9. Clusterbildung, Klassifikation und Mustererkennung

Begriffsklärung
(nach Voss & Süße 1991):

Objekt:
wird in diesem Kapitel mit einem zugeordneten *Merkmalstupel* (x_1, ..., x_M) identifiziert (Merkmalsextraktion wird also vorausgesetzt)

Klasse:
bezeichnet hier eine Teilmenge von Objekten, die aus numerischen, logischen, heuristischen oder subjektiven Gründen als zusammengehörig angesehen werden.

Stichprobe:
eine endliche Menge von Merkmalstupeln; eine Teilmenge der Menge aller Objekte.
klassifizierte Stichprobe: den Objekten sind Klassenkenn-zeichen zugeordnet,
unklassifizierte Stichprobe: Menge von Merkmalstupeln ohne Klassenkennzeichen.

Klassifikation:
der Prozess der Zuordnung eines Objektes zu einer Klasse (durch einen Algorithmus oder durch eine subjektive Entscheidung)

Klassifikator:
ein Algorithmus oder Programm, mit dessen Hilfe ein Merkmalstupel einer Klasse zugeordnet werden kann.

Lernen:
die Erarbeitung eines Klassifikators anhand einer klassifizierten oder unklassifizierten Stichprobe.
überwachtes Lernen: mit klassifizierter Stichprobe,
unüberwachtes Lernen: mit unklassifizierter Stichprobe.

Clusterbildung:
die Erarbeitung eines Klassifikators anhand einer unklassifizierten Stichprobe;
die Einteilung der Objektmenge in Teilmengen (Cluster).

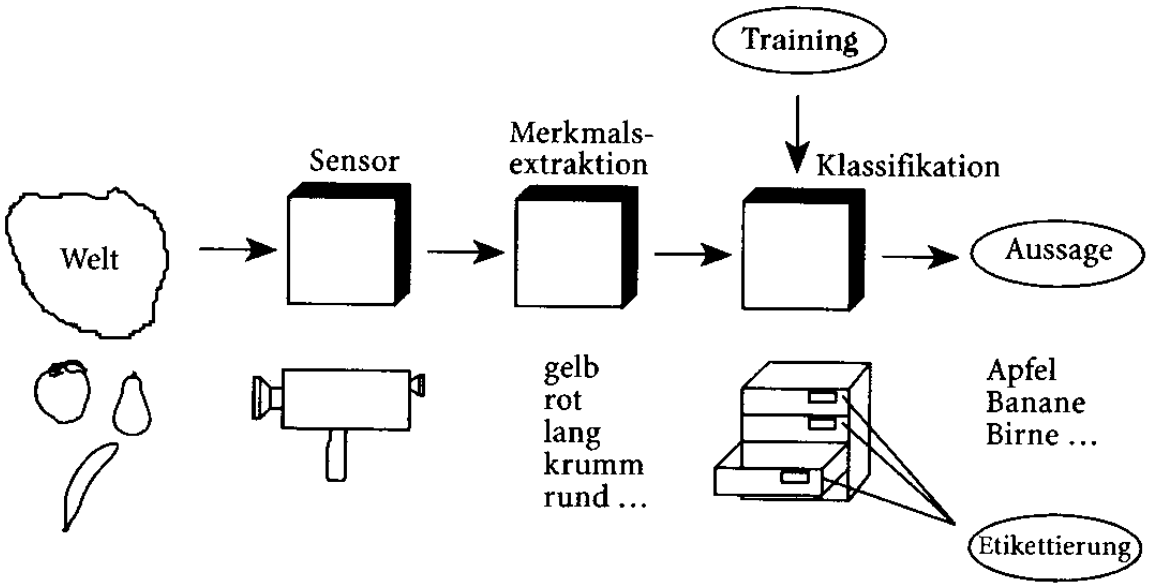
Beachte:
Clusterbildungs- (Clustering-) Verfahren bilden Klassen, Klassifikationsverfahren ordnen Objekte in vorgegebene Klassen ein.

andere Begriffe für Klassifikation:
in der Statistik: *Diskriminanzanalyse*
in der Bildverarbeitung / KI: *Mustererkennung* (pattern recognition) (ungenau, da Erkennung ≠ Klassifikation)

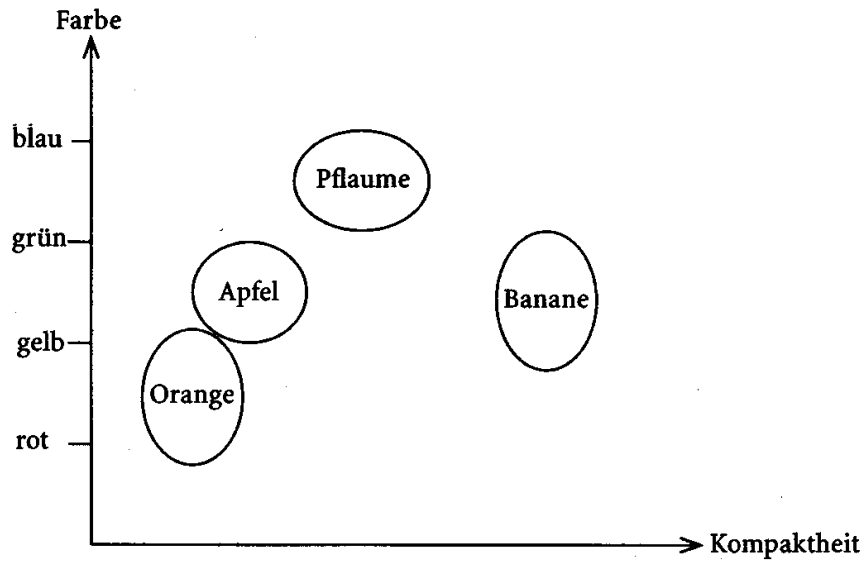
Beispiel (aus Bässmann & Kreyss 1998):

Identifikation von Obstsorten

Merkmale: Farbe, Formfaktor (Kompaktheit)



vorgegebene Klassen:



Klassifikation nach kleinster Distanz (*Minimum-Distance-Klassifikator*)
Außenbereich: *Rückweisungsklasse*

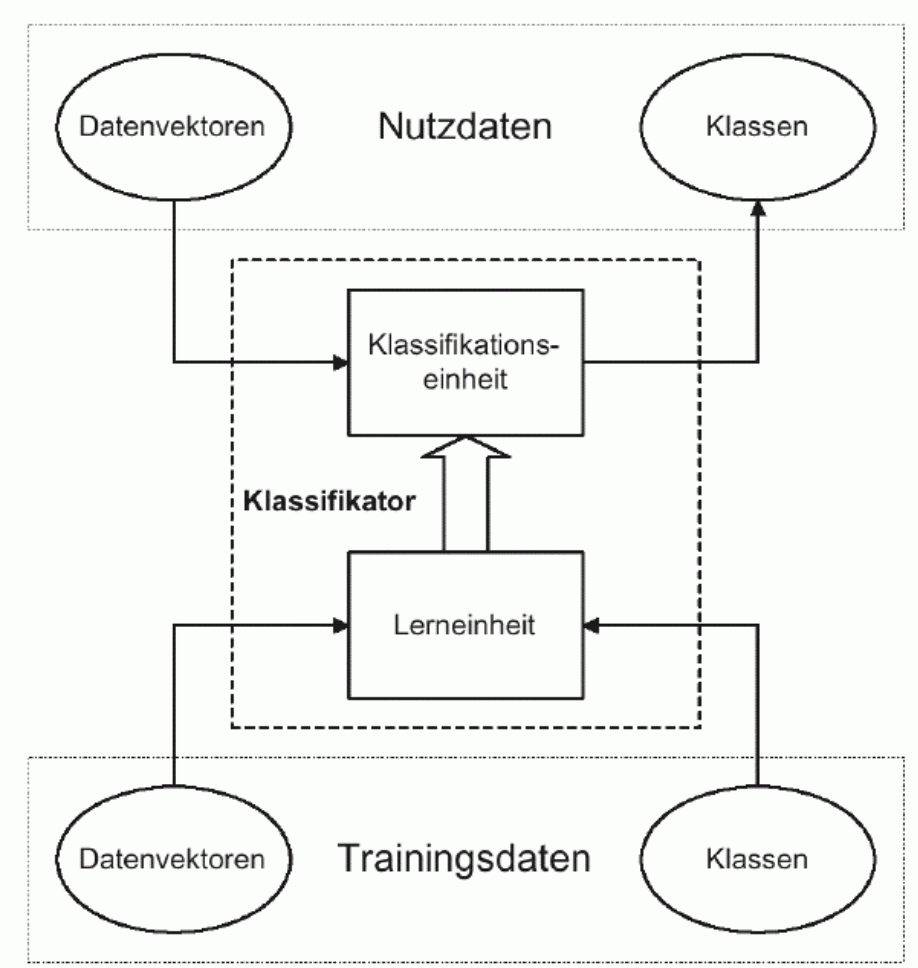
Probleme:

