Einführung in das Maschinelle Lernen I

Vorlesung “Computerlinguistische Techniken”

Alexander Koller

26. Januar 2015

Maschinelles Lernen

- Maschinelles Lernen (Machine Learning): äußerst aktiver und für CL nützlicher Zweig der Künstlichen Intelligenz.

- Grundaufgabe:
  - aus Beobachtungen ein Modell lernen
  - für neue, ungesehene Daten Vorhersagen treffen

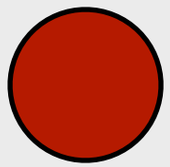
Grundaufgaben

- *Klassifikation:* Zu jeder Instanz gehört eine Klasse aus einer endlichen Menge; finde für neue Instanzen die richtige Klasse.

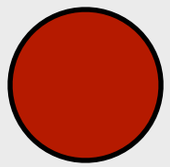
- *Regression:* Zu jeder Instanz gehört eine Zahl; finde für neue Instanzen die (ungefähr) richtige Zahl.

- Jede Instanz wird durch die Werte definiert, die vorgegebene *Features* annehmen.

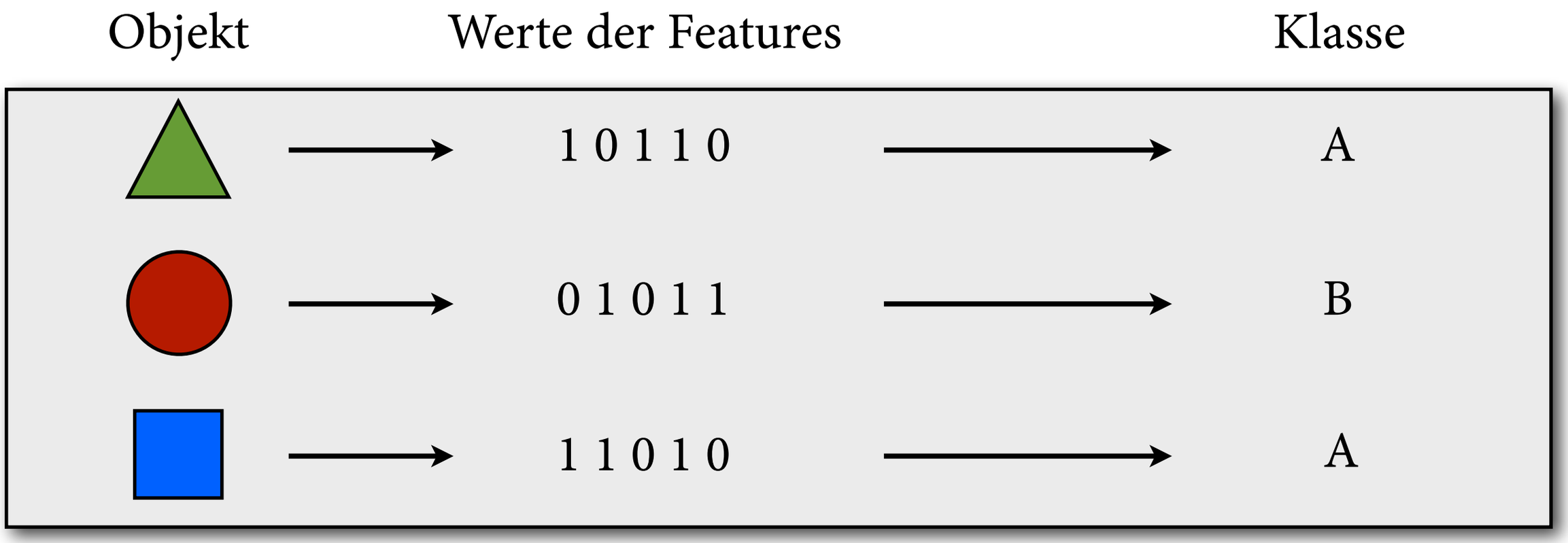
Klassifikation

