (Windy=FALSE) THEN Play=Yes

R_2: IF (Outlook=Sunny) AND (Windy=TRUE) THEN Play=No

R_3: IF (Outlook=Overcast) THEN Play=Yes

R_4: IF (Outlook=Rainy) AND (Humidity=High) THEN Play=No

R_5: IF (Outlook=Rain) AND (Humidity=Normal) THEN Play=Yes

Entropy

Entropie ist der Grad der Zufälligkeit, oder es ist das Mass der Unreinheit.

$$H = -\sum p(x) \log p(x)$$

```
For example,
if we have items as number of dice face occurrence in a throw event as 1123,
the entropy is
   p(1) = 0.5
   p(2) = 0.25
   p(3) = 0.25
entropy = - (0.5 * log(0.5)) - (0.25 * log(0.25)) -(0.25 * log(0.25)
        = 0.45
```

Information Gain

- Der Informationsgewinn basiert auf der Abnahme der Entropie nach der Aufteilung eines Datensatzes auf ein Attribut.
- Information Gain Maximisieren

Entropie aus der der Häufigkeitstabelle eines Attributs

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

Entropie aus der der Häufigkeitstabelle zwei Attributs

$$E(T, X) = \sum_{c \in X} P(c) E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)
= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971
= 0.693

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)
= 0.940 – 0.693 = 0.247

Beispiel

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

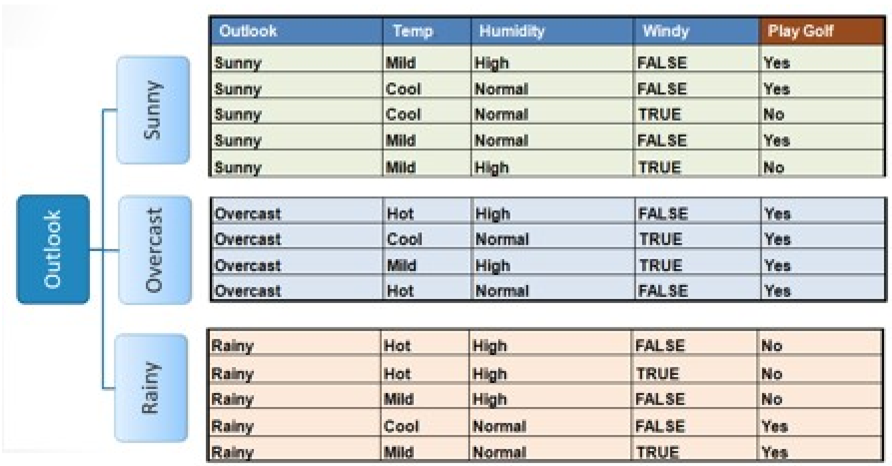
$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

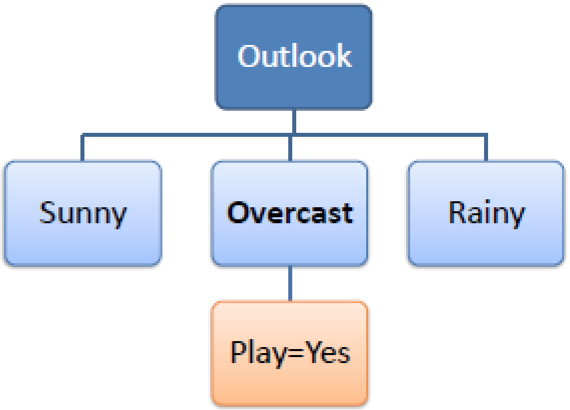
Maximum Information Gain

Beispiel



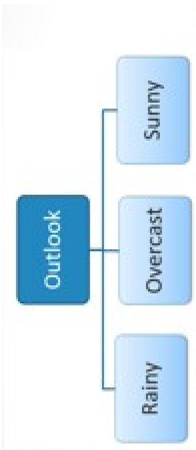
Outlook	Temp	Humidity	Windy	Play Golf
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

Temp.	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes



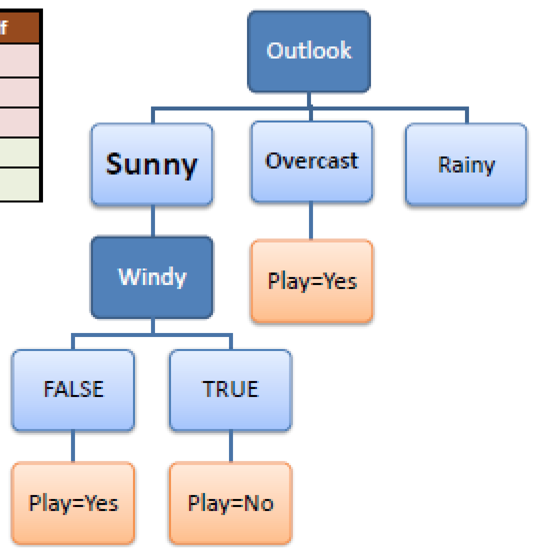
Entropy = 0 → Leaf

Beispiel



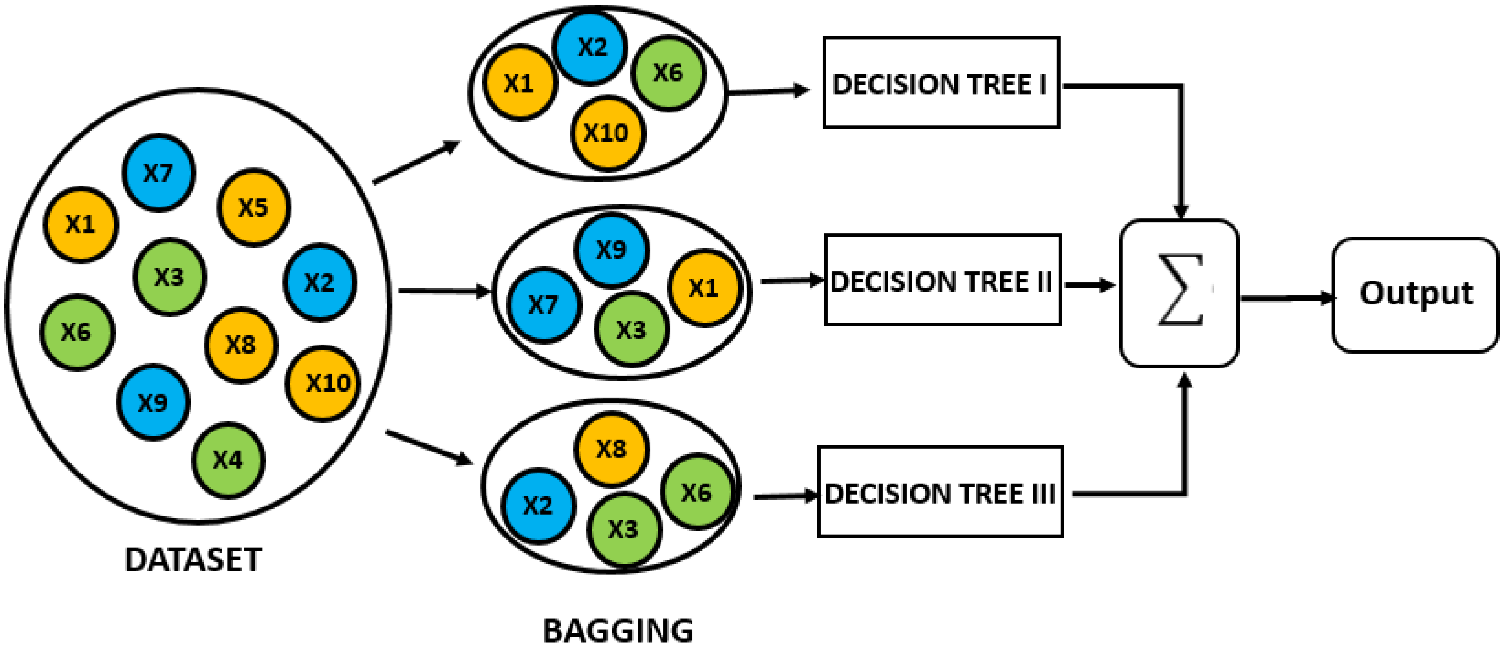
Outlook	Temp	Humidity	Windy	Play Golf
Sunny	Mild	High	FALSE	Yes
Sunny	Cool	Normal	FALSE	Yes
Sunny	Cool	Normal	TRUE	No
Sunny	Mild	Normal	FALSE	Yes
Sunny	Mild	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Overcast	Cool	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Hot	High	FALSE	No
Rainy	Hot	High	TRUE	No
Rainy	Mild	High	FALSE	No
Rainy	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	TRUE	Yes

Temp	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



Entropy not 0 → more splits

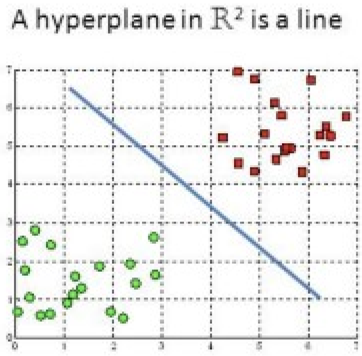
Random Forest

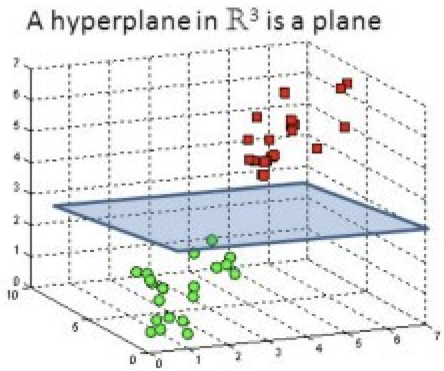


Support Vector Machines (SVM)

- Das Ziel des SVM ist es, eine Hyperebene in einem N-dimensionalen Raum (N = Anzahl Features) zu finden, die die Datenpunkte eindeutig klassifiziert

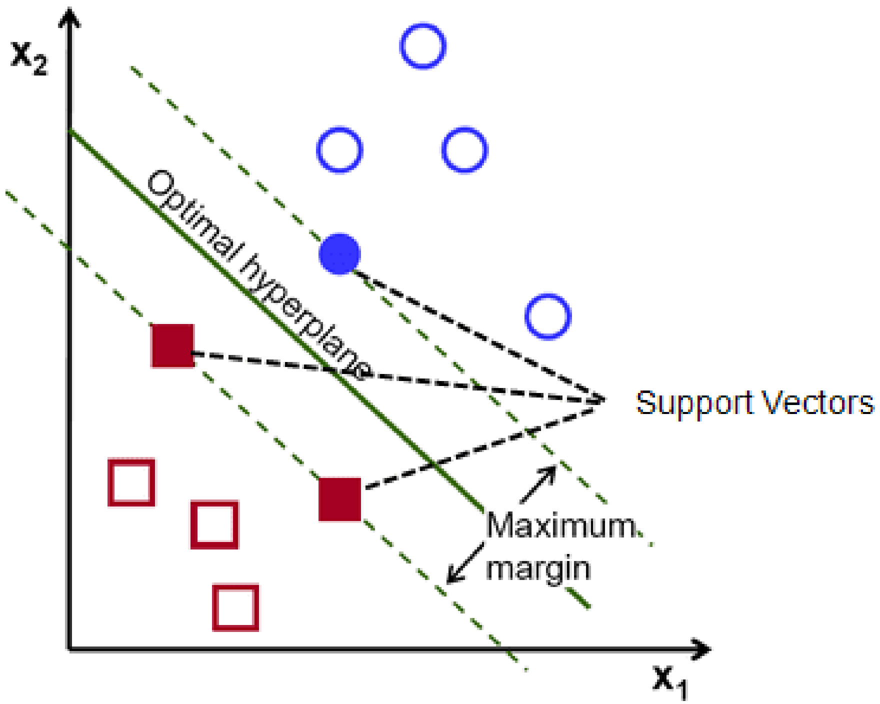
- In zwei Dimensionen ist diese Hyperebene eine Linie.
-
- In drei Dimensionen ist diese Hyperebene eine Ebene.



