



max planck institut
informatik

Ideen und Konzepte der Informatik

Maschinelles Lernen

Kurt Mehlhorn

Übersicht

- Lernen: Begriff
- Beispiele für den Stand der Kunst
- Spamerkennung
- Handschriftenerkennung
 - mit und ohne Trainingsdaten
- Gesichts- und Personenerkennung
- Neuronale Netzwerke, Teil 2 der Vorlesung
 - Maschinelles Sehen



Lernen

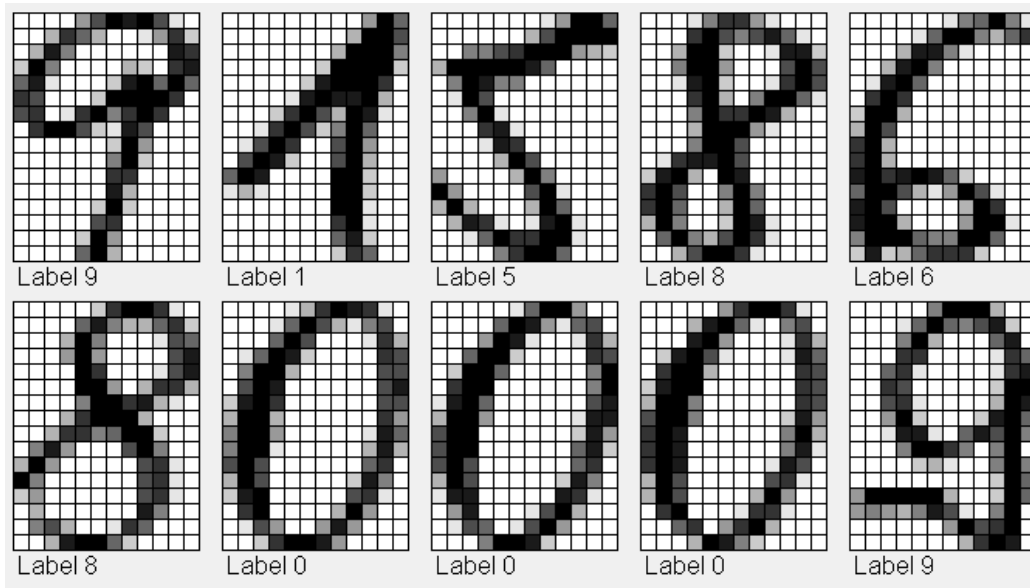
- Fähigkeit, Verhalten zu verbessern aufgrund von Erfahrungen
- Verallgemeinern von Erfahrungen
- Informatik: Programmieren durch Beispiele anstatt durch Angabe eines Programms
- Ein Lernalgorithmus entwickelt das Programm aus (vielen) Daten

Potential

- Suchmaschinen mit Bild / Sprach-Anfragen
- Personenerkennung, Überwachung
- Selbstfahrende Autos, Roboter
- Bessere Benutzerschnittstellen, Sprache, Gesten
- Maschinelle Übersetzung
- Sachbearbeiter
- Intelligente Assistenten