Objekt	Werte der Features	Klasse
		A
		B
		A

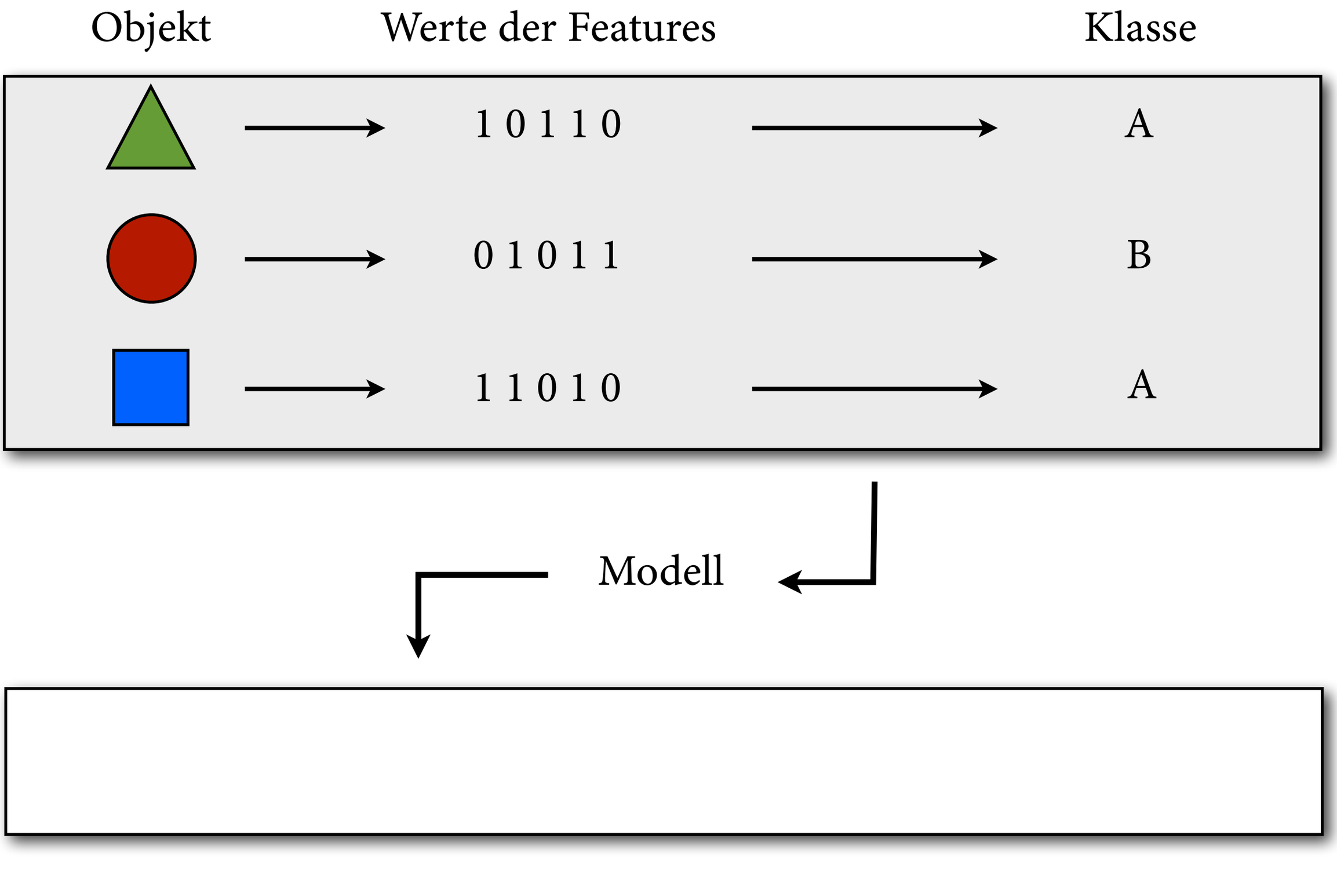
Klassifikation

Objekt	Werte der Features	Klasse
	1 0 1 1 0	A
	0 1 0 1 1	B
	1 1 0 1 0	A

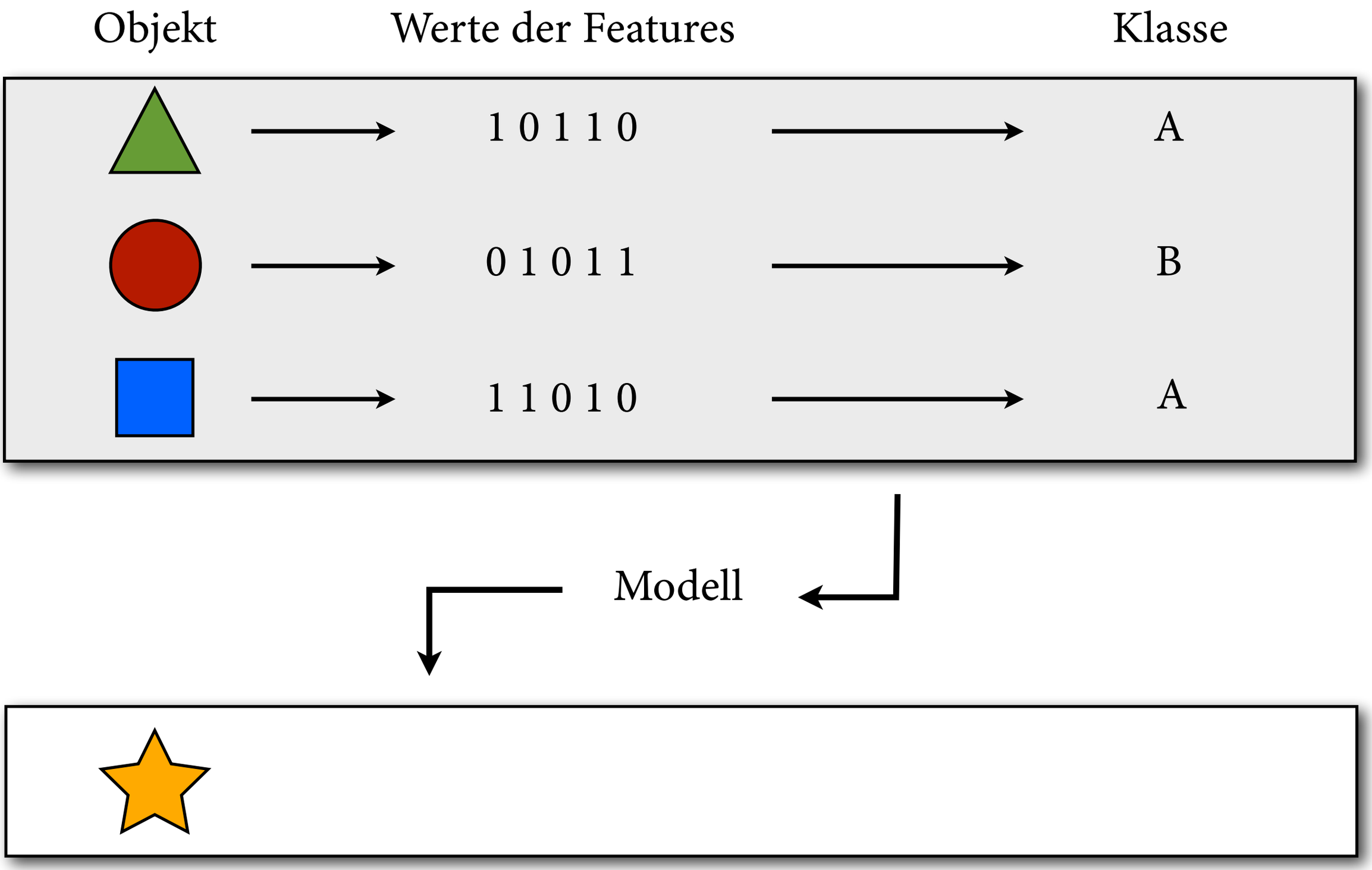
Klassifikation



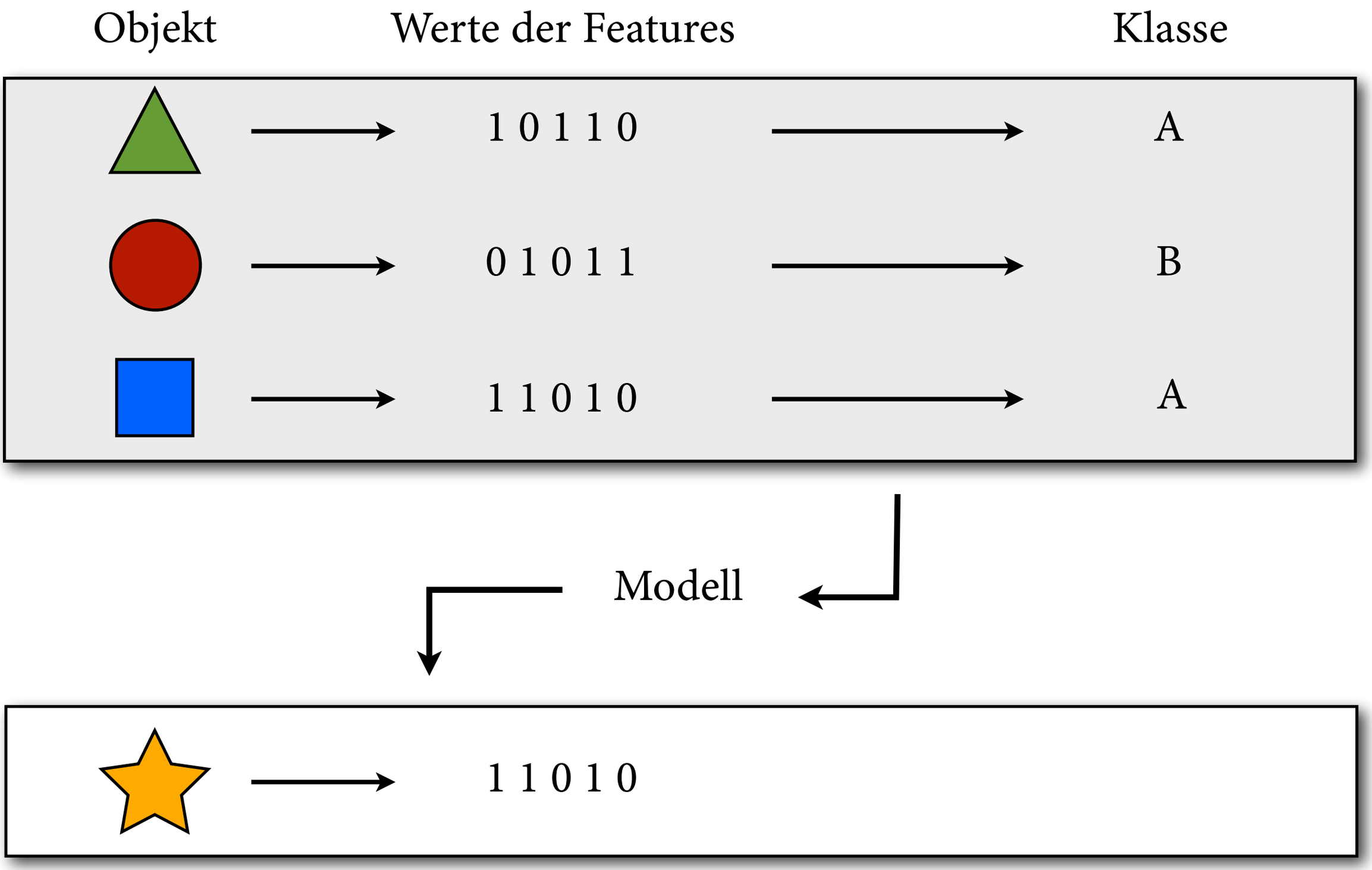
Klassifikation



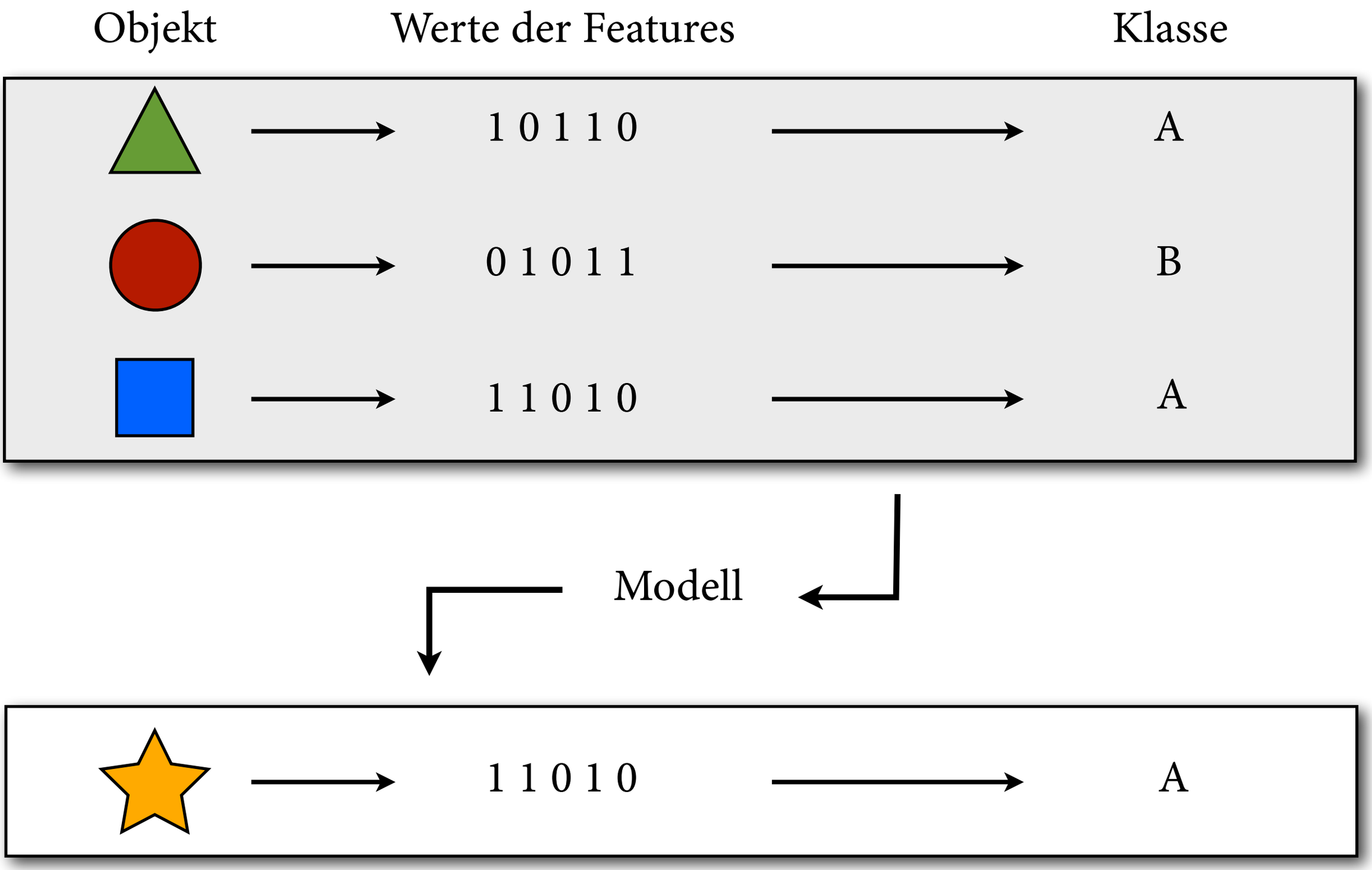
Klassifikation



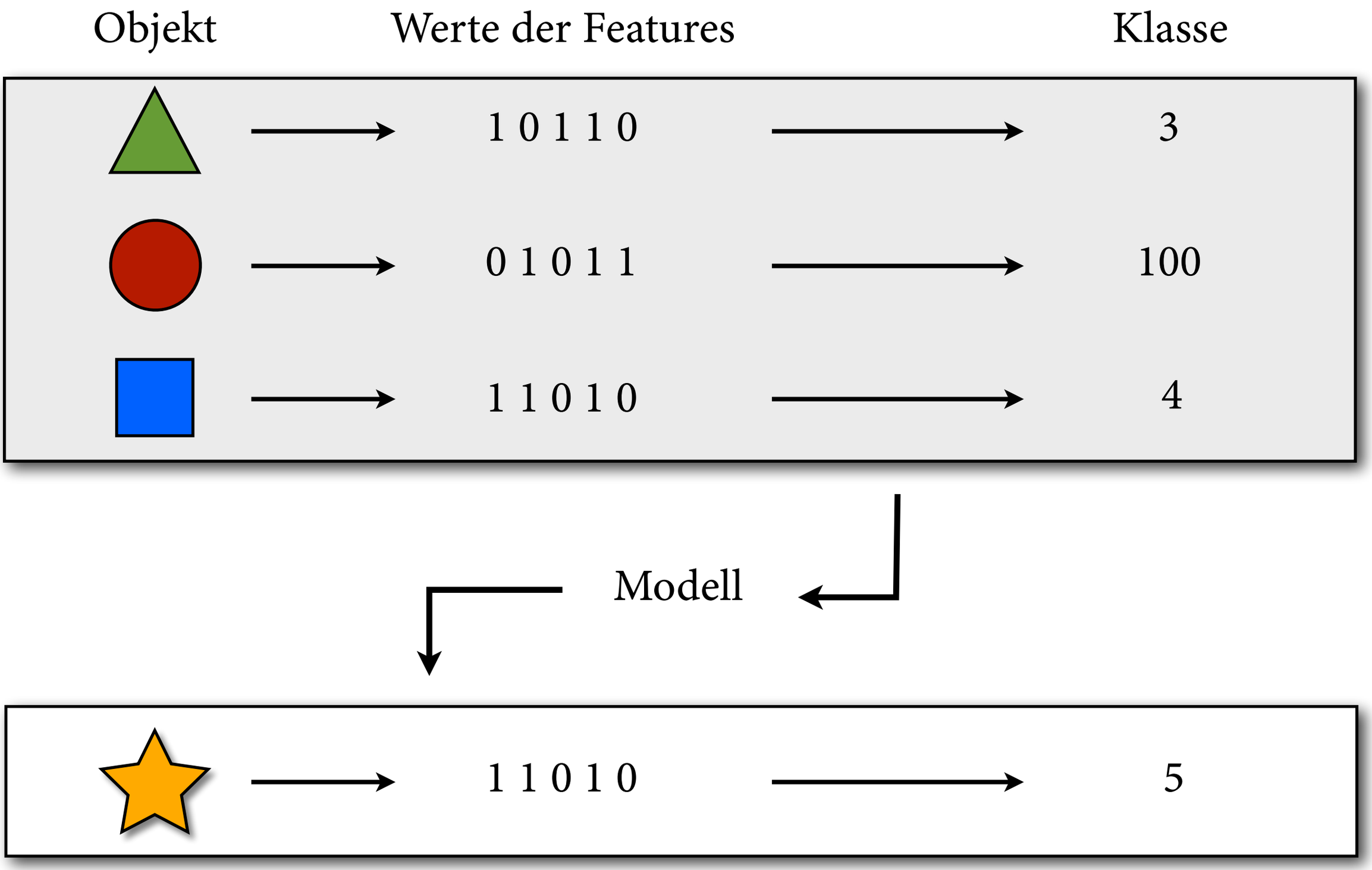
Klassifikation



Klassifikation