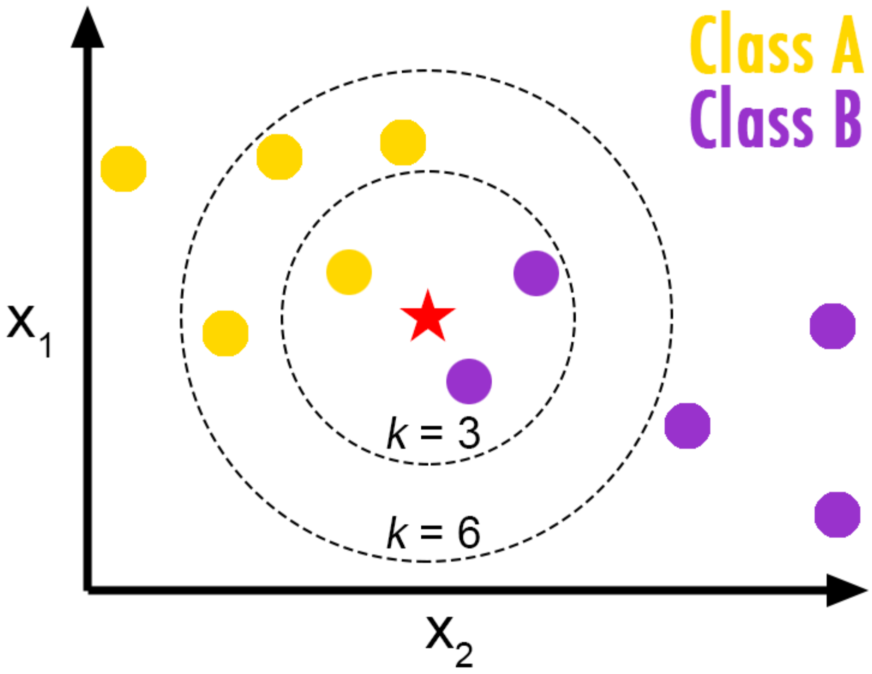


Support Vector Machines (SVM)

- Support Vectors sind Datenpunkte, die näher an der Hyperebene liegen und die Position und Orientierung der Hyperebene beeinflussen. Mit diesen Vektoren maximieren wir den Rand des Klassifikators



K Nearest Neighbors (KNN)



Euclidean

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Beispiel: Kredit geben?

Customer	Age	Loan	Default
John	25	40000	N
Smith	35	60000	N
Alex	45	80000	N
Jade	20	20000	N
Kate	35	120000	N
Mark	52	18000	N
Anil	23	95000	Y
Pat	40	62000	Y
George	60	100000	Y
Jim	48	220000	Y
Jack	33	150000	Y
Andrew	48	142000	?

Euclidean

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

We need to predict Andrew default status by using Euclidean distance

Customer	Age	Loan	Default	Euclidean distance
John	25	40000	N	1,02,000.00
Smith	35	60000	N	82,000.00
Alex	45	80000	N	62,000.00
Jade	20	20000	N	1,22,000.00
Kate	35	120000	N	22,000.00
Mark	52	18000	N	1,24,000.00
Anil	23	95000	Y	47,000.01
Pat	40	62000	Y	80,000.00
George	60	100000	Y	42,000.00
Jim	48	220000	Y	78,000.00
Jack	33	150000	Y	8,000.01
Andrew	**48**	**142000**	**?**	

First Step calculate the Euclidean distance dist(d) = Sq.rt $(x_1-y_1)^2 + (x_2-y_2)^2$
= Sq.rt$(48-25)^2 + (142000 - 40000)^2$
dist (d_1) = 1,02,000.

We need to calcuate the distance for all the datapoints

Customer	Age	Loan	Default	Euclidean distance	Minimum Euclidean Distance
John	25	40000	N	1,02,000.00	
Smith	35	60000	N	82,000.00	
Alex	**45**	**80000**	**N**	**62,000.00**	**5**
Jade	20	20000	N	1,22,000.00	
Kate	**35**	**120000**	**N**	**22,000.00**	**2**
Mark	52	18000	N	1,24,000.00	
Anil	**23**	**95000**	**Y**	**47,000.01**	**4**
Pat	40	62000	Y	80,000.00	
George	**60**	**100000**	**Y**	**42,000.00**	**3**
Jim	48	220000	Y	78,000.00	
Jack	**33**	**150000**	**Y**	**8,000.01**	**1**
Andrew	48	142000	?		

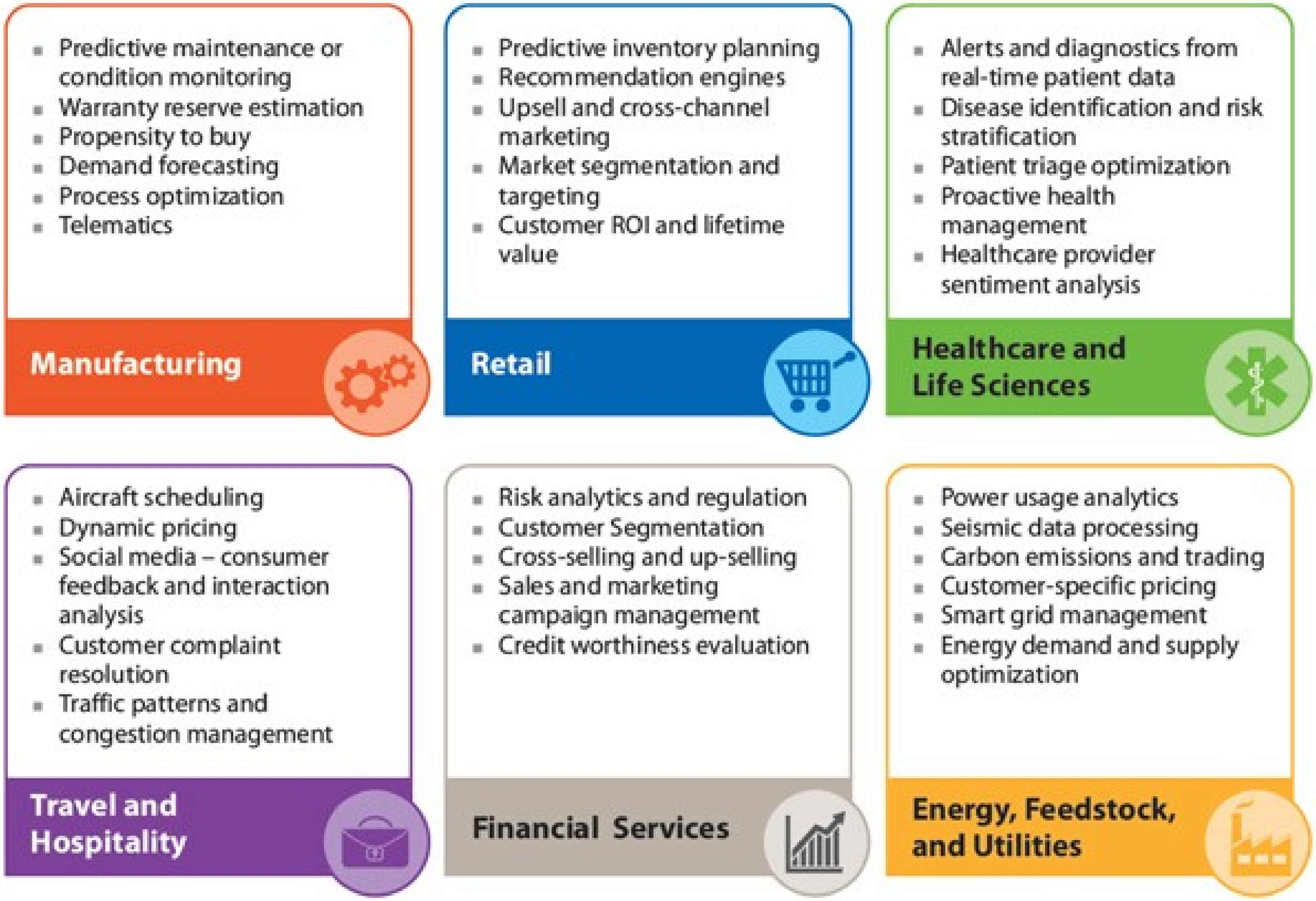
Let assume K = 5

Find minimum euclidean distance and rank in order (ascending)