Arten von Lernen

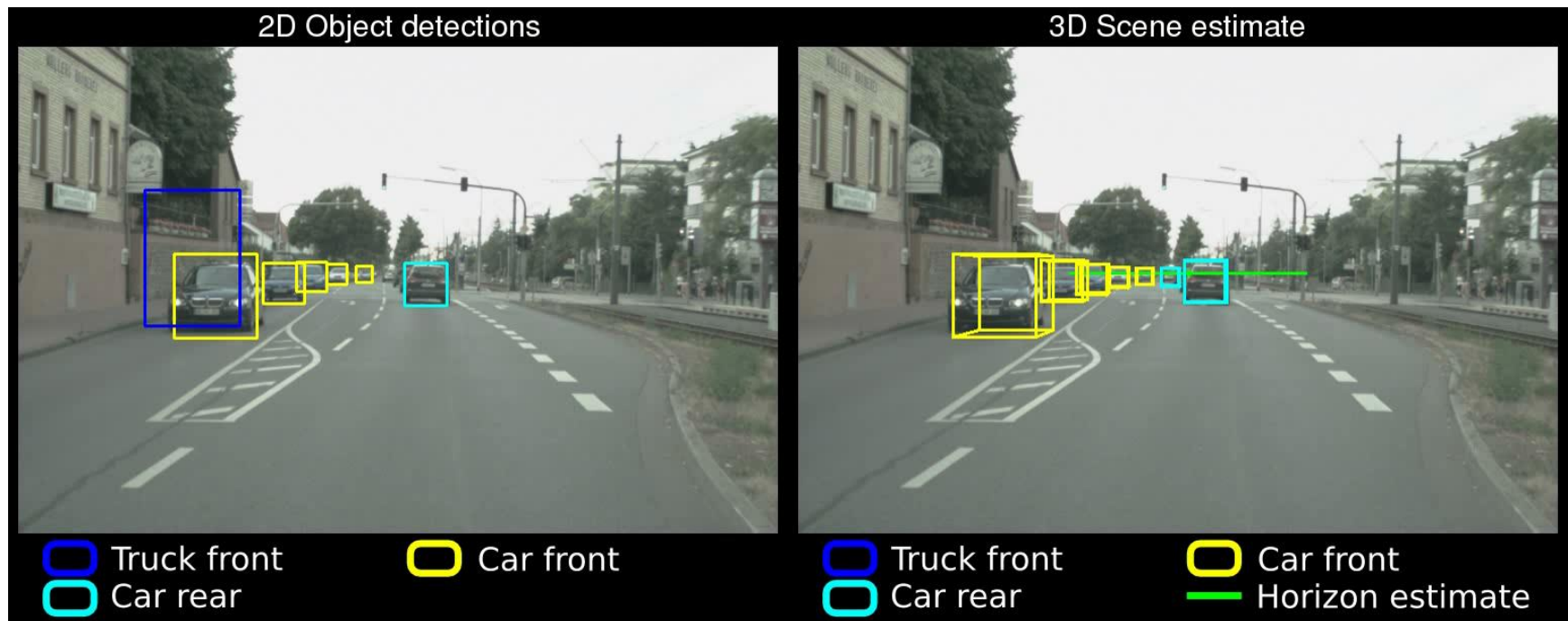
- **Supervised:** mit Trainingsdaten oder sogar mit Lehrer