- es könnten zuviele Objekte zurückgewiesen werden
- bei Vergrößerung einzelner Klassen können Überlappungen entstehen $\Rightarrow$ Mehrdeutigkeiten bei der Zuordnung

Typischer Ablauf eines "überwachten" Clusterbildungs- und Klassifikationsprozesses:

1. *Lernphase* (Erstellung eines Klassifikators):
Aus der Datenbasis werden Objekte (zufällig) ausgewählt und zu einer Trainingsmenge (*training data set*) zusammengestellt. Zu jedem Trainingsobjekt wird in einem zusätzlichen Attribut die Klasse festgelegt, zu der es gehört (überwachtes Lernen, *supervised learning*). Anhand der klassifizierten Trainingsdaten wird mittels eines Algorithmus ein Modell (z.B. ein Satz von Regeln) erstellt, das zu einem Merkmalstupel die zugehörige Klasse angeben kann ("Klassifikator").

2. *Klassifikationsphase* (Anwendung des Klassifikators):
die zu klassifizierenden Objekte werden dem Modell unterworfen. Als Ergebnis wird zu jedem Objekt seine Klasse ausgegeben.



Aufbau eines Klassifikators (nach Beichel 2002)

Unüberwachtes Lernen (eigentliche Clusteranalyse):
die Cluster (Klassen) werden automatisch aus den Daten gebildet (auch: "automatische Klassifikation").

- Extraktion von Strukturen aus den Rohdaten
- auch in anderen Bereichen außerhalb der Bildanalyse wichtig (*Data mining*)
- Reduktion der Informationsmenge
- Clusterbildung wird auch zur Unterstützung anderer Algorithmen in der Bildanalyse eingesetzt, z.B. bei der Konturfindung (Hough-Transformation, Clustering im Akkumulatorraum)

Clusteranalyse

Begriffsdefinition und Voraussetzungen

- Clusterungsproblem:
  - gegegen: Eine Menge $S = O_1,...,O_N$ mit N Objekten bzw. Untersuchungsfällen O_i , mit *p* Schlüsseleigenschaften (Merkmalen)
  - gesucht: Klasseneinteilung der Objekte (bzw. Cluster, Gruppen, Partitionen) in *k* Klassen (i = 1..,k) (unsupervised)

$$A_i \subset S$$

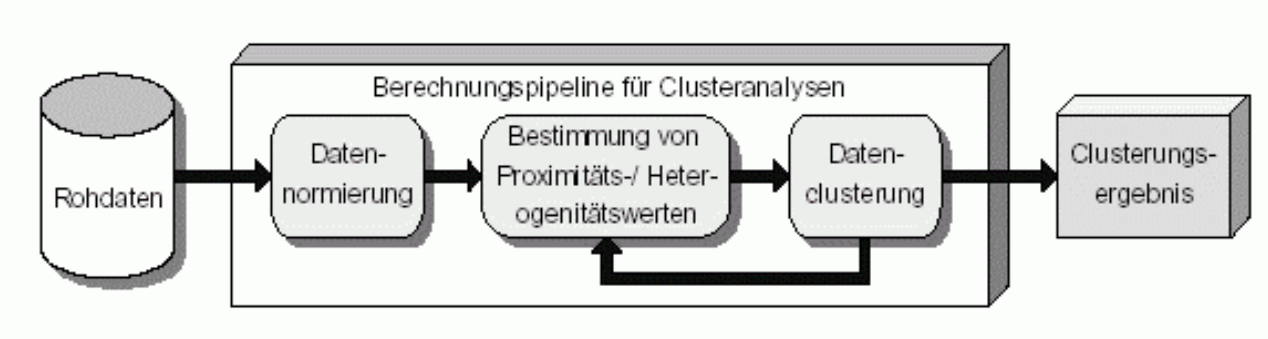
- Ziele:
  - ähnliche Objekte in die gleiche Klasse und unähnliche Objekte in unterschiedliche Klassen
  - Objekte innerhalb einer Klasse möglichst homogen
  - Klassen gut getrennt (große Heterogenität der Klassen untereinander)
- Voraussetzung:
  - Formalisierung und Umsetzung von Maßen für die Begriffe Ähnlichkeit, Distanz, Homogenität und Heterogenität
  - Entwicklung von Clusterverfahren, die die (Ähnlichkeits-, ...) Struktur der Daten adäquat auf Cluster abbilden

Anforderungen an Clustering-Verfahren:

- Endeckung von “natürlichen”, stabilen Clustern
- Entdeckung von Clustern mit beliebiger Form
- Skalierbarkeit: Effektivität und Effizienz auch bei sehr großen Datenmengen
  - große Anzahl von Datenobjekten
  - hohe Dimensionalität
- Stabilität der Verfahren:
  - gegenüber Ausreissern, fehlenden und fehlerbehafteten Daten
  - Unabhängigkeit von der Ordung der Datenwerte
- Geeignete Informationsreduktion, ohne relevante Informationen zu verlieren
  - Über- und Unterklassifikation vermeiden

- Beachtung der spezifischen Besonderheiten der Daten: Integration von Merkmalen verschiedener Art (nominal, ordinal, metrisch...), spezielle Eigenschaften der Merkmale
- Beachtung von Randbedingungen
- Nutzerunterstützung

Berechnungspipeline für Clusteranalysen:



Auswahl der Merkmale:

- zu viele Merkmale können sich negativ auf die Fehlerrate auswirken ("curse of dimensionality")
- Speicherbedarf / Rechenzeit nimmt mit der Anzahl der Merkmale zu!

- Es gibt keine Kochrezepte zur Generierung guter Merkmale
- Auswahl guter Merkmale geht nur über Kombinatorik

$$C(f,n) = \frac{f!}{n!(f-n)!}$$

  - 5 aus 20 - 15.504 Untermengen
  - 5 aus 100 - 75.287.520 Untermengen
- Optimale Lösungen nicht berechenbar, aber brauchbare Lösungen können gefunden werden

- *sequential forward selection* (starte mit leerer Merkmalsmenge, füge in jedem Schritt neues Merkmal hinzu, Auswahl aufgrund von Separabilitätskriterium)
- *sequential backward selection* (starte mit der Gesamtmenge der Merkmale, entferne in jedem Schritt eines)
- Plus-*l*-take-away-*r*-Verfahren (*l* beste Merkmale hinzufügen, *r* schlechteste wieder entfernen, dieses iterieren)

Preprocessing:

- Datenbereinigung (Behandlung von Fehlwerten (missing), von verrauschten Daten, Ausreißerdetektion, Behandlung inkonsistenter Daten)
- Datenintegration (z.B. Erkennung und Elimination von Redundanzen; Normierungen)
- Datenreduktion (Komprimierung von Wertebereichen - Diskretisierung, Merkmalselimination)

Datennormierung für die Clusterung:

Zweck: Merkmale vergleichbar machen (gleicher Wertebereich).

Hauptsächlich 2 Methoden

Intervall-0-1-Normierung:

$$x_i(normiert) = \frac{x_i - \min(x_j)}{\max(x_j) - \min(x_j)}$$

Mittelwert 0 - Varianz 1 - Normierung (Z-Score-Normierung):

$$x_i(normiert) = \frac{x_i - m}{s},$$

wobei *m* das arithm. Mittel und *s* die Standardabweichung des Merkmals ist – das normalisierte Merkmal hat dann Mittelwert 0 und Varianz 1.

Intervall-0-1-Normierung ist der Mittelwert0-Varianz1-Normierung in vielen Fällen überlegen (Jiang & Bunke 1997).

nächster Schritt: Berechnung der Distanzen bzw. Ähnlichkeiten der Objekte

- Aufstellen einer Matrix, welche die Ähnlichkeiten bzw. Distanzen zwischen den Objekten aufgrund ihrer Merkmalsvektoren enthält

	O1	O2	O3	O4	...
O1	0.0	*d(1,2)*	*d(1,3)*	*d(1,4)*	
O2		0.0	*d(2,3)*	*d(2,4)*	
O3			0.0	*d(3,4)*	
O4				0.0	
...					