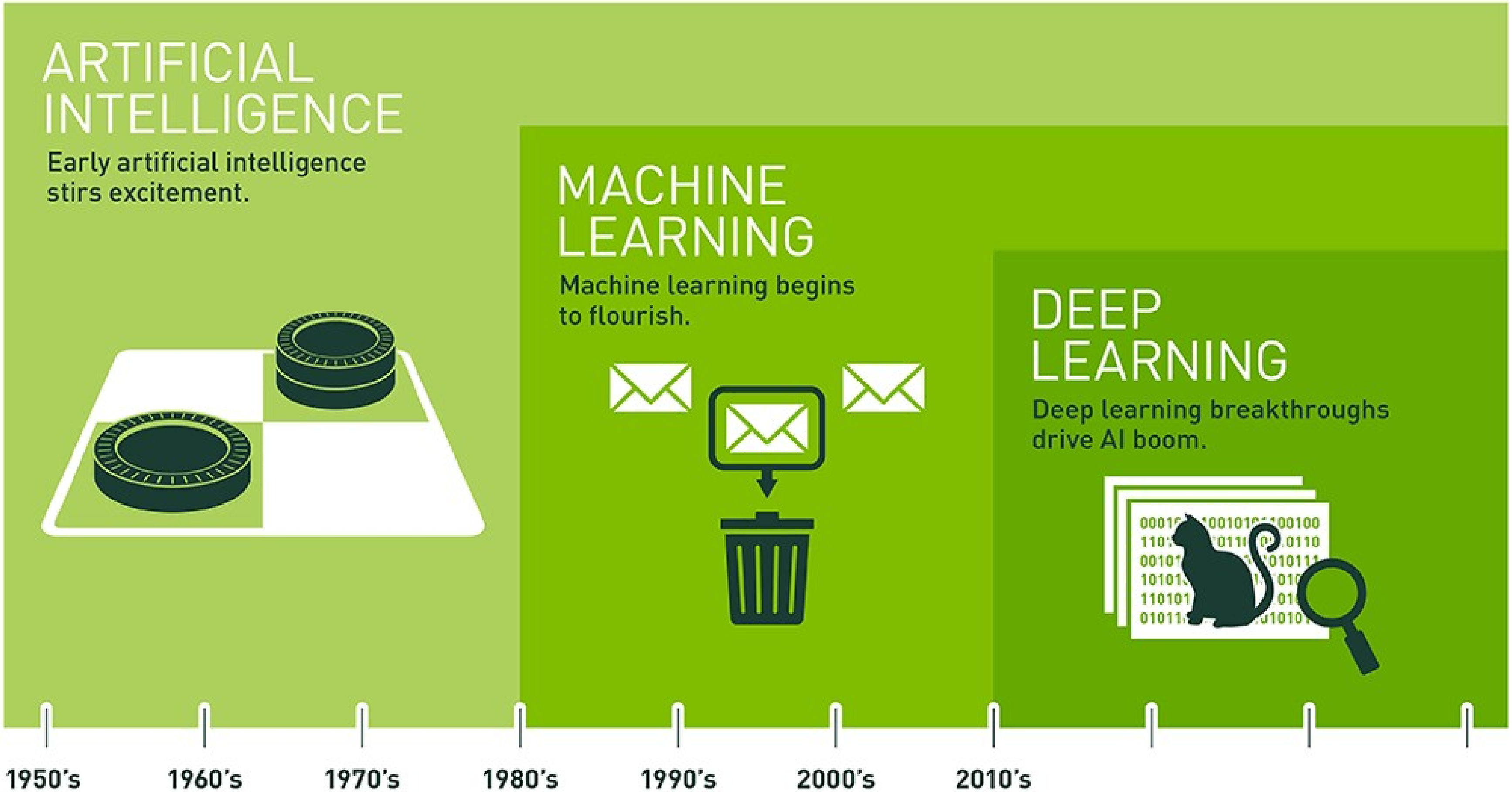
In this case, 5 minimum euclidean distance. With k=5, there are two Default = N and three Default = Y out of five closest neighbors.

We can say Andrew default stauts is 'Y' (Yes)

Machine Learning Anwendungen in der Industrie

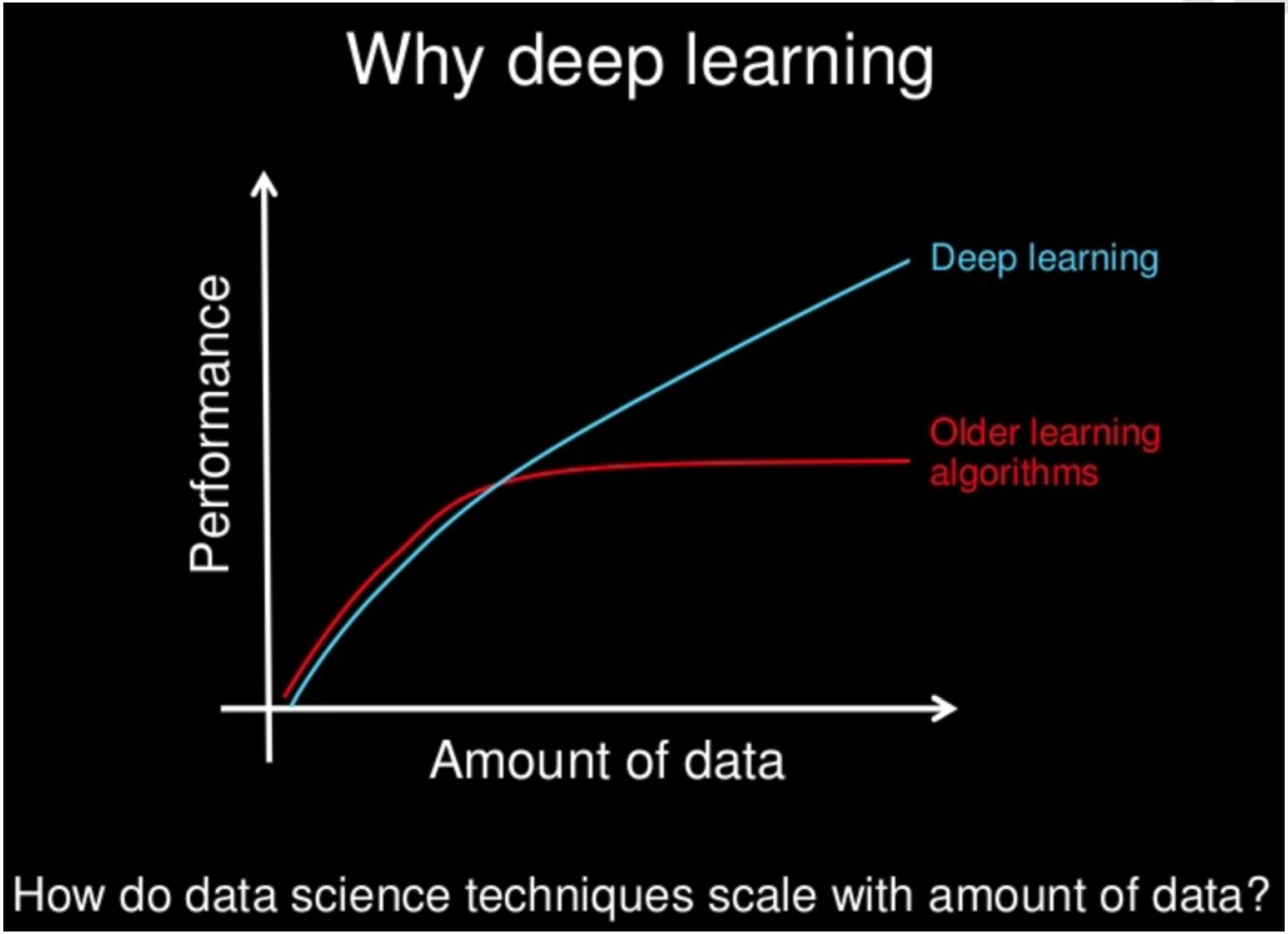


Deep Learning

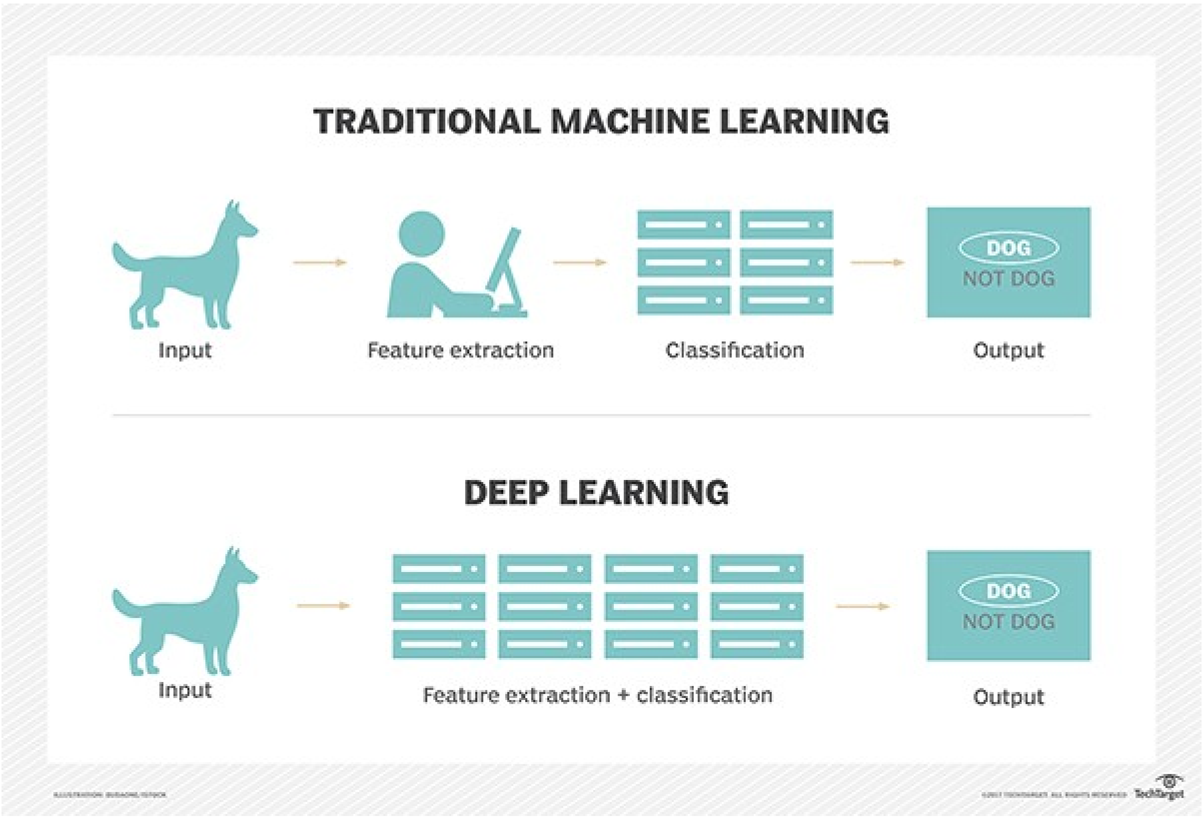


Quelle: NVidia

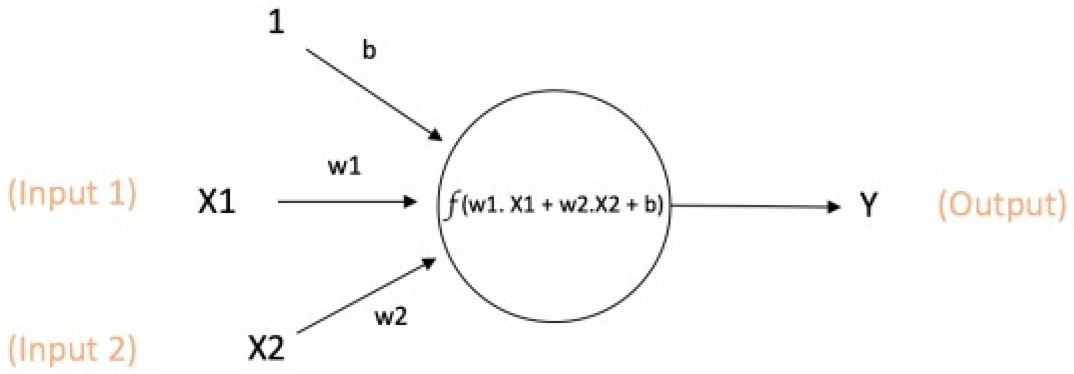
Warum Deep Learning?