- **Unsupervised:** ohne Trainingsdaten; dann ist es mehr Entdecken als Lernen

Objekterkennung

Abteilung Schiele: MPI für Informatik

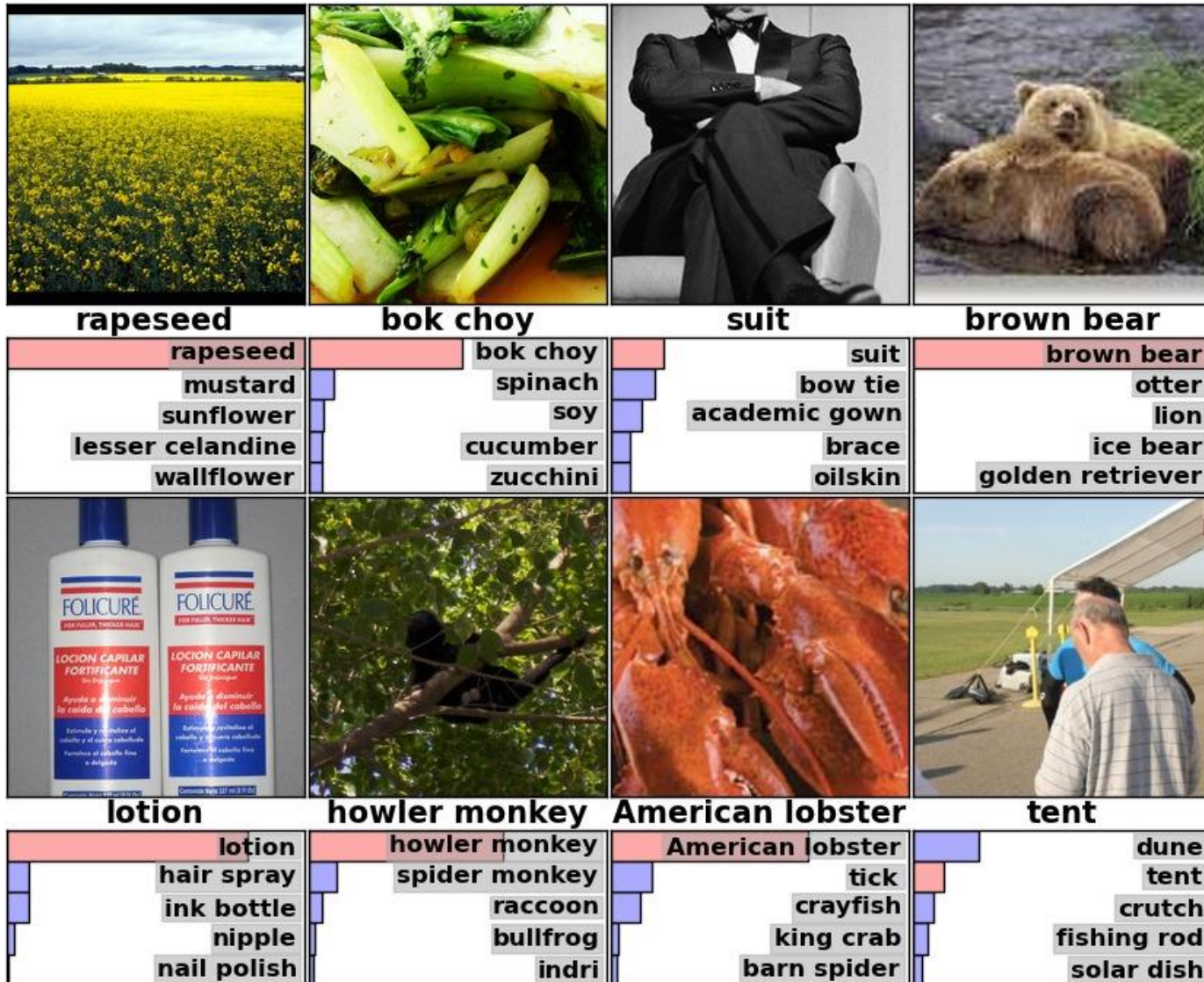


Personenerkennung

Abteilung Schiele: MPI für Informatik

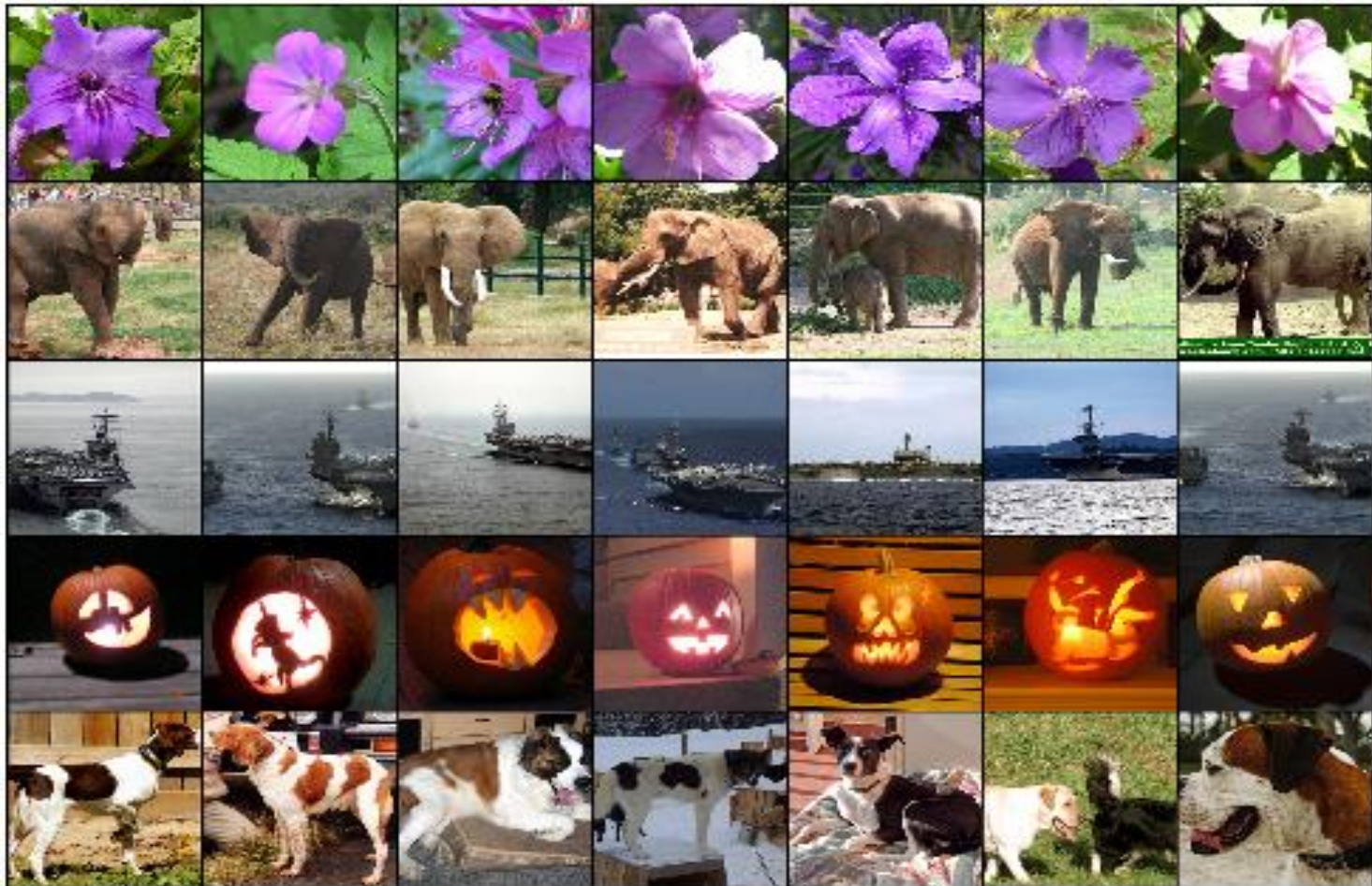


Klassifikation (Krizhevsky et al., 2012)



Suche (Krizhevsky et al., 2012)

Linke Spalte: Suchanfrage, andere Spalten: ähnliche Bilder im Fundus



Spamerkennung

Spam, Junk = unerwünschte Nachrichten

Ham = erwünschte Nachrichten

Wir lernen einen Bayes'schen Filter kennen.

Spam (ursprünglich): Spiced Ham, Markenname für sehr verbreitetes Dosenfleisch,

Bayes'sche Regel

(englischer Pfarrer und Mathematiker, 1701 – 1761)

In einem Sack sind 900 Äpfel und 100 Paprika. Von den Äpfeln sind 10% rot und 90% grün. Bei den Paprika sind es jeweils 50%.

Ich entnehme eine Frucht zufällig. Sie ist rot. Was für eine Frucht ist es?

- Bayes: Entscheide dich für den wahrscheinlicheren Fall.

Bayes'sche Regel

In einem Sack sind 900 Äpfel und 100 Paprika. Von den Äpfeln sind 10% rot und 90% grün. Bei den Paprika sind es jeweils 50%.

$$P(\text{Apfel} \mid \text{rot}) = \frac{\# \text{ rote Äpfel}}{\# \text{ rote Früchte}}$$

(Prozentsatz der Äpfel unter den roten Früchten)

Bayes'sche Regel

In einem Sack sind 900 Äpfel und 100 Paprika. Von den Äpfeln sind 10% rot und 90% grün. Bei den Paprika sind es jeweils 50%.

$$P(\text{Apfel} \mid \text{rot}) = \frac{P(\text{Apfel und rot})}{P(\text{rot})} = \frac{P(\text{rot} \mid \text{Apfel}) P(\text{Apfel})}{P(\text{rot} \mid \text{Apfel}) P(\text{Apfel}) + P(\text{rot} \mid \text{Paprika}) P(\text{Paprika})}$$