- Anforderungen an ein Ähnlichkeitsmaß s:
  - für alle k, j mit 1 <= k, j<=N:
    - $0 <= s_{kj} <= 1$
    - $s_{kj} = s_{jk}$ (Symmetrie)
    - $s_{kk} = 1$
- Die Distanz (Unähnlichkeit) d kann aus s z.B. durch $d_{jk} := 1 - s_{jk}$ bestimmt werden

- *d* muss nicht immer eine Metrik im math. Sinne sein (oft verzichtet man auf die Dreiecksungleichung: "*Pseudometrik*")

Beispiele für Ähnlichkeitsmaße:

- Ähnlichkeitsmaße für binäre Daten:
  - M-Koeffizient: $$s_{jk} = \frac{Anzahl(0=0) + Anzahl(1=1)}{Anzahl(Merkmale)}$$
  - S-Koeffizient: $$s_{jk} = \frac{Anzahl(0=0)}{Anzahl(Merkmale) - Anzahl(1=1)}$$
- Ähnlichkeitsmaße für nominale Daten:
  - Verallgemeinerter M-Koeffizient: $$s_{jk} = \frac{Anzahl(gleiche\_Merkmale)}{Anzahl(Merkmale)}$$

- Distanzmaße für quantitative Daten:
  - L_r- Distanzen: $d_{jk}^{(r)} = \left(\sum_{i=1}^{p} | x_{ki} - x_{ji} |^r\right)^{1/r}$
  - Manhattan-Distanz (r=1): $d_{jk} := \sum_{i=1}^{p} | x_{ki} - x_{ji} |$
  - Euklidischer Abstand (r=2): $d_{jk} := \| x_k - x_j \| := \sqrt{\sum_{i=1}^{p} (x_{ki} - x_{ji})^2}$

Maße für die Homogenität einer Klasse:

- z.B. mittleres Ähnlichkeitsmaß für alle Paarungen innerhalb der Klasse
- mittlerer Abstand von einem Repräsentanten (oder Schwerpunkt; Zentroid) der Klasse

- Maße für die Güte der gesamten Clusterung
  - z.B. Varianzkriterium: $H(S) = \sum_{i=1}^{k} \sum_{j \in A_i} (x_j - \overline{x}_{A_i})^2 \rightarrow \min$

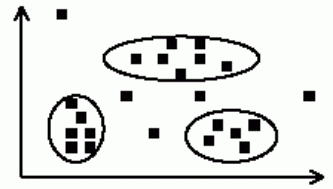
x_j : Merkmalsvektor des Objekts O_j

$\overline{x}_{A_i}$: Arithmetisches Mittel aller zu A_i gehörigen Merkmalsvektoren

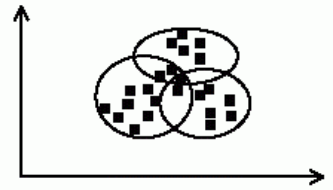
  - Summe der Gütemaße der einzelnen Klassen

Einteilung der Clustering-Algorithmen:

- Nach Art des Ergebnisses:
  - disjunktiv / nicht-disjunktiv
  - hierarchisch / nicht-hierarchisch
  - exhaustiv / nicht-exhaustiv
  - fuzzy / nicht fuzzy
  - konkret / visuell
- Nach dem Schwierigkeitsgrad (für disjunktive Clusterungen):
  - Klassenzahl k vorgegeben, exhaustiv
  - k unbekannt, exhaustiv
  - k unbekannt, nicht-exhaustiv



Bsp.1: disjunkte nicht-exhaustive Gruppierung S mit 3 Clustern



Bsp.2: nicht-disjunktive Gruppierung S mit 3 Clustern

- Nach der Klassenform:
  - runde Form der Cluster
  - Klassentrennung via Hyperebenen des Merkmalsraumes
  - beliebige Klassenformen sind möglich (z.B. über Ketten oder zusammengehörige Punktdichten)
- Nach dem zugrundeliegenden Ansatz (Versuch):
  - ähnlichkeits-/distanzbasierte Ansätze
  - merkmalsvektorbasierte Verfahren
    - Punktdichtemodelle
    - Unterteilung des Merkmalsraumes (Grid-based-Methods)
  - statistische bzw. entscheidungstheoretische Modelle
  - deterministische Modelle:
    - optimale Klassifikation mittels Gütekriterium
    - heuristische Verfahren
    - axiomatische (graphentheorethische) Gruppierung
  - neuronale Netze
  - Abbildung auf 2D-, 3D- Räume -> visuelle Clusterung

Deterministische Modelle / heuristische Verfahren:

wichtigstes Beispiel:

Minimum-Distanz-Verfahren (auch: *k-means clustering*)
(geometrisches Verfahren)

- Eingabe: Clusterzahl k und n Objekte mit den Merkmalsvektoren x_j
- Ausgabe: k Cluster (exhaustiv), die das Varianzkriterium minimieren
- Algorithmus:

1. Zufällige Auswahl von k Objekten als Anfangs-Clusterzentren

Wiederhole

{

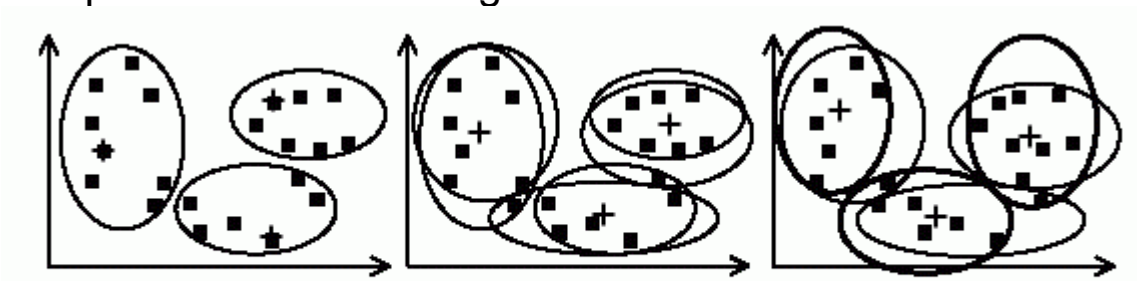
2. Ordne jedes Objekt dem Cluster zu, zu dem das Objekt am ähnlichsten ist (basierend auf dem aktuellen Clusterzentrum)

3. Aktualisiere die Clusterzentren (Mittelwert der zugehörigen Merkmalsvektoren)

} bis sich die Zielfunktion nicht mehr ändert.

→ siehe Übung, Aufgabe U23

Beispiel und Einschätzung:

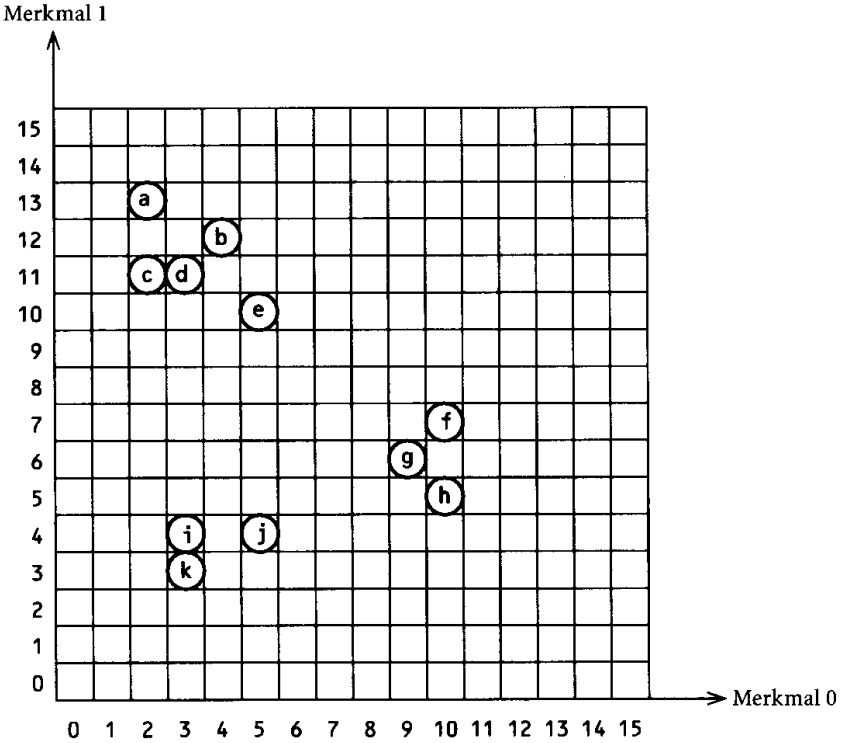


- Vorteile:
  - Umsortierung bereits einsortierter Objekte
  - arbeitet gut auf relativ gut separierten Klassen
- Probleme:
  - Klassenzahl muß bekannt sein
  - Funktioniert nicht auf qualitativen Daten
  - Sensitivität gegenüber Ausreissern
  - mittlere Geschwindigkeit
  - kann in schlechtem lokalem Minimum steckenbleiben