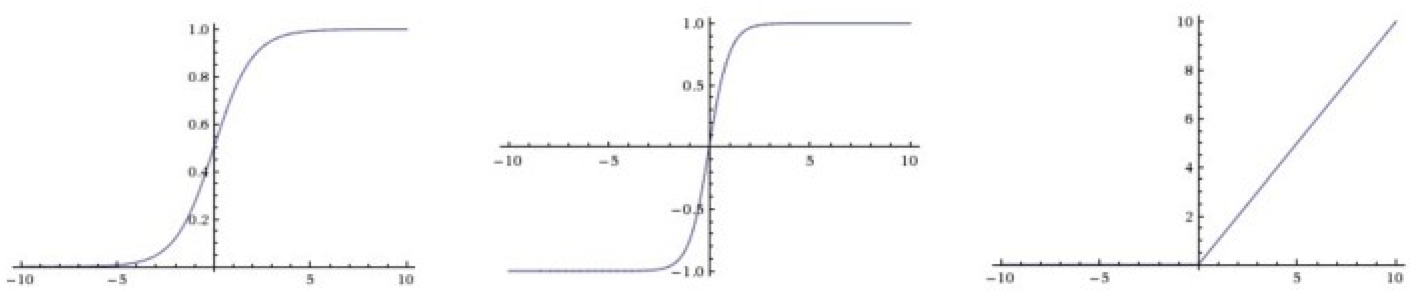
Warum Deep Learning?



Neural Networks

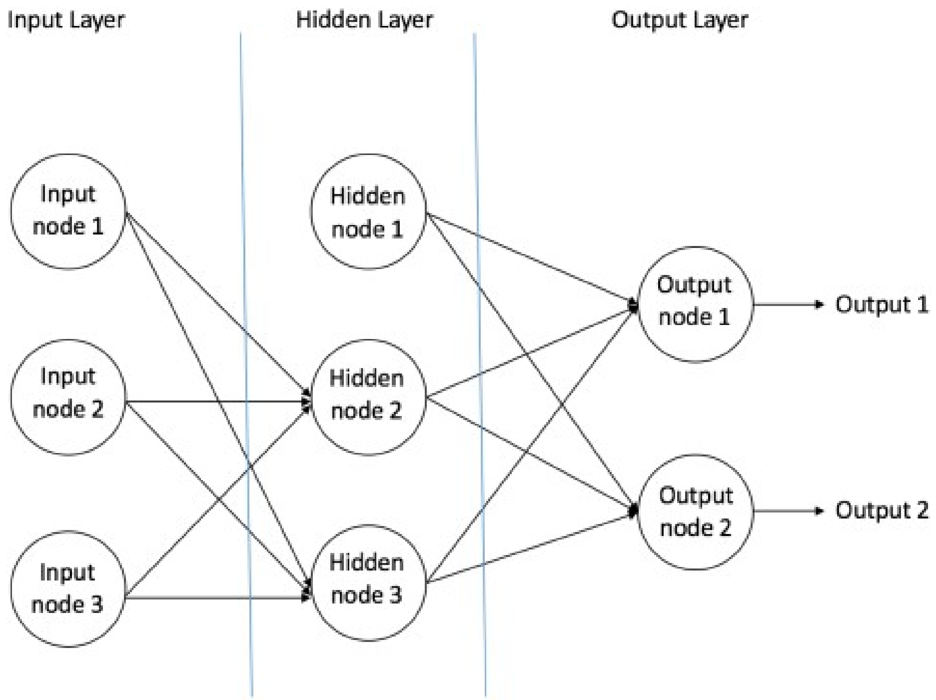


Output of neuron = Y= f(w1. X1 + w2.X2 + b)



Sigmoid tanh ReLU

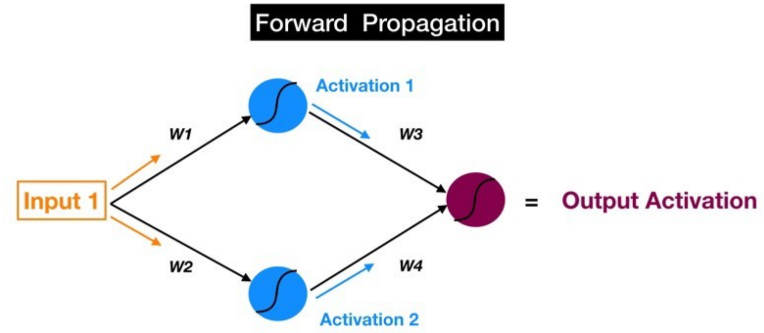
Feed Forward Neural Network



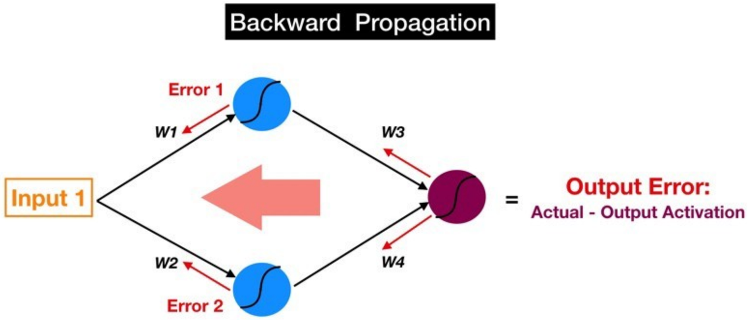
Single Layer Perceptron → No Hidden Layer

Multi Layer Perceptron → Ein oder mehrere Hidden Layers

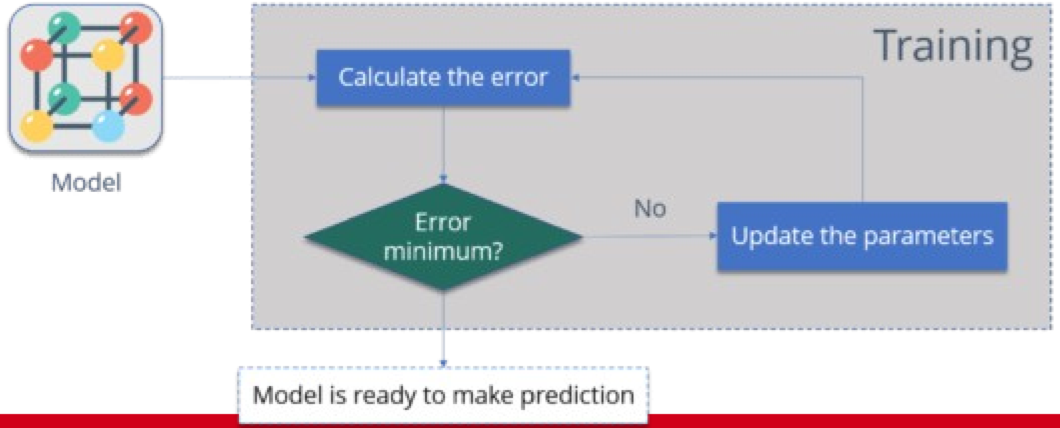
Backpropagation



Move Signal forward

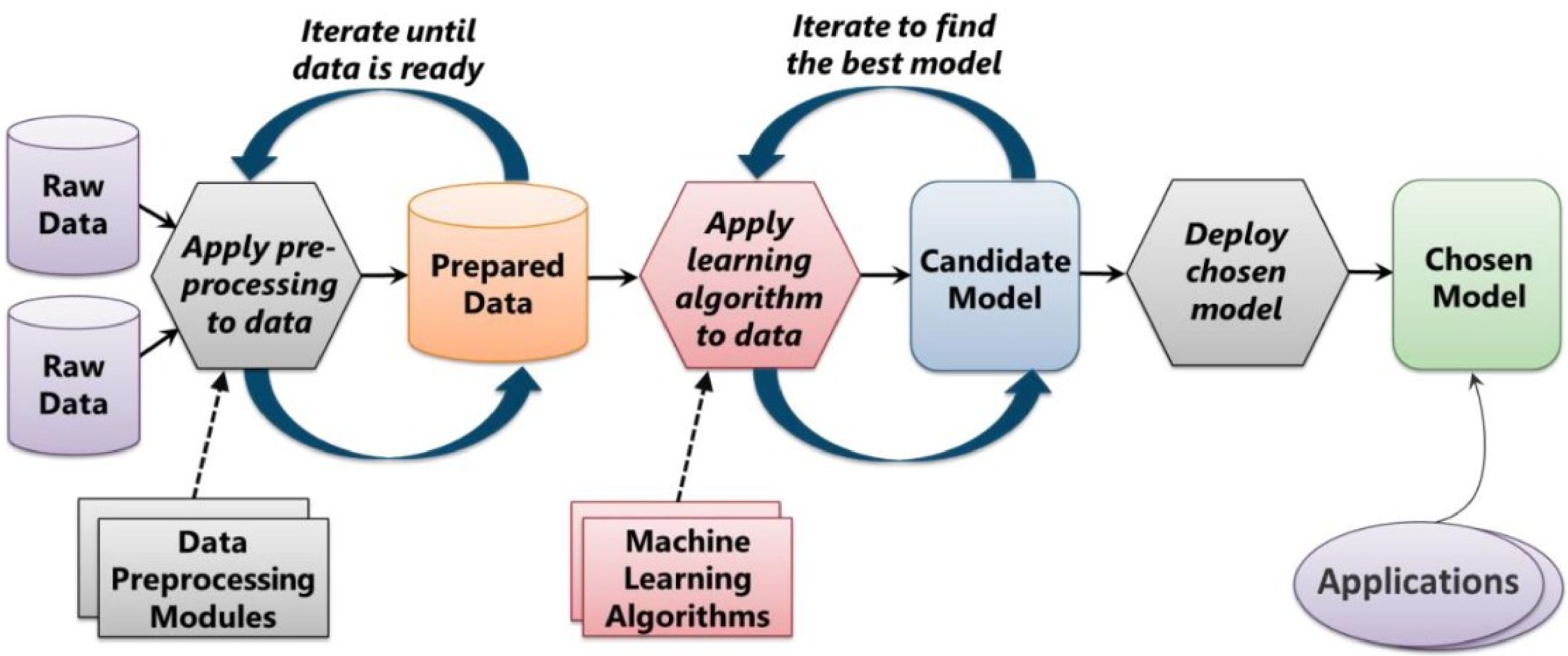


Move Error Backwards



Vorgehen im Machine Learning

Das generelle Machine Learning vorgehen



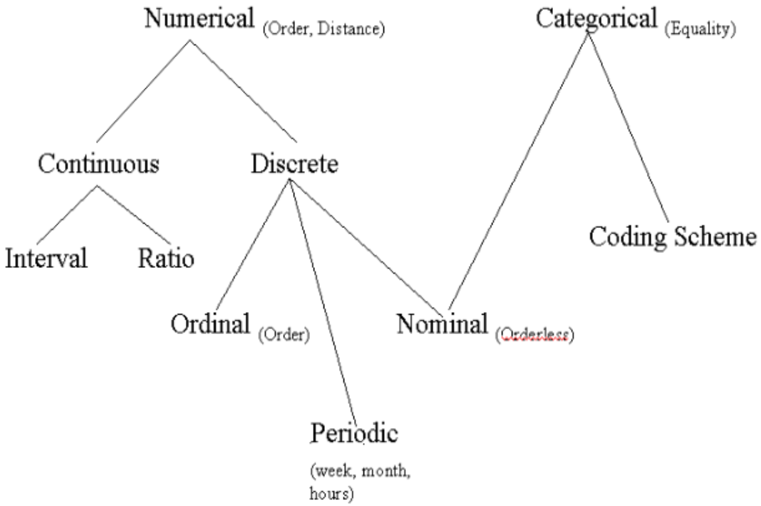
- Quelle: [Chappell: Introduction to Machine Learning, 2015"]() p.5