Spam versus Ham (Junk-Mail)

- Absenderbasiert
 - E-Mail von Bekannten ist kein Spam
 - Schwarze Listen
- Inhaltsbasiert
 - Nutzer klassifiziert E-Mails als gut und schlecht; System lernt daraus; Nutzer muss immer weniger eingreifen

Inhaltsbasierte Filter

- In der Trainingsphase lernen wir
 - Wahrscheinlichkeit von Ham und Spam
 - Jeweils Wahrscheinlichkeiten für Worte
- 70% ist Ham, 30% ist Spam

- Ham

	Freund	Vorle- sung	Algorith- mus	Geld	Viagra	schnell
	0.1	0.3	0.3	0.1	0.1	0.1

- Spam

	Freund	Vorle- sung	Algorith- mus	Geld	Viagra	schnell
	0.2	0.1	0.1	0.2	0.3	0.1

Trainingsphase

- Nutzer klassifiziert E-Mails als Spam und Ham (damit beide Wahrscheinlichkeiten)
- Sei n die Gesamtlänge meiner guten E-Mails (in Worten), sei v die Anzahl der Vorkommen eines bestimmten Wortes
- Wahrscheinlichkeit des Wortes in Ham

$$= \frac{v}{n}$$

Inhaltsbasierte Filter (Bayes Modell)

- Wahrscheinlichkeitsverteilung auf Worten

- Ham

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.1	0.3	0.3	0.1	0.1	0.1

- Spam

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.2	0.1	0.1	0.2	0.3	0.1

- $P(\text{Text} \mid \text{Ham}) = \text{Produkt der Ham-Wahrscheinlichkeiten der Worte im Text}$

Inhaltsbasierte Filter

- Ham

Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
0.1	0.3	0.3	0.1	0.1	0.1

- Spam

Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
0.2	0.1	0.1	0.2	0.3	0.1

- Viagra Geld Freund
 - $P(\text{Text} \mid \text{Ham}) = 0.1 \cdot 0.1 \cdot 0.1 =$
 - $P(\text{Text} \mid \text{Spam}) =$
- Bei 70% Ham und 30% Spam