Variante mit festen Clusterzentren (anfangs gewählte Repräsentanten bleiben Clusterzentrum):

Beispiel vgl. "Äpfel und Birnen"-Beispiel, siehe oben

Merkmalsraum:



- Distanzmaß: "City-Block-Distanz" (Manhattan-Metrik) bzw. L_1-Distanz (Summe der Beträge der Koordinatendifferenzen)
- Zurückweisungsschwelle: 6 (Objekte mit größerer Distanz zu allen Cluster-Repräsentanten werden zu Repräsentanten neuer Cluster)
- Systematischer, zeilenweiser Durchlauf der Objekte
- Klassen werden während des Verfahrens gebildet und aufgefüllt

Annahme

- City block distance: $d(x, y) = |x_0 - x_1| + |y_0 - y_1|$
- Zurückweisungsschwelle $d_{max} = 6$

Ablauf der Klassifikation

Init → $z = \{a\}$

$d(a, b) = 3$ → $k_0 = \{a, b\}$

$d(a, c) = 2$ → $k_0 = \{a, b, c\}$

$d(a, d) = 3$ → $k_0 = \{a, b, c, d\}$

$d(a, e) = 6$ → $k_0 = \{a, b, c, d, e\}$

$d(a, f) = 14$ → $z = \{a, f\}$

$d(a, g) = 14$, $d(f, g) = 2$ → $k_1 = \{f, g\}$

$d(a, h) = 16$, $d(f, h) = 2$ → $k_1 = \{f, g, h\}$

$d(a, i) = 10$, $d(f, i) = 10$ → $z = \{a, f, i\}$

$d(a, j) = 12$, $d(f, j) = 8$, $d(i, j) = 2$ → $k_2 = \{i, j\}$

$d(a, k) = 11$, $d(f, k) = 11$, $d(i, k) = 1$ → $k_2 = \{i, j, k\}$

Ablauf der Clusterbildung im Beispiel (unüberwachtes Clustering; aus Bässmann & Kreyss 1998)

Verbesserungen des Minimum-Distanz-Clustering:

- über hierarchische Verfahren (s.u.) zuerst die Anzahl der Zielcluster bestimmen
- gewichtete Zugehörigkeit der Objekte zu verschiedenen Clustern
- Objekte als Klassenrepräsentanten können während des Verfahrens ausgetauscht werden ("k-Medoid-Clustering")
- für große Datenmengen: Clustering nur für "repräsentative" Teilmenge durchführen

Nichtdisjunktive Clusterung

- Vorteil: Gruppierung kann graduelle Abstufungen mit fließendem Übergang erfassen
- Maximale Cliquen
  - graphentheoretisches Verfahren mit Objekten O_i einer gewissen Umgebung d: $d_{ij} \leq d \quad \forall O_i, O_j \in A$
  - Probleme:
    - schlechte Separation
    - viele ähnliche Klassen sind möglich
    - Beschränkung auf einen festen Durchmesser d oft nicht ausreichend
    - Anfälligkeit für Fehler
  - Verbesserungen:
    - mehrere d-Werte (hierarchisches Verfahren)
    - harte Gruppenbedingungen aufweichen: mittlere Ähnlichkeiten (R-Gruppen)
    - zusätzlich zur Homogenität innerhalb der Gruppe wird Heterogenität zu Nichtgruppenmitgliedern gefordert (GR-Gruppen)

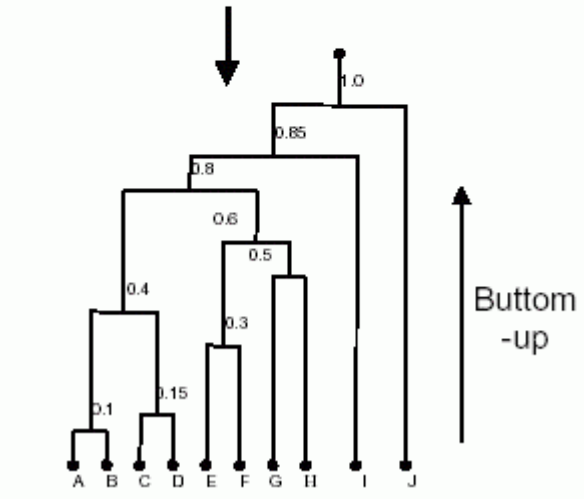
Hierarchische Methoden

- Herangehensweise:
  - keine einzige disjunkte Gruppierung bestimmen, sondern eine ganze Folge solcher Clusterungen mit steigender/sinkender Anforderung an die Homogenität (und steigender/sinkende Klassenzahl) erzeugen
- Forderung:
  - Gruppen sollen vergleichbar sein -> Prinzip der schrittweisen Verfeinerung (größere Klassen in Unterklassen aufteilen) bzw. der Agglomeration (kleinere Klassen zu größeren generalisieren)
- Verfahren:
  - Agglomerative Verfahren (buttom-up)
  - Divisive Verfahren (top-down)
  - Methoden der disjunkten bzw. nichtdisjunkten Klassifikation mit sich änderndem Parameter Homogenität

Hierarchisch agglomerative Clusterung

- 1. Anfangsklassifikation: ein-elementige Cluster bestehend aus den Objekten (hier: A, B, ...)
- 2. Zusammenfassung von Objekten/Clustern bei Minimierung eines "Cluster-Kriteriums" basierend auf der Distanzmatrix
- 3. Berechnung eines indizierten Dendrogramms: Zusammenfassung von Clustern auf verschiedenen Heterogenitätsebenen
- 4. Fahre fort bis alle Objekte zu einer einzigen Klasse vereinigt sind

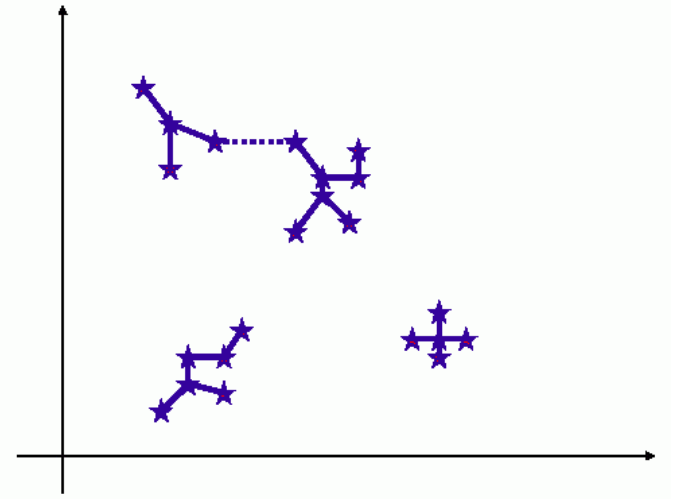
	A	B	C	D	. . .
A	0,0	*d(1,2)*	*d(1,3)*	*d(1,4)*	
B		0,0	*d(2,3)*	*d(2,4)*	
C			0,0	*d(3,4)*	
D				0,0	
. . .					

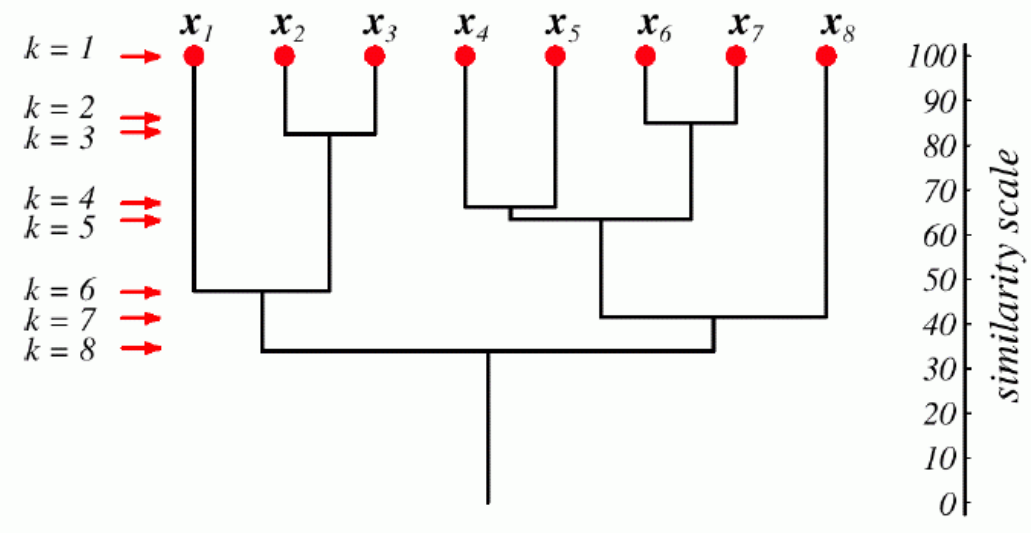


- Clusterkriterien - Resultierende Verfahren:
  - Single-Linkage: $D_{A_r A_s} := \min_{\forall j \in A_r, k \in A_s} d_{jk}$
    - neigt zur Kettenbildung, Identifikation von Ausreissern ist möglich, wenige große Klassen
  - Complete-Linkage: $D_{A_r A_s} := \max_{\forall j \in A_r, k \in A_s} d_{jk}$
    - neigt zur Bildung kleiner Gruppen und versucht gleich starke Gruppen zu bilden
  - Average-Linkage: $D_{A_r A_s} := \frac{1}{|A_r||A_s|} \sum_{j \in A_r} \sum_{k \in A_s} d_{jk}$
    - konservativ