Regression


Objekt
Werte der Features
Klasse
1 0 1 1 0
3
0 1 0 1 1
100
1 1 0 1 0
4
Modell
1 1 0 1 0
5

Grundansätze

- *Überwachtes (supervised)* Lernen: In den Trainingsdaten ist zu jeder Instanz die richtige Klasse angegeben.

- *Unüberwachtes (unsupervised)* Lernen: Trainingsdaten sind nicht mit Klassen annotiert.
  - Wahrscheinlichste Klassen raten, z.B. EM-Algorithmus
  - Instanzen zu “natürlichen” Klassen zusammenfassen = Clustering

Beispiel

Wir betrachten zunächst *überwachte* Klassifikation.
Beispiel: Pilze.

Instanzen

Features

Klasse

	Hutfarbe	Hutform	Geruch	essbar?
	r	b	s	nein
	w	b	n	ja
	y	c	s	ja
	w	f	f	nein

Beispiel

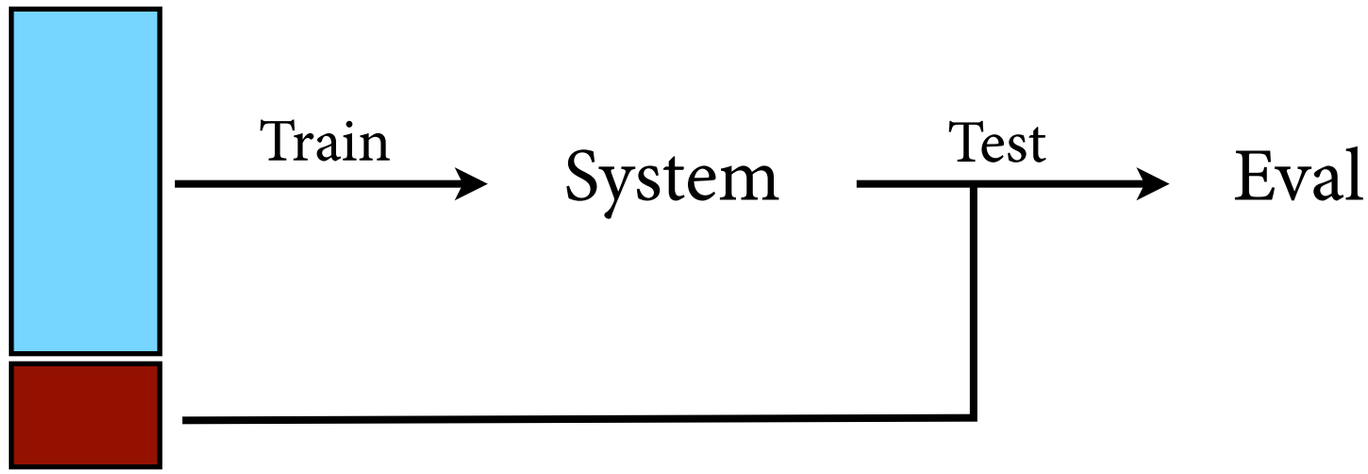
- Angenommen, wir haben folgendes Modell gelernt (≈ *Entscheidungsbaum*):
  - Hutfarbe = rot (r) ⇒ nicht essbar
  - Geruch = faulig (f) ⇒ nicht essbar
  - sonst essbar