Inhaltsbasierte Filter

- Ham

Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
0.1	0.3	0.3	0.1	0.1	0.1

- Spam

Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
0.2	0.1	0.1	0.2	0.3	0.1

- Viagra Geld Freund

- $P(\text{Text} \mid \text{Ham}) = 0.1 \cdot 0.1 \cdot 0.1 =$

- $P(\text{Text} \mid \text{Spam}) =$

- Bei 70% Ham und 30% Spam ist $P(\text{Spam} \mid \text{Text}) = \dots$

Inhaltsbasierte Filter

- Viagra Geld Freund
 - $P(\text{Text} \mid \text{Ham}) = 0.1 \cdot 0.1 \cdot 0.1 = 1/1000$
 - $P(\text{Text} \mid \text{Spam}) = 12/1000$
- Bei 70% Ham und 30% Spam erhalten wir
- $$P(\text{Spam} \mid \text{Text}) = \frac{P(\text{Text} \mid \text{Spam})P(\text{Spam})}{(P(\text{Text} \mid \text{Ham})P(\text{Ham}) + P(\text{Text} \mid \text{Spam})P(\text{Spam}))} =$$
$$12 \cdot 3 / (1 \cdot 7 + 12 \cdot 3) = 36/43.$$

Inhaltsbasierte Filter

- Ham

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.1	0.3	0.3	0.1	0.1	0.1

- Spam

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.2	0.1	0.1	0.2	0.3	0.1

- Vorlesung Algorithmus schnell
 - Falls Ham:
 - Falls Spam:
- 70% Ham und 30% Spam

Inhaltsbasierte Filter

- Ham

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.1	0.3	0.3	0.1	0.1	0.1

- Spam

	Freund	Vorlesung	Algorithmus	Geld	Viagra	schnell
	0.2	0.1	0.1	0.2	0.3	0.1

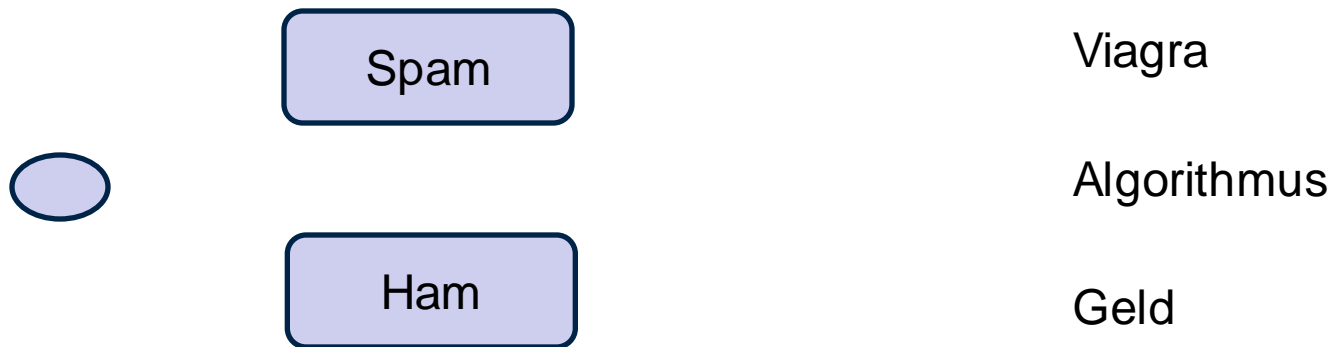
- Viagra Algorithmus schnell
 - Falls Ham: $0.1 \cdot 0.3 \cdot 0.1 = \frac{3}{1000}$
 - Falls Spam: $0.3 \cdot 0.1 \cdot 0.1 = \frac{3}{1000}$
- Bei 10% Ham und 90% Spam

Nutzungsphase

- Nutzungsphase: System klassifiziert.
- Verteilung wird weiter trainiert (seltene Worte).
- Nutzer kann widersprechen.
- Spammer lernen auch dazu: V!agra statt Viagra.

Zusammenfassung

- Wir haben Modell, wie Ereignisse (E-Mails) erzeugt werden



- Lernen das Modell in der Trainingsphase.
- Geben für jedes Ereignis die wahrscheinlichste Erklärung (Bayes).
- Klassifizierung in: Geschäftspost, Privatpost, Spam.