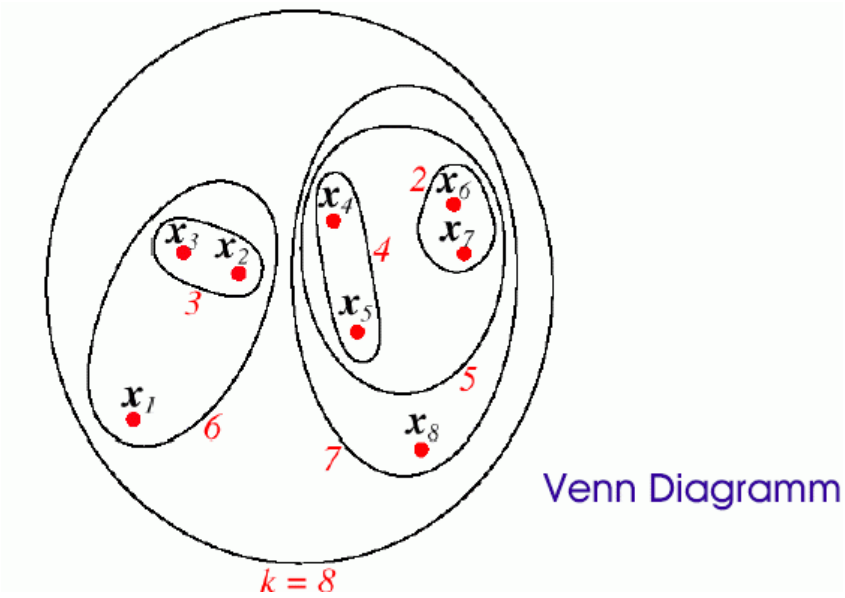
Beispiele für hierarchisches Clustern:

anhand der euklidischen Distanz



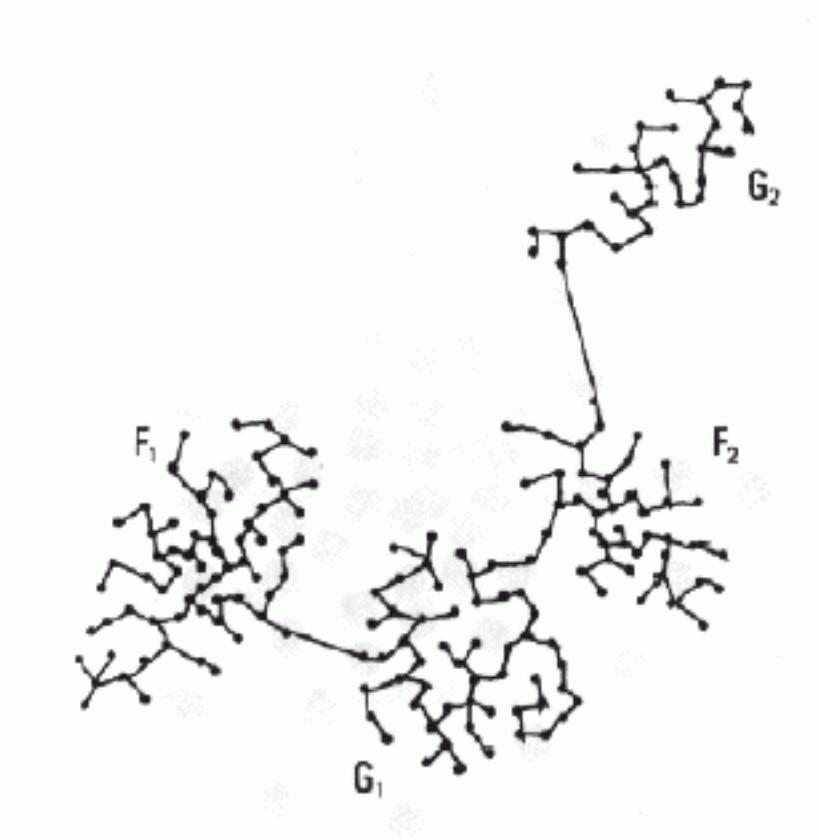


Dendrogramm



Graphentheoretische Methode für hierarchisches Clustern:

- Aufbau eines minimalen aufspannenden Baumes (MST) im Merkmalsraum (vgl. Übung, Aufg. U21)
- sukzessives Entfernen der jeweils längsten Kanten im MST erzeugt hierarchische Clusterung



(aus Beichel 2002)

Hierarchische Methoden
Vor- und Nachteile

- Vorteile von hierarchischer Clusterung
  - Unabhängigkeit von der Anzahl der Klassen
  - Hierarchie erlaubt mehr Aussagen über die Struktur der Datenmenge als nicht-hierarchische Clusterungen
- Nachteile:
  - durchgeführte Agglomerations- oder Teilungsentscheidung kann nicht wieder zugunsten einer besseren Einteilung rückgängig gemacht werden
  - Dendrogramm bedeutet kaum Informationsreduktion -> sehr große, mglw. unübersichtliche Bäume

Dichtebasierte Verfahren

es werden Bereiche im Merkmalsraum ermittelt, die besonders dicht von Objekten belegt sind

- jedes Objekt in einem Cluster besitzt in seiner Umgebung entweder (a) eine festgelegte Mindestzahl von anderen Objekten oder (b) zumindest ein anderes Objekt, das zu diesem Cluster gehört – für das also (a) oder (b) erfüllt ist.
- Objekte, die zu keinem Cluster gehören, weil sie in zu dünn besiedelten Bereichen liegen, werden als Ausreißer angesehen

Gitter-Verfahren

- unterteilen den Merkmalsraum gitterartig und führen das Clustering nur für die Gitterzellen aus
- für hochdimensionale metrische Merkmalsräume
- Vorteil: Verarbeitungsgeschwindigkeit

Fuzzy clustering

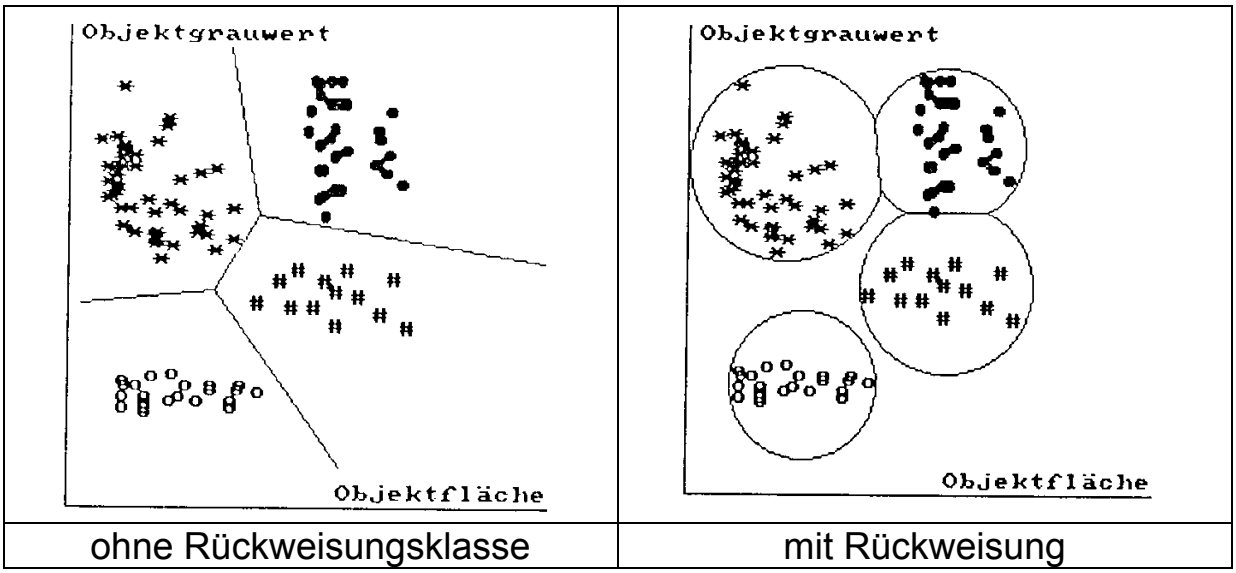
- beruht auf Fuzzy sets – Verallgemeinerung des Mengen-begriffs, *fuzzy membership function* drückt "Grad der Zugehörigkeit" zu einer Fuzzy-Menge aus
- Formalisierung der Unsicherheit der Klassenzuordnung
- liefert oft bessere Ergebnisse als klassische Verfahren
- siehe Beichel 2002