- Dann können wir neue Instanzen klassifizieren:

Hutfarbe	Hutform	Geruch	essbar?
r	c	a	nein
w	b	l	ja

Evaluation von Klassifikatoren

- Um Klassifikationsalgorithmus zu evaluieren, bekannter Ansatz:



- Accuracy (= Anteil der korrekt klassifizierten Testinstanzen) wird auch hier verwendet.

- Für binäre Klassifikationsprobleme (= genau 2 Klassen) auch Precision, Recall, f-Score.

Memory-Based Learning

- Ein sehr einfacher Algorithmus für Klassifikation ist Memory-Based Learning
  (= k-nearest-neighbor learning).

- Idee von 1-nearest-neighbor:
  - angenommen, wir haben eine Ähnlichkeitsfunktion auf Instanzen
  - Training = wir speichern alle Instanzen
  - Klasse von neuer Instanz *a* = Klasse derjenigen Trainingsinstanz, die zu *a* am ähnlichsten ist.

Beispiel

Beispiel

essbar? = nein

Beispiel

essbar? = nein

essbar? = ja

Beispiel

essbar? = nein

essbar? = ja

essbar? = ja

Beispiel

essbar? = nein

essbar? = ja

essbar? = ja

essbar? = nein

Beispiel

essbar? = nein

a

essbar? = ja

essbar? = ja

essbar? = nein

Beispiel

essbar? = nein

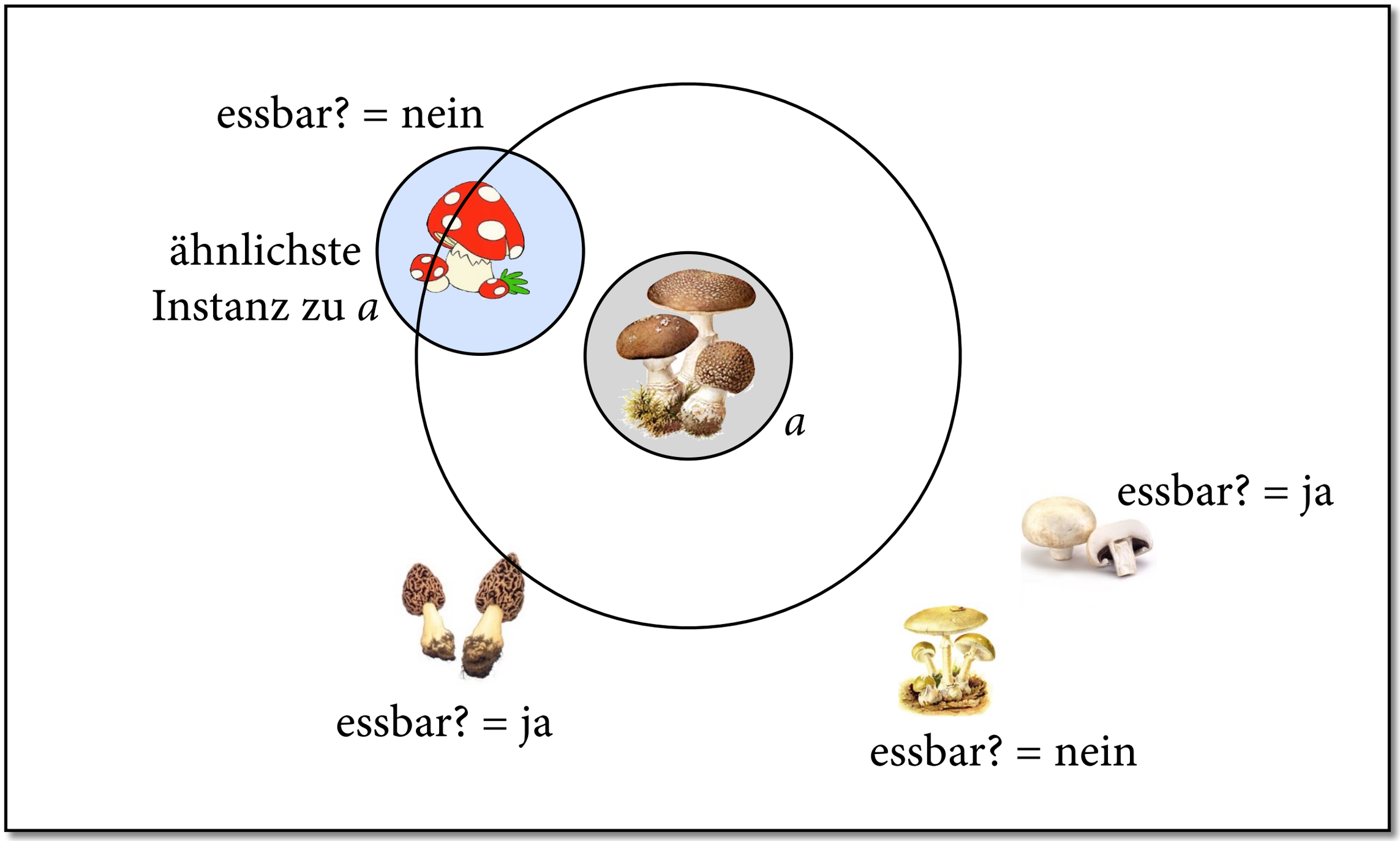
a

essbar? = ja

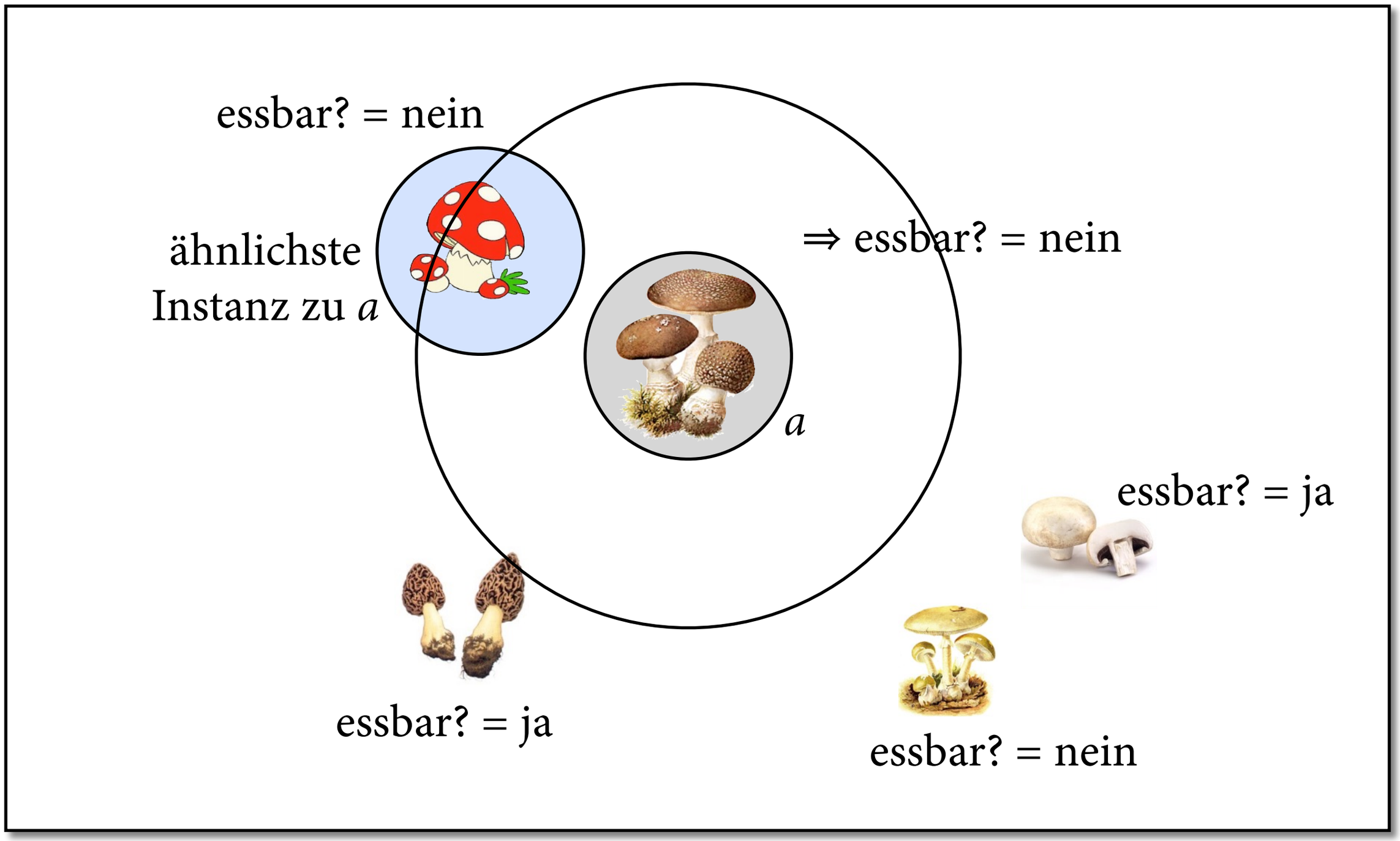
essbar? = ja

essbar? = nein

Beispiel


essbar? = nein
ähnlichste
Instanz zu *a*
a
essbar? = ja
essbar? = ja
essbar? = nein

Beispiel


essbar? = nein
ähnlichste
Instanz zu *a*
⇒ essbar? = nein
a
essbar? = ja
essbar? = ja
essbar? = nein

k-nearest-neighbors

- Verallgemeinerung von 1-nearest-neighbor:
  - betrachte statt dem einen nächsten Nachbarn die k nächsten Nachbarn für ein festes $k \geq 1$
  - Klasse der neuen Instanz = Mehrheitsklasse der k nächsten Nachbarn
- Konzeptuell sehr einfach, effiziente Implementierung nicht ganz einfach.