Ziffernerkennung Übersicht

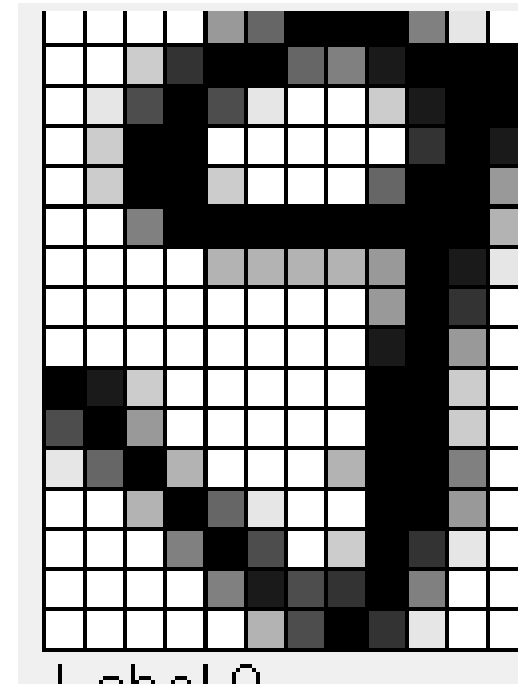
- Darstellung von Bildern in Rechnern
- Trainingsdaten: handgeschriebene Ziffern
- Supervised Learning: mit Label (die Ziffer)
- Unsupervised Learning: ohne Label

Bilder = Matrizen von Zahlen

Ziffer = 12 x 16 Matrix von
Grauwerten in [0,1]

Vektor von Grauwerten der
Länge 192

0.0	0.0	0.0	0.2	0.3	0.4	0.8	1.0	1.0	0.7	0.3	0.1
0.0	0.5	0.8	0.9	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5
0.3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.9
0.3	1.0	1.0	1.0	1.0	1.0	0.8	1.0	1.0	1.0	1.0	0.8
0.3	1.0	1.0	1.0	1.0	0.8	0.1	0.9	1.0	1.0	0.8	0.2
0.0	0.7	1.0	1.0	1.0	0.8	0.8	1.0	1.0	1.0	0.4	0.0
0.0	0.2	1.0	1.0	1.0	1.0	1.0	1.0	0.9	0.3	0.0	0.0
0.0	0.1	0.7	1.0	1.0	1.0	1.0	0.8	0.1	0.0	0.0	0.0
0.0	0.6	1.0	1.0	1.0	1.0	0.9	0.1	0.0	0.0	0.0	0.0
0.6	1.0	1.0	1.0	1.0	1.0	1.0	0.3	0.0	0.0	0.0	0.0
0.8	1.0	1.0	0.5	0.1	0.7	1.0	0.8	0.2	0.0	0.0	0.0
0.5	1.0	1.0	0.3	0.0	0.0	0.9	1.0	0.9	0.1	0.0	0.0
0.4	1.0	1.0	0.3	0.0	0.0	0.5	1.0	1.0	0.5	0.0	0.0
0.0	0.4	1.0	1.0	0.5	0.3	0.5	1.0	1.0	1.0	0.2	0.0
0.0	0.0	0.5	1.0	1.0	1.0	1.0	1.0	1.0	0.8	0.1	0.0
0.0	0.0	0.0	0.2	0.5	0.7	1.0	1.0	0.9	0.3	0.0	0.0



Ihr Gehirn sieht
Ziffern, Ihr Auge und
Computer sehen nur
eine Matrix von
Grauwerten

Trainingsdaten



Ziemlich gutmütig

Grundidee

- Zwei Bilder repräsentieren die gleiche Ziffer, wenn die Bilder sich ähnlich sind.
- Ähnlich = ähnliche Grauwertverteilung.
- Vektoren der Grauwerte sind sich ähnlich.
 - Euklidische Distanz zweier Vektoren
 - Cos-Distanz, Winkel zwischen Vektoren

Ähnlichkeit von Vektoren

- Zwei Vektoren x und y sind ähnlich,
 - wenn $x - y$ kurz ist
 - wenn der aufgespannte Winkel klein ist

- Länge eines Vektors $x = (x_1, \dots, x_n)$

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$$

- Winkel zwischen x und y

$$\cos \alpha = \frac{x \cdot y}{\|x\| \cdot \|y\|}$$

Verfahren: Nearest Neighbor

Um die Bedeutung des Bildes p zu finden, finde das Trainingsbild x mit $\text{dist}(p,x)$ minimal (durch lineare Suche über alle Trainingsdaten)

Gib das Label von x aus

- Erkennungsrate bei Euklidischer Distanz 0.934
- Majority of 3 nearest neighbors 0.945
- Majority of 9 nearest neighbors 0.920