Data Preprocessing

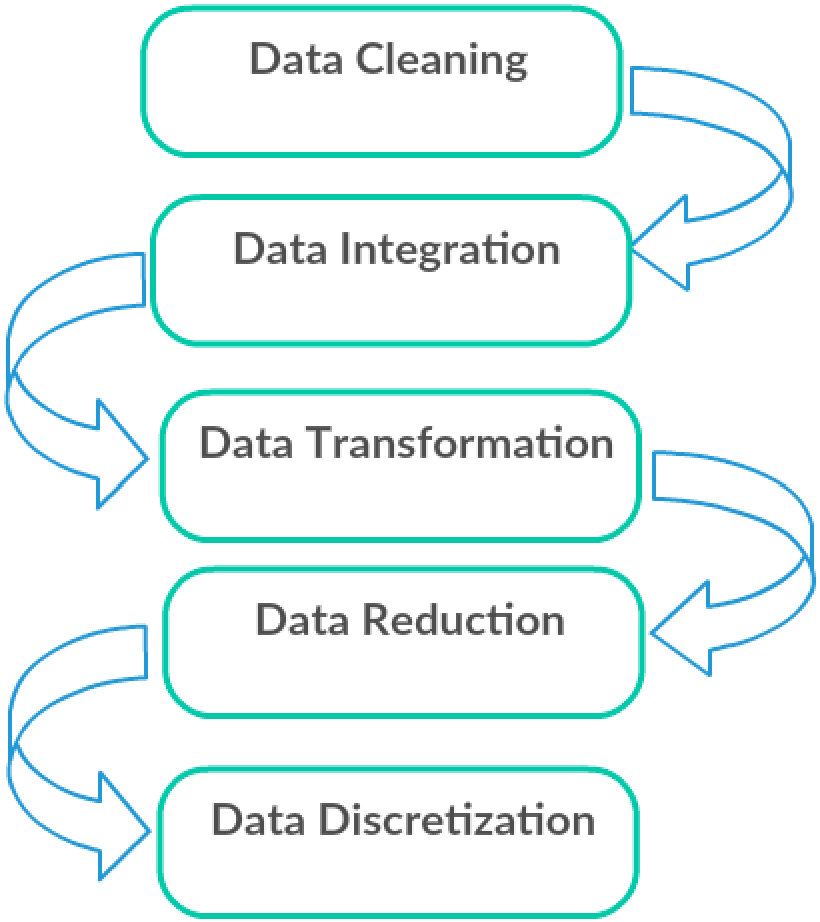
A1	A2	...	An	C

- Data types
  - Numerical, categorical
  - Static, dynamic (temporal)
  - Discrete Attributes (counts, Wörter, Postleitzahl)
  - Continuous Attributes (Temperatur, Gewicht)

- Andere
  - Distributed Data
  - Text
  - Web
  - Metadata
  - Images
  - Videos
  - Audio

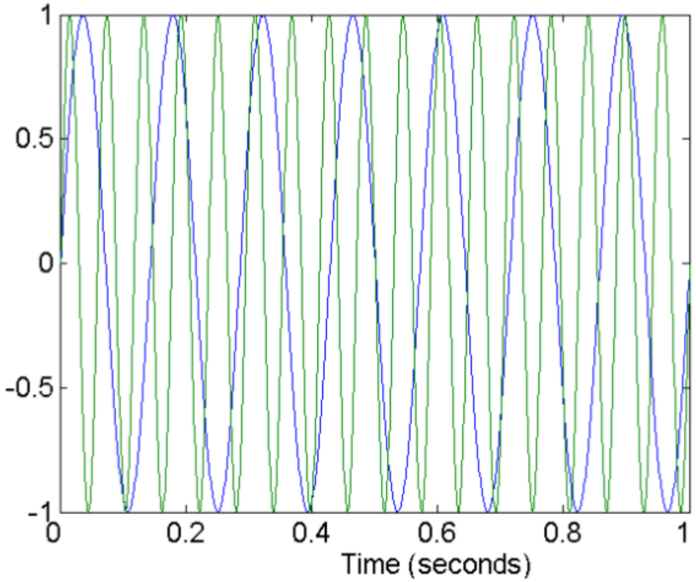


Data Preprocessing Steps

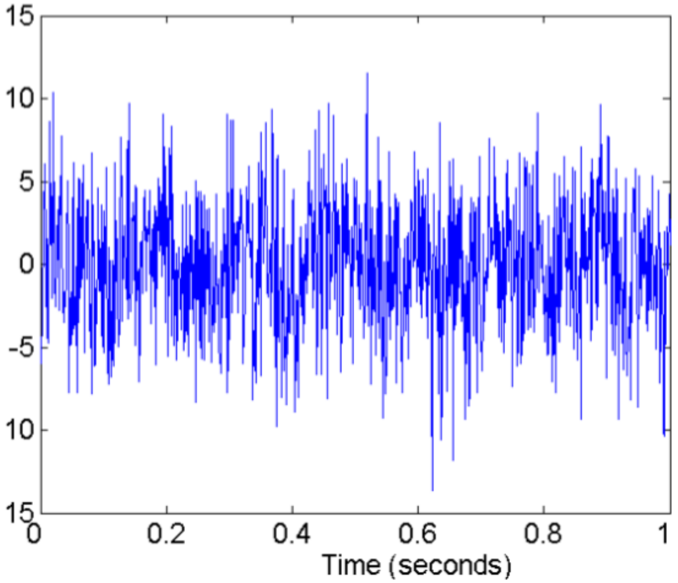


Data Preprocessing Beispiel - Noise

Verzerrung (Distorsion) der Stimme einer Person, wenn sie mit einem schlechten Telefon redet



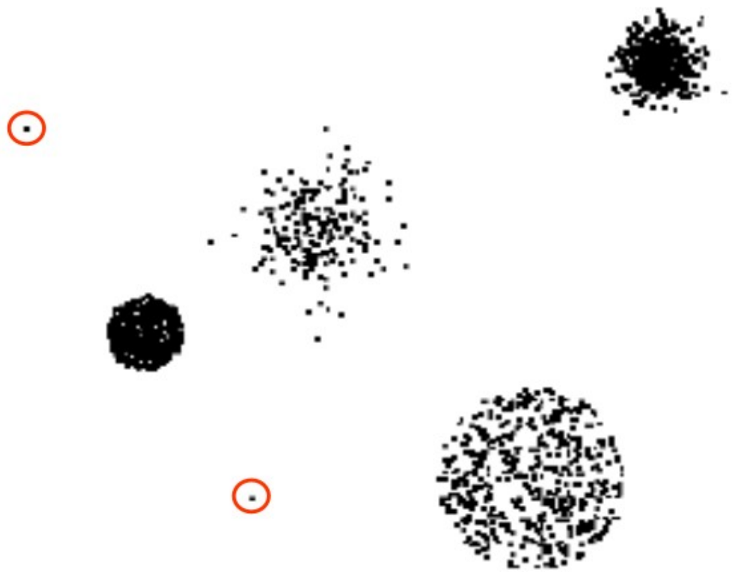
Two Sine Waves



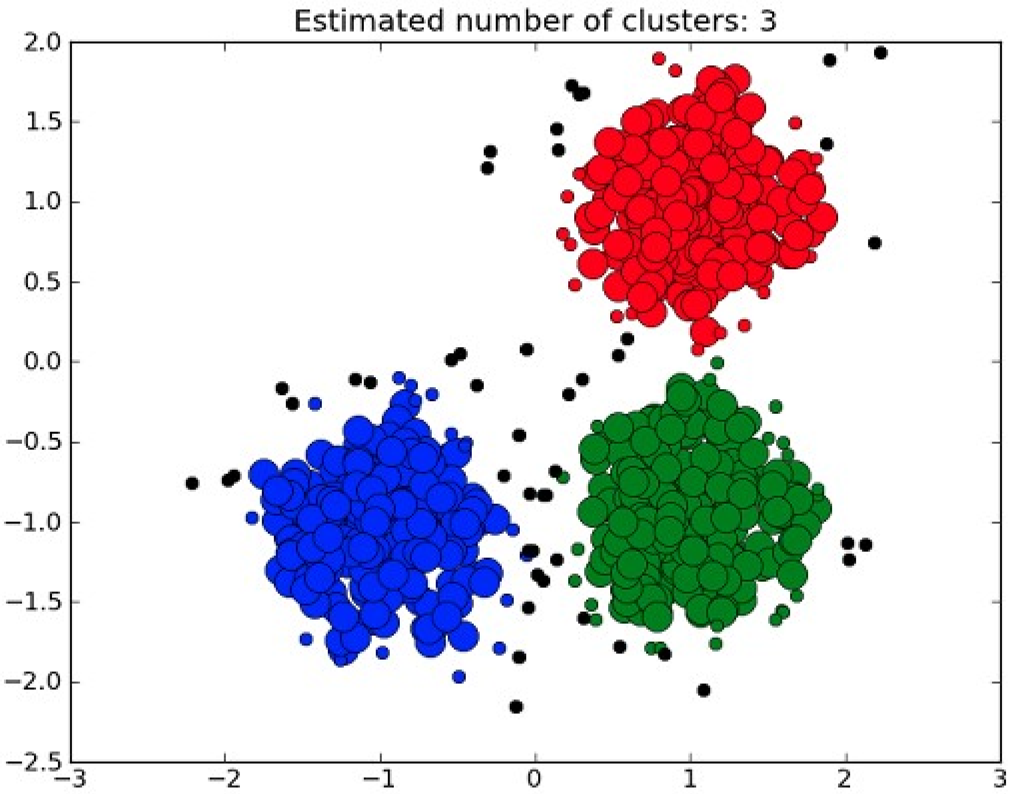
Two Sine Waves + Noise

Data Preprocessing Beispiel - Outliers

Outliers sind Datenobjekte die wesentlich anders als die meisten anderen Datenobjekte sind