Ähnlichkeitsfunktion

- Wahl der Ähnlichkeitsfunktion ist entscheidend.
- Eine Möglichkeit ("overlap metric"):

$$\mathrm{sim}(x, y) = \sum_{f} \delta(f(x), f(y))$$

f(x) Wert des Features f auf Instanz x

δ ist Kronecker-Symbol, d.h.
δ(a,b) = 1 gdw a = b, sonst δ(a,b) = 0.

Probabilistische Klassifikation

- Klassifikation: Lerne aus Trainingsinstanzen (x, c) ein Modell, um neuen Instanzen x die richtige Klasse c zuzuweisen.

- Probleme mit k-NN:
  - bekomme nur Klasse, keine Konfidenz
  - Ansatz ad-hoc, nicht auf Prinzipien von W. erklärt

Probabilistische Klassifikation

- Wir betrachten hier zwei wichtige Ansätze auf Grundlage von W.modellen:
  - Naive Bayes (heute)
  - Maximum Entropy (nächstes Mal)

- Illustriere an konkreten Beispielen, aber anwendbar auf beliebige Klassifikationsprobleme.

Naive Bayes: Motivation

- Betrachte Textklassifikation, z.B. für Spam:

... Nigeria ... bank ...	Spam
... Viagra ... Tabletten ...	Spam
... Vorlesung ... fällt aus ...	kein Spam

- Formal: Klassifiziere String $w = w_1 \ldots w_n$ binär, mit den Klassen “Spam” oder “kein Spam”.

- Allgemein: endliche Menge von Klassen;
hier binäres Klassifikationsproblem.

Naive Bayes: Grundidee

- Angenommen, wir hätten W.verteilung P(c|w). Dann Klassifikation:

$$\underset{c}{\operatorname{argmax}}\, P(c|w)$$

- Wir hätten auch ein Maß für Konfidenz, die *odds ratio* O(c):

$$O(\text{Spam}) = \frac{P(c = \text{Spam}|w)}{P(c = \text{keinSpam}|w)}$$

- Problem: Wie soll man P(c|w) schätzen?

Bayes'sche Regel

- Mit der Bayes'schen Regel kann man zwischen P(c|w) und P(w|c) umrechnen:

$$P(c|w) = \frac{P(w|c) \cdot P(c)}{P(w)}$$

- A-*posteriori*-W. P(c|w) entsteht durch Update der a-*priori*-W. P(c) durch *Likelihood* P(w|c).

- Da w beobachtet wurde, ist P(w) irrelevant, also

$$\underset{c}{\operatorname{argmax}}\, P(c|w) = \underset{c}{\operatorname{argmax}}\, P(w|c) \cdot P(c)$$

Naive Bayes

- P(c) leicht zu schätzen als rel. Häufigkeit der Klassen.

- P(w|c) viel unklarer:

$$P(w|c) = P(w_n|c, w_1, \ldots, w_{n-1}) \cdot \ldots \cdot P(w_1|c)$$

- Grundannahme von Naive Bayes: Alle P(w_i| c, ...) statistisch unabhängig. Damit viel einfacher:

$$P(w|c) = P(w_n|c) \cdot \ldots \cdot P(w_1|c)$$

- Einzelne P(w_i|c) als rel. Häufigkeit schätzbar (als C(w_i, c) / C(c)).

Beispiel

Nigeria ... prince ... bank	Spam
... Viagra ... Tabletten ...	Spam
Nigeria ... president ... bank	Spam
... Vorlesung ... fällt aus ...	kein Spam

P(Spam) = 0.75 P(Nigeria|Spam) = 0.66 P(Viagra|Spam) = 0.33

P(kein Spam) = 0.25 P(bank|Spam) = 0.66 P(Vorlesung|Spam) = 0

P(Spam|Nigeria ... Viagra ...) ~ P(Nigeria|Spam) * P(Viagra|Spam) * P(Spam) = 0.327

P(kein Sp.|Nigeria ... Viagra ...) ~ P(Nigeria|k.Sp.) * P(Viagra|k.Sp.) * P(k.Sp.) = 0

Daher: Dokument als “Spam” klassifizieren.

Vor- und Nachteile

- Unabhängigkeitsannahme in der Praxis normalerweise verletzt.

- Trotzdem funktioniert NB oft gut, weil korrekte Klassifikation nur odds ratio > 1 vs. < 1 erfordert.

- Naive Bayes funktioniert oft schon für kleine Menge an Trainingsdaten.

Zusammenfassung

- Maschinelles Lernen:
  - Klassifikation vs. Regression
  - überwacht vs. unüberwacht
- Memory-Based Learning: k-nearest-neighbor
- Probabilistischer Ansatz: Naive Bayes