- Mit cos-Distanz 0.940
- Majority of 3 nearest neighbors 0.920

Detaillierte Ergebnisse

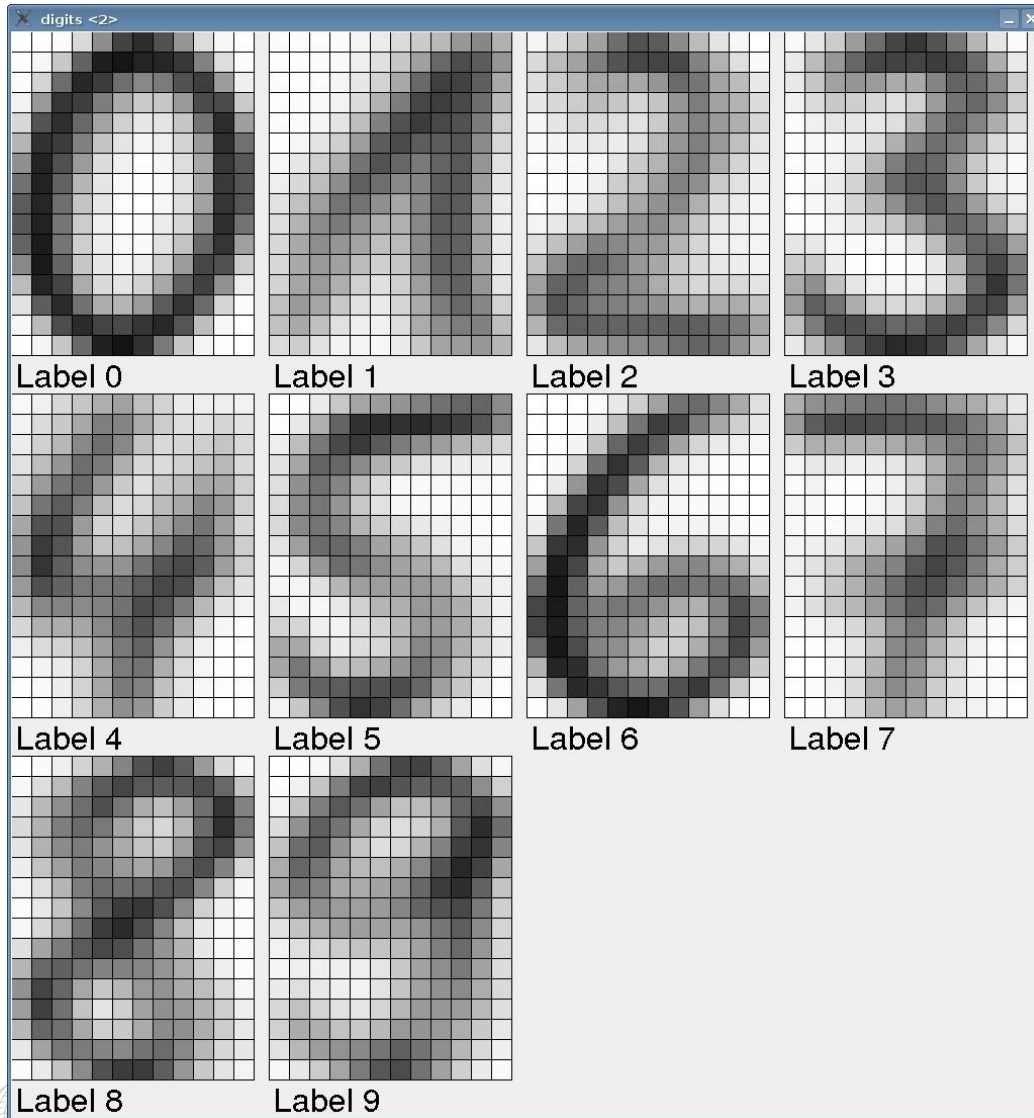
digit = 0 accuracy = 1.0
digit = 1 accuracy = 0.90
digit = 2 accuracy = 0.92
digit = 3 accuracy = 1.0
digit = 4 accuracy = 0.95
digit = 5 accuracy = 0.85
digit = 6 accuracy = 0.84
digit = 7 accuracy = 1.0
digit = 8 accuracy = 0.7
digit = 9 accuracy = 0.94

Klassifizierung ist recht gut,
aber sie dauert sehr **lang**,
da jedes Mal ALLE
Trainingsdaten angeschaut
werden