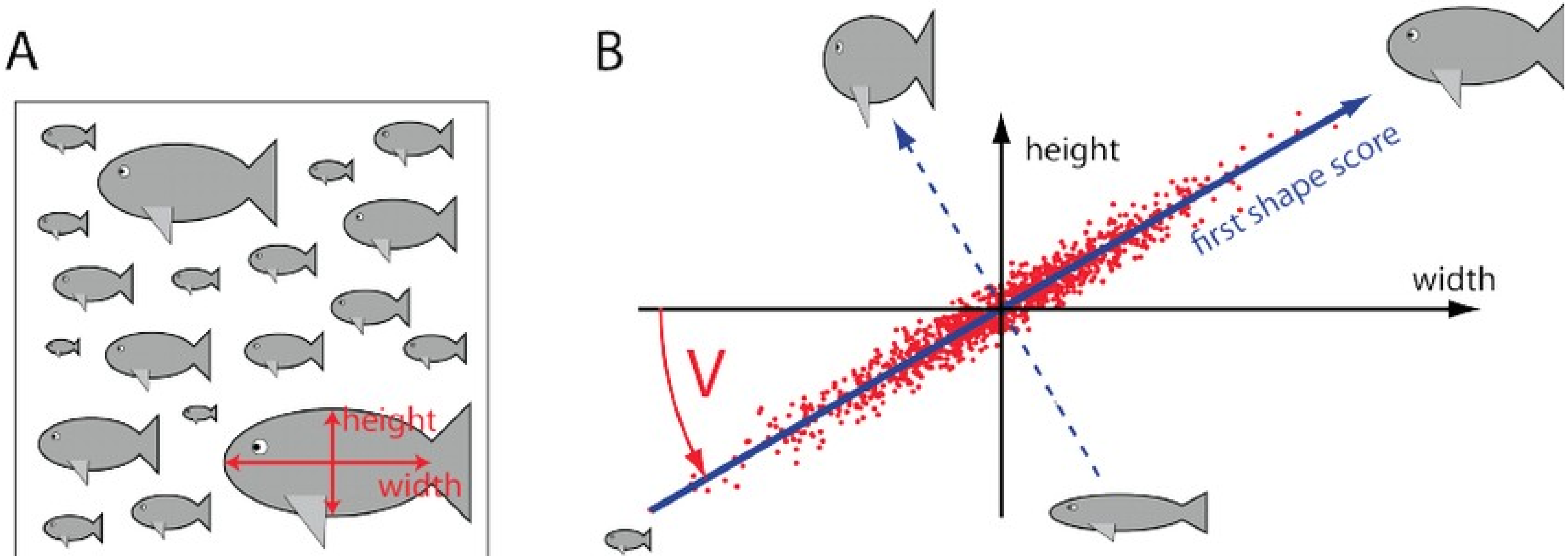


Data Preprocessing Beispiel - Clustering

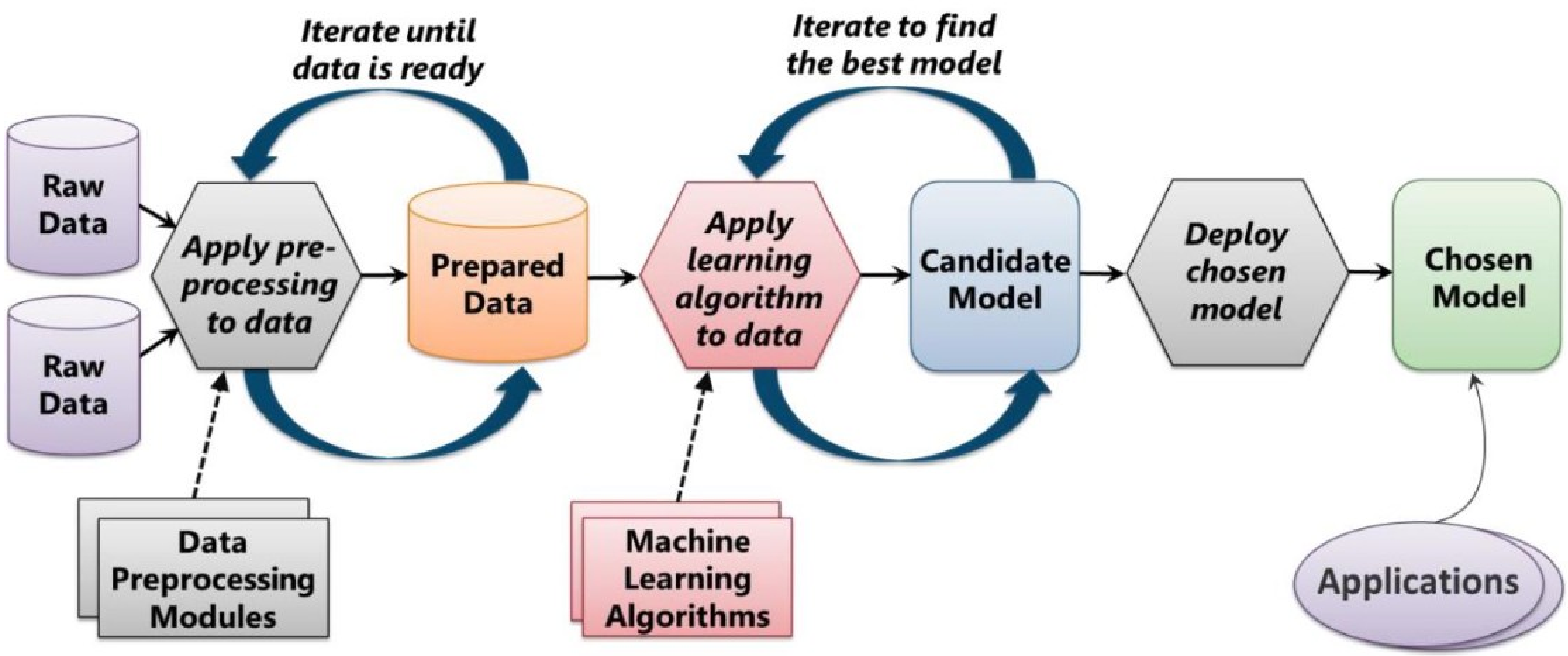


Data Preprocessing Beispiel - PCA

Ziel ist es, eine Projektion zu finden, die die grösste Variationsbreite der Daten enthält