Klassifikationsverfahren

Minimum-Distanz-Verfahren:
aufbauend auf Minimum-Distanz-Clustering (siehe oben)

auch in überwachter Version:

- Ermittlung der Cluster in Trainingsphase
- jedes Cluster wird durch Repräsentanten oder Schwerpunkt vertreten
- Zuordnung eines Objekts anhand der minimalen Distanz zum Repräsentanten



ohne Rückweisungsklasse	mit Rückweisung

Beispiel: Mittelwerte und Varianzen der Cluster, wenn die Objekte a-k aus obigem "Obst-Beispiel" als Trainingsvektoren benutzt werden (aus Bässmann & Kreyss 1998):

		Mittelwert 0	Mittelwert 1		Varianz 0	Varianz 1
K_0	a	2	13		$(2-3.2)^2$	$(13-11.4)^2$
	b	4	12		$(4-3.2)^2$	$(12-11.4)^2$
	c	2	11		$(2-3.2)^2$	$(11-11.4)^2$
	d	3	11		$(3-3.2)^2$	$(11-11.4)^2$
	e	5	10		$(5-3.2)^2$	$(10-11.4)^2$
		16	57		6.8	5.2
	÷5	3.2	11.4	÷4	1.7	1.3
K_1	f	10	7		$(10-9.7)^2$	$(7-6)^2$
	g	9	6		$(9-9.7)^2$	$(6-6)^2$
	h	10	5		$(10-9.7)^2$	$(5-6)^2$
		29	18		0.67	2
	÷3	9.7	6	÷2	0.335	1
K_2	i	3	4		$(3-3.7)^2$	$(4-3.7)^2$
	j	5	4		$(5-3.7)^2$	$(4-3.7)^2$
	k	3	3		$(3-3.7)^2$	$(3-3.7)^2$
		11	11		2.67	0.67
	÷3	3.7	3.7	÷2	1.335	0.335

als Dispersionsmaß dient das Maximum der Varianzen der beiden Komponenten

Ablauf der Klassifikation:

- Bestimmung der Distanzen des gegebenen Merkmalsvektors zu sämtlichen Clusterzentren
- vorläufige Zuordnung zu dem Cluster, zu dessen Zentrum die geringste Distanz besteht
- endgültige Zuordnung, falls diese Distanz das als Zurückweisungsschwelle dienende Dispersionsmaß des Clusters nicht überschreitet.

Stochastischer Ansatz (Bayes-Klassifikation)

= Entscheidungstheoretisches Modell

- jede Objektklasse wird als (i.allg. multivariate) Zufalls-variable aufgefasst
- Parameter dieser Zufallsvariablen werden aus Stichprobe geschätzt ("Trainingsphase")
- es wird versucht, unter "vernünftigen" stochastischen Annahmen die Wahrscheinlichkeit einer Fehlzuordnung zu minimieren
- d.h. ein Objekt wird derjenigen Klasse zugeordnet, die für seine individuelle Merkmalskombination am wahrscheinlichsten ist.

Grundlage hierfür: *bedingte Wahrscheinlichkeiten*,
Satz von Bayes

Notationen:

- Datenvektoren $X = \left\{ \vec{x}_k \in \Re^o \mid k = 1,2,...n \right\}$
- Klassen $\Omega = \left\{ \omega_i \mid i = 1,2,...c \right\}$
- Anzahl der Klassen $c \in \left\{ j \in \aleph \mid 2 \le j \right\}$
- Klassifikationsprozeß $\Theta : X \to \Omega$

Bedingte Wahrscheinlichkeit:

P(*A* | *B*) = Wahrscheinlichkeit von *A* unter der Bedingung *B*
= W'keit von *A*, wenn *B* schon eingetreten ist

W'keit des gemeinsamen Eintretens von *A* und *B*:
$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$

wenn *A* und *B* stochastisch unabhängig sind, gilt:
$P(A \cap B) = P(A)P(B)$, $P(A|B) = P(A)$, $P(B|A) = P(B)$.

Wenn $P(B)>0$:
$P(A|B) = P(A \cap B) / P(B)$
$P(B)$ heißt "*a-priori-Wahrscheinlichkeit*"

Beispiel z. Rechnen mit bedingten W'keiten (aus Hermes 2002):

Nach einem Picknick vermisst eine Familie ihren Hund. 3 Hypothesen, wo sich der Hund befinden kann:
(1) er ist heimgelaufen (Ereignis A)
(2) er bearbeitet noch den großen Knochen auf dem Picknickplatz (B)
(3) er streunt im Wald herum (C).
Durch Kenntnis der Gewohnheiten des Hundes schätzt man die A-priori-Wahrscheinlichkeiten zu ¼, ½ und ¼ . Ein Kind sucht bei 2, ein Kind bei 3. Ist der Hund bei 2., dann ist es leicht, ihn zu finden (90%). Ist der Hund im Wald, stehen die Chancen bei 50%. Frage: Mit welcher W'keit wird der Hund gefunden (=Ereignis D)?

Gegeben: P(A) = ¼, P(B) = ½, P(C) = ¼.
P(D|A) = 0; P(D|B) = 0,9; P(D|C) = 0,5.

$$P(D) = P(A)\cdot P(D\mid A) + P(B)\cdot P(D\mid B) + P(C)\cdot P(D\mid C)$$
$$= \frac{1}{4}\cdot 0 + \frac{1}{2}\cdot 0{,}9 + \frac{1}{2}\cdot 0{,}5 = \frac{115}{200} \approx 58\%$$

allgemein:

- *a-priori-W'keit:* Die W'keit, dass eine Hypothese zutrifft, bevor irgendein Anhaltspunkt vorliegt
- *bedingte W'keit:* Die W'keit, dass ein bestimmtes Ereignis eintritt, nachdem ein anderes bereits eingetreten ist.
- *a-posteriori-W'keit:* Die W'keit, dass eine Hypothese zutrifft, nachdem das Eintreten eines bestimmten Ereignisses berücksichtigt worden ist.

am Beispiel der Klassifikation von Fischen nach den Merkmalen Länge und Helligkeit (Beichel 2002):

$P(\omega_1)$, $P(\omega_2)$ … a priori Wahrscheinlichkeiten

sea bass salmon

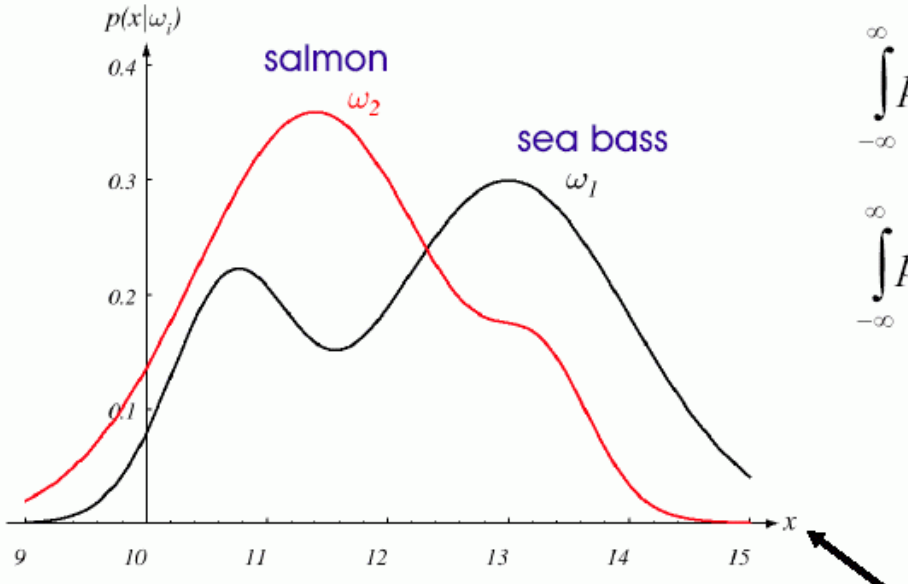
$P(\omega_1)+P(\omega_2)=1$ (Annahme: Es treten nur die Klassen ω_1 und ω_2 auf!)