Klassen → Klassenzentren

- Vorbereitung: Berechne für jede Klasse (Ziffer) das Klassenzentrum durch Durchschnittsbildung (*siehe nächste Folie*)
- Suche: Finde das nächstgelegene Zentrum (10 Vergleiche)
- Erkennungsrate bei Euklidischer Distanz: 0.854
- Mit cos-distance 0.894
- Sehr effizient, aber schlechter

Die Klassenzentren

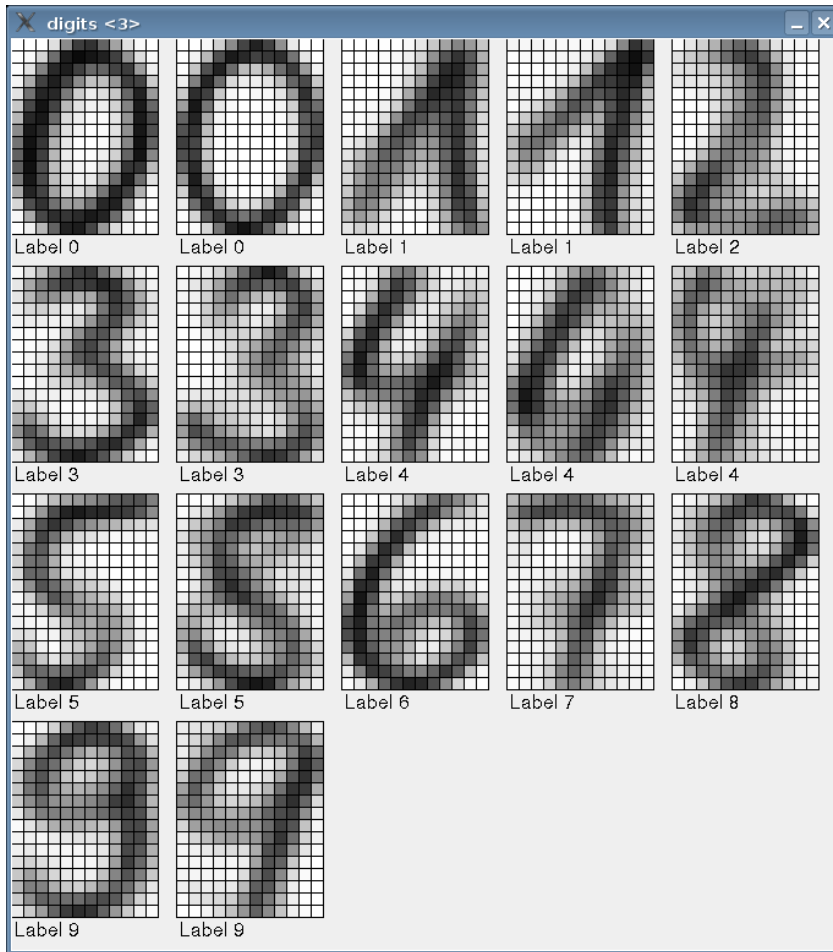


Durchschnitt
(Schwerpunkt einer
Klasse) =
Summe der Elemente
der Klasse / Anzahl der
Elemente

Unsupervised Lernen

- Vorbereiten der Trainingsdaten ist mühsam
- Können wir Klassen entdecken, ohne dass uns Klassenlabels gesagt werden?
- Automatische Klassifizierung durch k -Means Algorithmus
- Danach Vergleich mit den Klassenzentren
- $k = 10$, Rate 0.683
- $k = 17$, 0.733
- with cos-distance,
- $k = 10$, 0.728
- $k = 17$, 0.783
- $k = 30$, 0.864

k-Means Algorithmus



- Automatische Klassifizierung in 17 Klassen
- Danach (!!!) Zuweisung eines Labels per Hand und Wegwerfen von schlechten Zentren
- Identifiziert die zwei Schreibweisen der Neun und der Eins

k -Means Algorithmus

Teilt n Punkte in k Cluster (Haufen) ein.

1. Starte mit k beliebigen (zufälligen) Zentren.
2. Weise jeden Punkt dem nächstgelegenen Zentrum zu und bilde so k Cluster.
3. Berechne für jeden Cluster seinen Schwerpunkt; das sind die neuen Zentren.
4. Gehe nach 2.

Gesichtserkennung

- Gesicht → Vektor
 - Augenfarbe, Abstand der Augen, Abstand Nase – Augen, Hautfarbe, Abstand Ohren, und so weiter, dann wie bei Ziffernerkennung.
- Aufgaben
 - Zwei (gute) Bilder: gleiche Person?
 - Klassifiziere Bilder nach abgebildeter Person
 - Finde Gesicht in einer Datenbank.



Personen



Zugehörig



Fotos von diesem Tag anzeigen