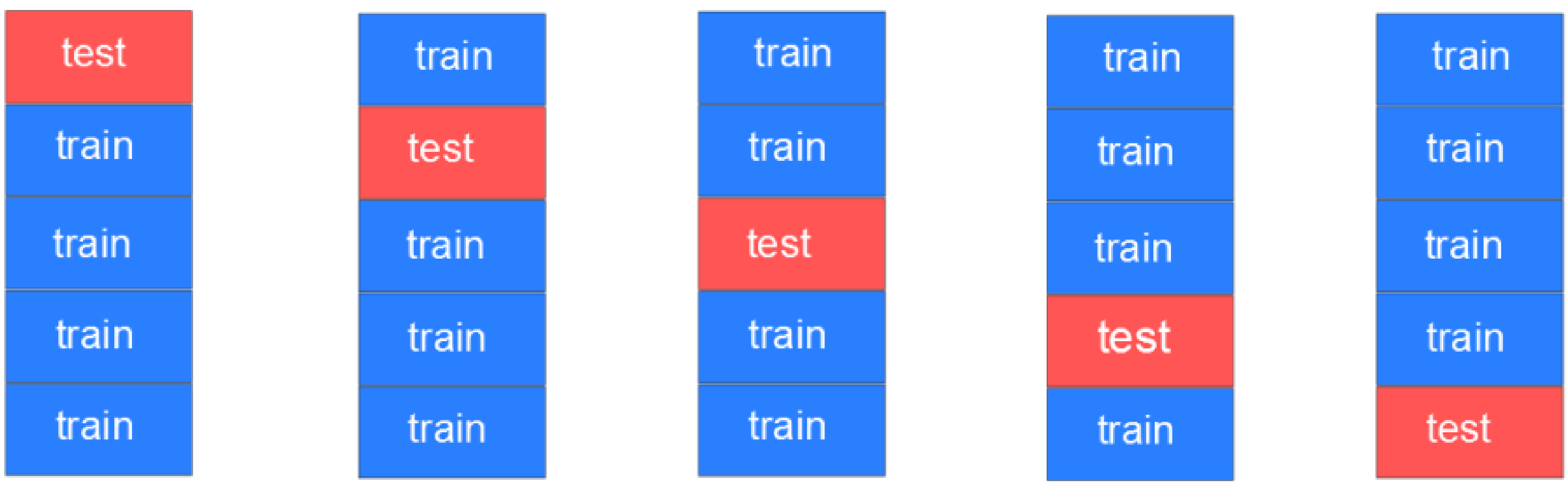


Das generelle Machine Learning vorgehen

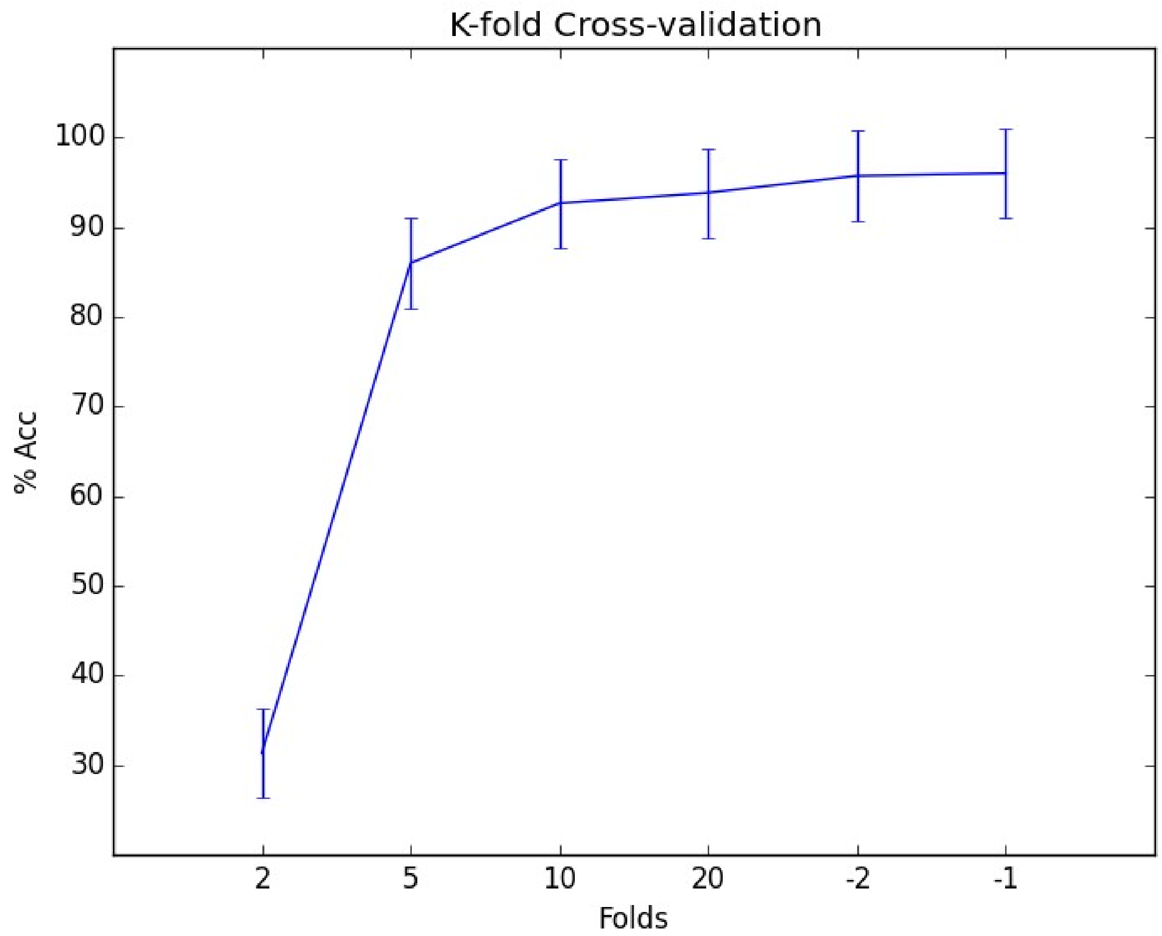


- Quelle: Chappell: Introduction to Machine Learning, 2015" p.5

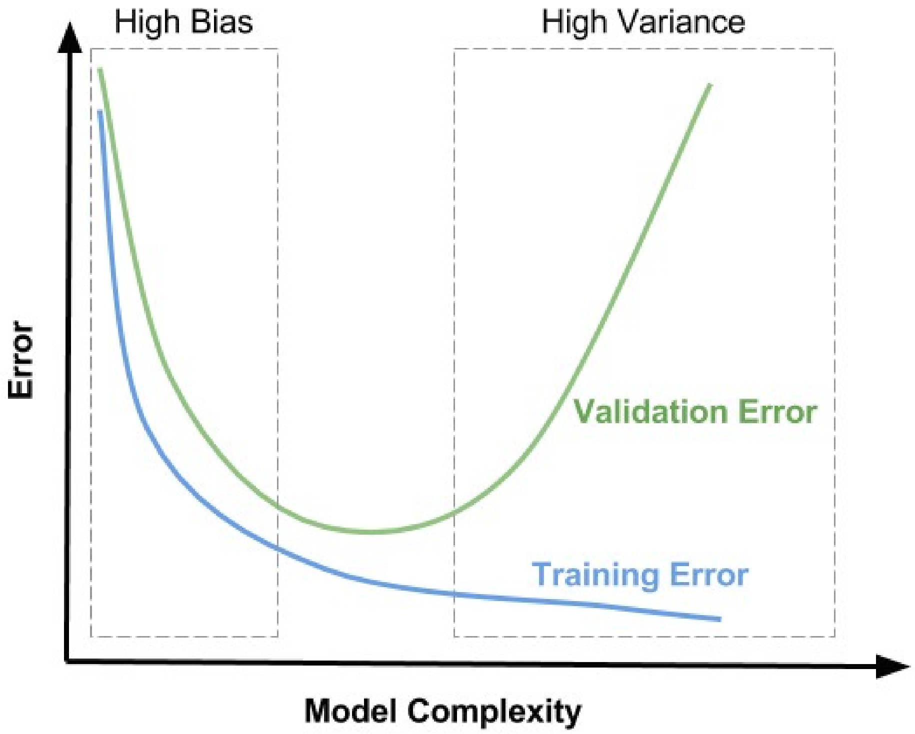
Kreuzvalidierung



Kreuzvalidierung



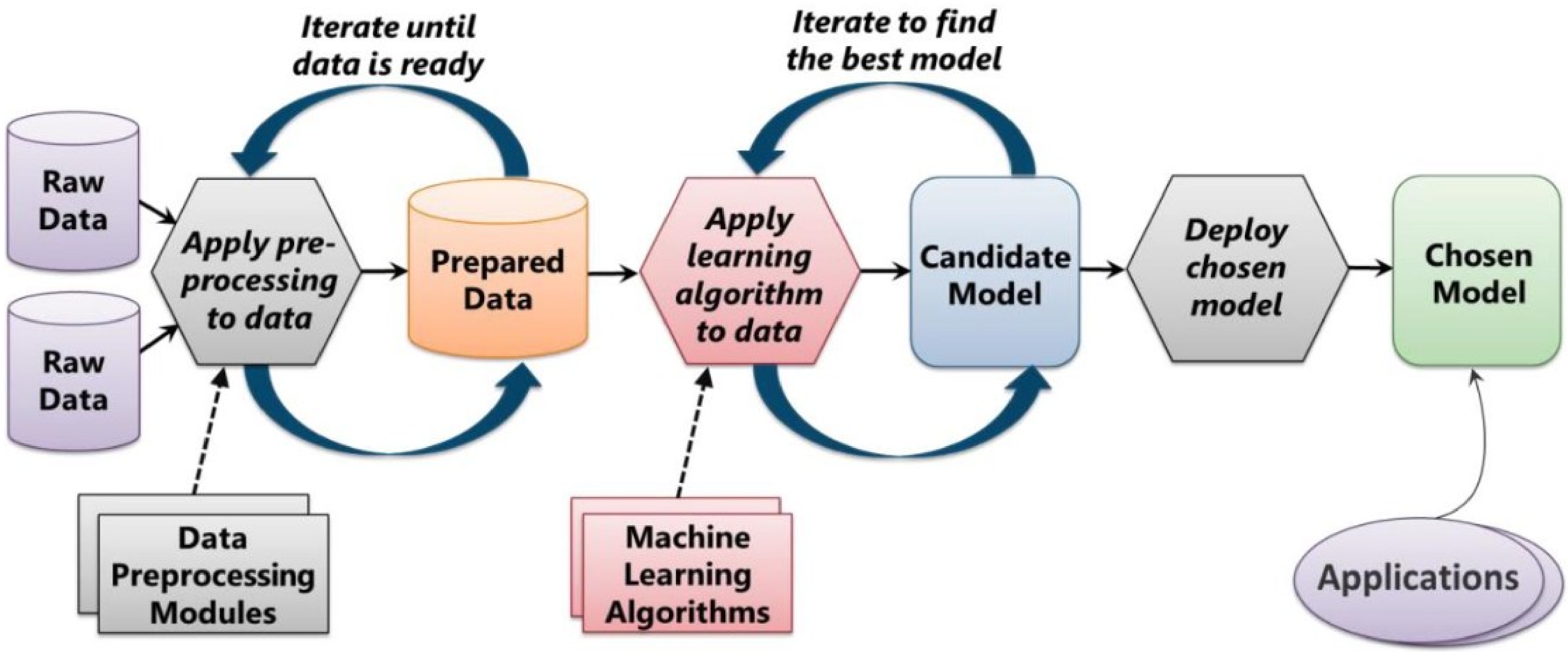
Kreuzvalidierung



- Gegen high Bias
  - Länger Trainieren
  - Komplexeres Modell trainieren
  - Mehr Features
  - Neues Modell trainieren

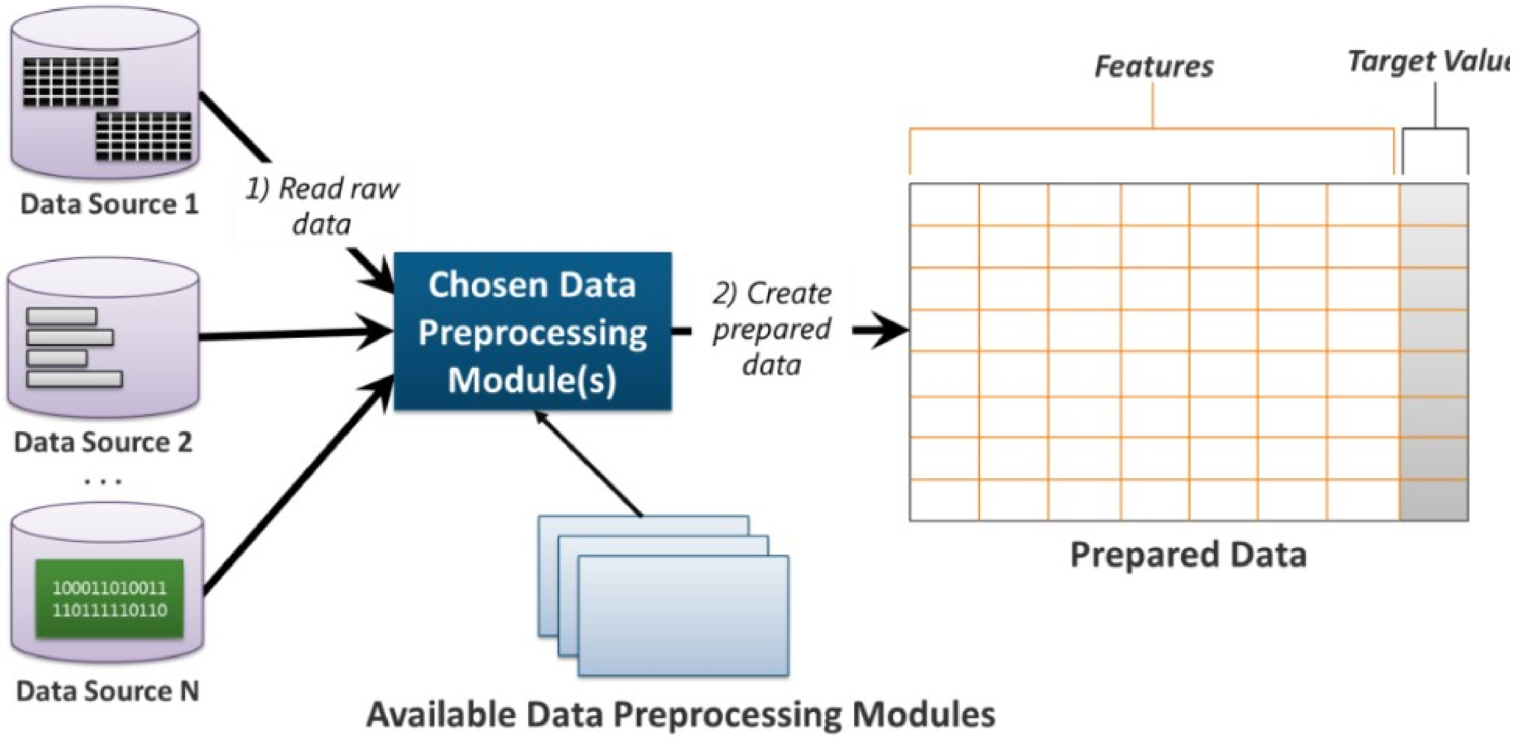
- Gegen high Variance
  - Mehr Daten
  - Wenige Features
  - Neues Modell trainieren