- Sehr einfache Klassifikationsregel:

$$S=\begin{cases}\omega_1 & falls \quad P(\omega_1)>P(\omega_2)\\ \omega_2 & sonst\end{cases}$$

Entscheidung ohne Features!

Dichtefunktionen der bedingten W'keiten der Klassen (Dichtefunktionen der Merkmalsverteilungen, empirisch als Grenzfall relativer Häufigkeiten):
Bedingung = Merkmal (Feature)



$$\int_{-\infty}^{\infty} p(x|\omega_1)dx=1$$

$$\int_{-\infty}^{\infty} p(x|\omega_2)dx=1$$

Der Satz von Bayes

- Bayes Theorem:

$$P(\omega_j|\vec{x}) = \frac{p(\vec{x}|\omega_j)P(\omega_j)}{p(\vec{x})} \quad \text{mit} \quad p(\vec{x}) = \sum_{j=1}^{c} p(\vec{x}|\omega_j)P(\omega_j)$$

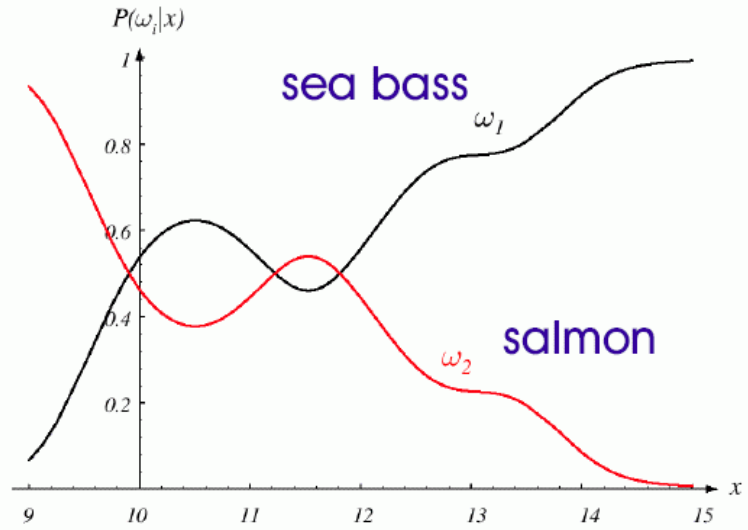
Klassifikation:

$$S = \begin{cases} \omega_1 & falls \quad P(\omega_1|\vec{x}) > P(\omega_2|\vec{x}) \\ \omega_2 & sonst \end{cases}$$

Error Probability:

$$P(error|\vec{x}) = \begin{cases} P(\omega_1|\vec{x}) & falls \quad S = \omega_2 \\ P(\omega_2|\vec{x}) & falls \quad S = \omega_1 \end{cases}$$

damit lassen sich die a-posteriori-Wahrscheinlichkeitsdichten bestimmen:



$$P(\omega_1|\vec{x}) + P(\omega_2|\vec{x}) = 1$$

- Vorteile: Bayes-Klassifikation = schnelles Verfahren, hohe Genauigkeit bei großen Datenmengen
- Problem: zu viele der Wahrscheinlichkeiten in der Bayes'schen Formel sind i.allg. unbekannt und dann wird das Verfahren ungenau

- Praxis: Bayes Klassifikator nicht anwendbar
- Einschränkungen müssen getroffen werden
- Annahme: alle $p(\vec{x}\,|\,\omega_i)$ entsprechen einem bestimmten Modell

→ Abschätzung der Modellparameter

Modellverteilung: die *Normalverteilung* (Gauß-Verteilung)

$$f(x) = \frac{1}{\sigma \cdot \sqrt{2\pi}} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Parameter: μ (= Erwartungswert, Maximalstelle der Glockenkurve), σ (= Standardabweichung; zwischen $\mu - \sigma$ und $\mu + \sigma$ liegen ca. 68 % der Fläche, zwischen μ - 2σ und $\mu + 2\sigma$ ca. 95 %).

jedoch wird hier die mehrdimensionale (multivariate) Form benötigt:

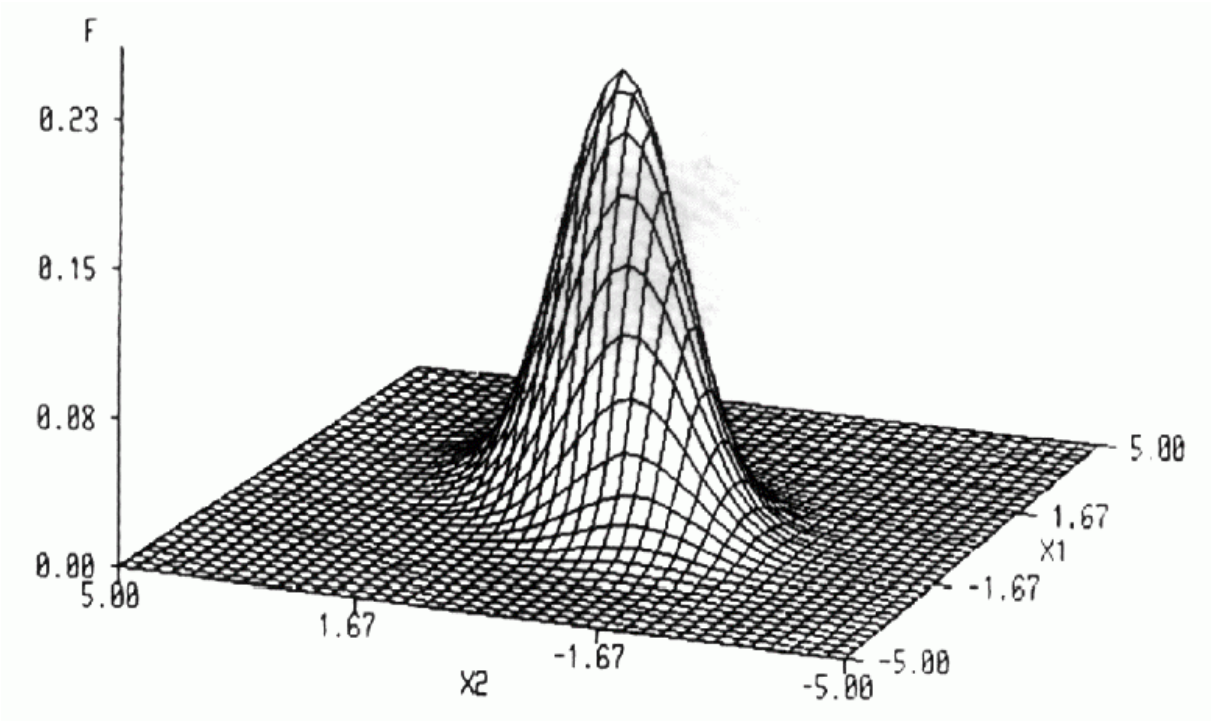
Beispiel: zweidimensionale Normalverteilung

Wahrscheinlichkeitsdichte:

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \cdot \cdot e^{\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_x)^2}{\sigma_x^2} - 2\rho\frac{(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y} + \frac{(y-\mu_y)^2}{\sigma_y^2}\right)}$$

Parameter: μ_x, μ_y, σ_x, σ_y, ρ .

ρ entspricht dem Korrelationskoeffizienten der beiden Merkmale. Für $\rho = 0$ hat man stochastische Unabhängigkeit, und es ergibt sich $f(x, y) = f(x) \cdot f(y)$ mit den Wahrscheinlichkeitsdichten $f(x)$, $f(y)$ der eindimensionalen Normalverteilung.



bivariate Normalverteilung

allgemeiner Fall: multivariate Normalverteilung
mit gegebenen Kovarianzen der Einzelmerkmale untereinander

$p(\vec{x} \mid \omega_i)$ wird meistens als Verteilung $N_l(\mu, \Sigma)$ angenommen

$$f(x_1, x_2, ..., x_l) = \frac{1}{(2\pi)^{\frac{l}{2}} \mid \Sigma \mid^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu})\}$$

μ	Mittelwertsvektor
Σ	Kovarianzmatrix (nicht singulär)
$\lvert\Sigma\rvert$	Determinante der Kovarianzmatrix

Kovarianzmatrix: drückt die lin. Zusammenhänge zwischen den Merkmalen aus; eigentlich Varianz-Kovarianzmatrix, da in der Diagonale die Varianzen stehen.