Das generelle Machine Learning vorgehen



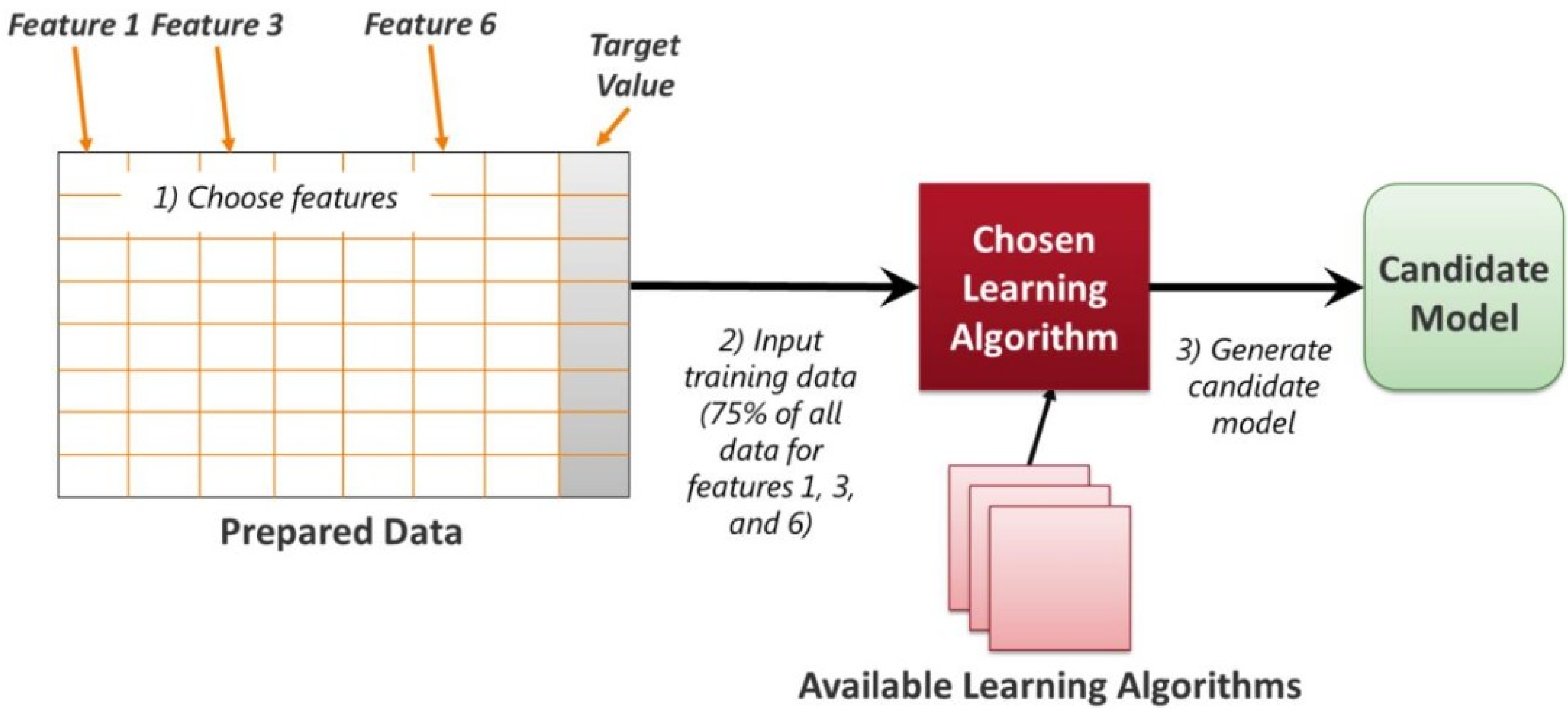
- Quelle: Chappell: Introduction to Machine Learning, 2015" p.5

Das generelle Machine Learning vorgehen



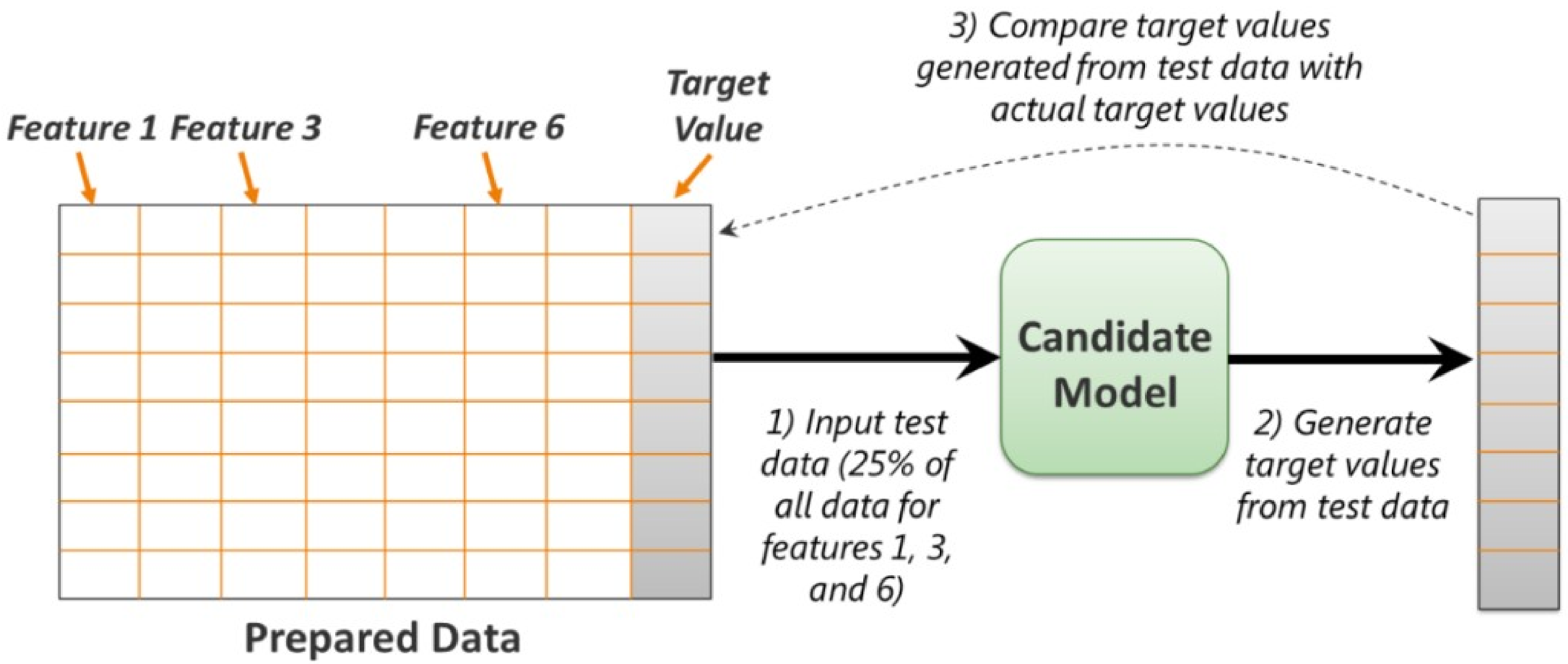
- Quelle: Chappell: Introduction to Machine Learning, 2015" p.10

Das generelle Machine Learning vorgehen



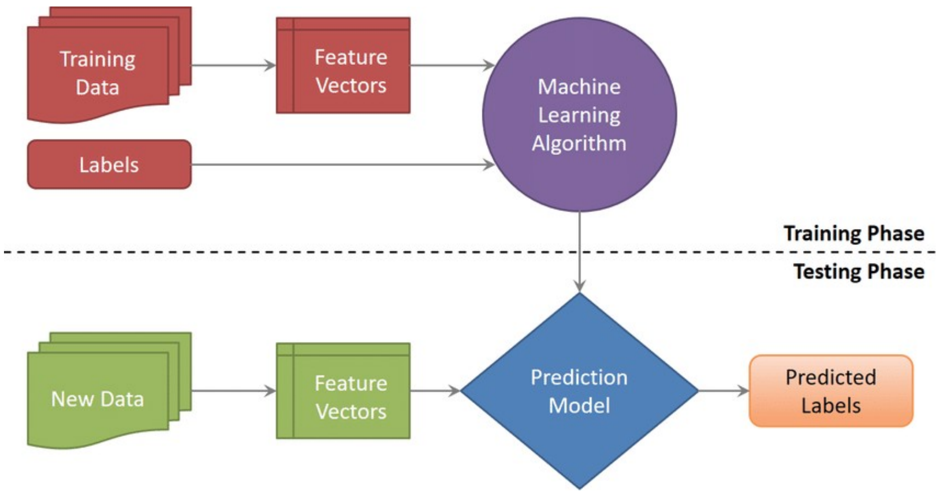
- Quelle: Chappell: Introduction to Machine Learning, 2015" p.12

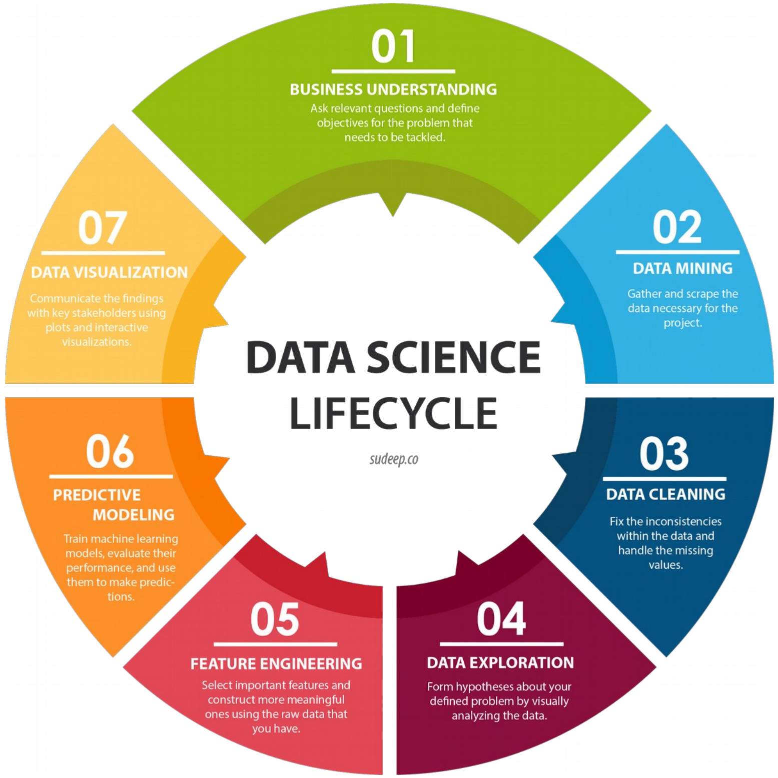
Das generelle Machine Learning vorgehen



- Quelle: Chappell: Introduction to Machine Learning, 2015" p.14

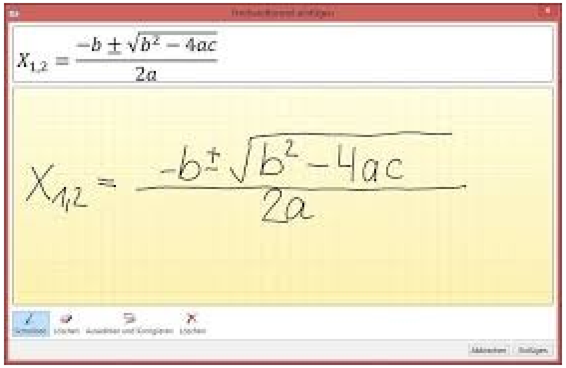
Das generelle Machine Learning vorgehen





Computer Vision

- Computer Vision ist definiert als ein Bereich, das Techniken entwickeln will, die Computern helfen, den Inhalt digitaler Bilder wie Fotos und Videos zu "sehen" und zu verstehen.
- Eine klassische Anwendung des Computer Vision ist die Handschrifterkennung



https://www.youtube.com/watch?v=OcycT1Jwsns

Workshop 1

Object Detection anwenden und die Qualität bewerten.

Buch Unterkapitel 3.4.3 und 3.5.2

Workshop 2

Teil 1

Classification mit Naive Bayes

Teil 2

Classification von Emails in Spam/Nicht Spam mit Decision Tree

https://medium.com/machine-learning-101/chapter-3-decision-tree-classifier-coding-ae7df4284e99