- auf [–1, 1] normierte Kovarianz: Korrelation

$$\Sigma = E[\ (\mathbf{x}- \mu)(\mathbf{x}- \mu)^T\ ]$$

$\mu = E[\mathbf{x}]$ … Mittelwertsvektor

Σ wird elementweise berechnet:

$\Sigma = \{\sigma_{ij}\}$ $\quad \sigma_{ij} = E[\ (x_i- \mu_i)(x_j- \mu_j)\ ]$

… quadratische Matrix, Größe = I (# Merkmale)

$\sigma_{ii} = \sigma_i$ … Varianz des Merkmals i

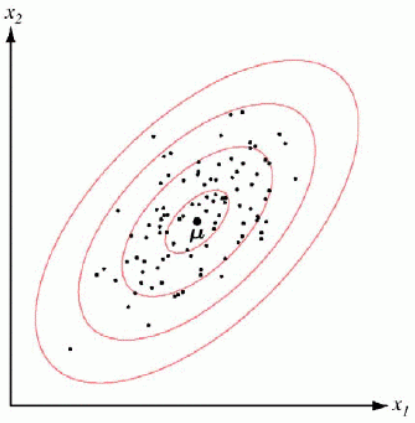
σ_{ij} … Kovarianz zwischen Merkmal i und j
(je kleiner, desto weniger Zusammenhang zwischen i und j)

2-dim. Fall:

Konturen konst. Dichte sind Ellipsen um den Mittelwert

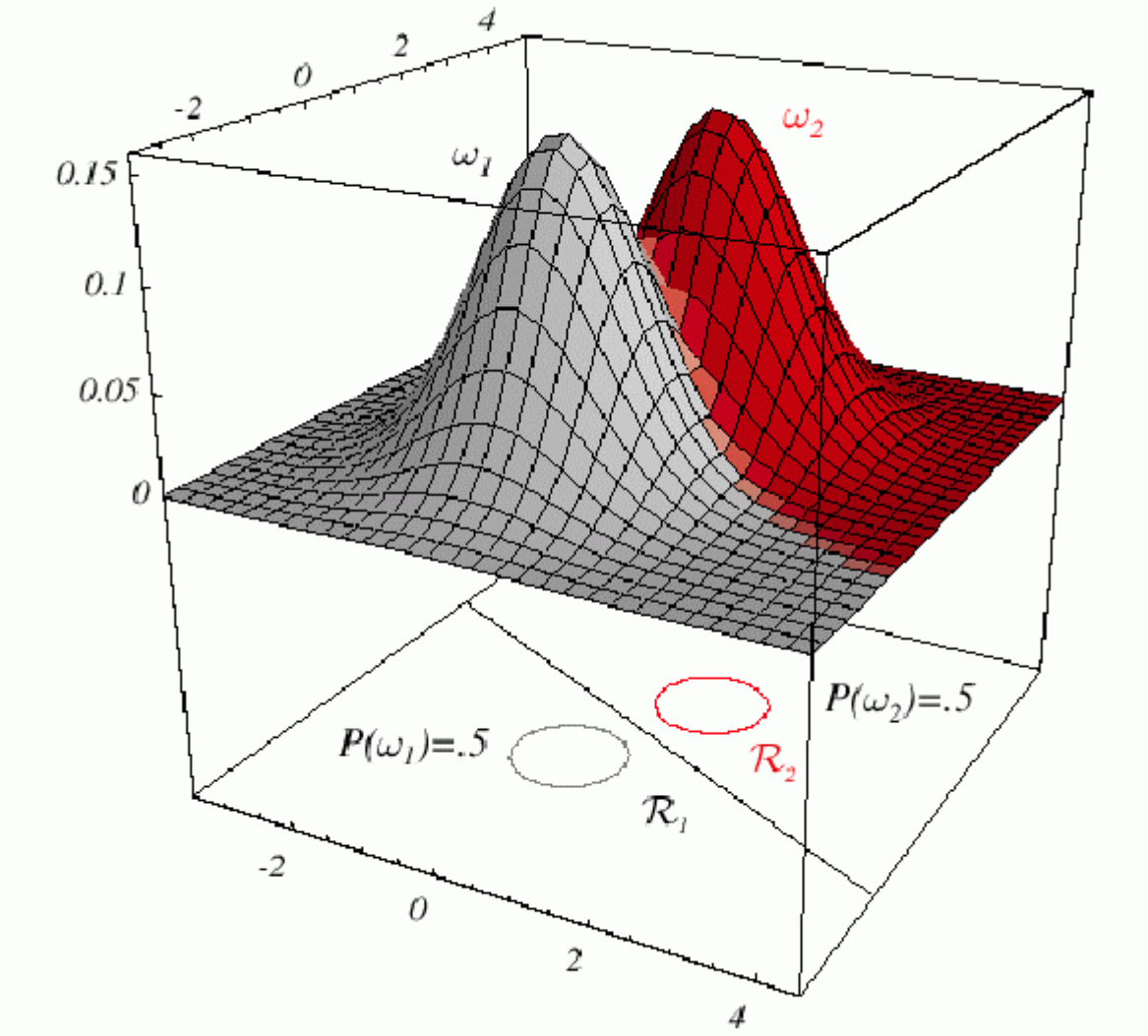
$$(\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu}) = c^2$$

Zentrum bei $(\mu_1 \mu_2)$:



Einsatz in der Klassifikation:
die Zuordnung erfolgt zu derjenigen Klasse, für die $P(\omega_i \mid x)$ am größten ist (wobei sich dieses nun ausrechnen lässt)

= *Maximum-Likelihood-Klassifikator*



- Die Abschätzung der unbekannten Parameter $\vec{\mu}_i$ und Σ_i erfolgt aus den Lerndaten

$$\vec{\mu}_i^* = \frac{1}{n_i}\sum_{j=1}^{n_i}\vec{x}_j$$

$$\Sigma_i^* = \frac{1}{n_i - 1}\sum_{j=1}^{n_i}\left(\vec{x}_j - \vec{\mu}_i^*\right)\left(\vec{x}_j - \vec{\mu}_i^*\right)^T$$

Formel für die bedingte W'keit unter den getroffenen Annahmen:

$$p(\vec{x} \mid \omega_i) = \frac{1}{(2\pi)^{\frac{l}{2}} |\Sigma_i|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\vec{x} - \vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x} - \vec{\mu}_i) \right]$$

In vielen Fällen ist nur der Exponent interessant (nur dieser enthält den Merkmalsvektor)
$\Rightarrow$ man betrachtet als Entscheidungskriterium die Größe von

$$(\vec{x} - \vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x} - \vec{\mu}_i)$$

= *"Mahalanobis-Distanz"* von *x* und μ_i

Der *Mahalanobis-Distanz-Klassifikator* bestimmt die kleinste Mahalanobis-Distanz zu den Clusterzentren:

- Mahalanobis-Distanz jeder Klasse
- Wähle Klasse minimaler Distanz, unterhalb Zurückweisungsschwelle
- Min. M.Distanz ->max. Wahrscheinlichkeit

Isolinien der Mahalanobis-Distanz:

- Mehrdimensionale Hyperellipsoide im Merkmalsraum

Wenn man zusätzlich noch die Annahme trifft, dass die Kovarianzmatrix ein Vielfaches der Einheitsmatrix ist (stochastische Unabhängigkeit und gleiche Varianzen), so ergibt sich wieder der einfache Minimum-Distanz-Klassifikator: die Mahalanobis-Distanz wird unter diesen Annahmen zu

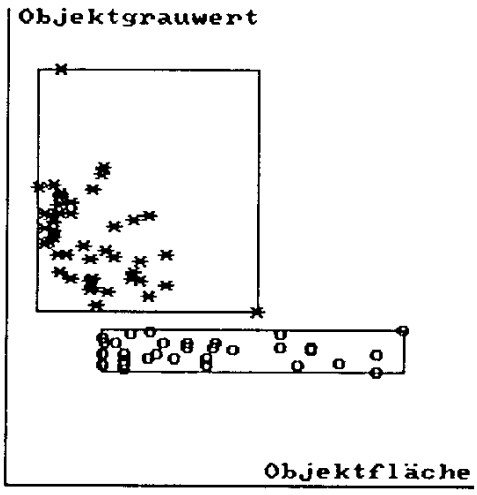
$$\frac{(\vec{x}-\vec{\mu}_i)^T(\vec{x}-\vec{\mu}_i)}{2\sigma^2},$$

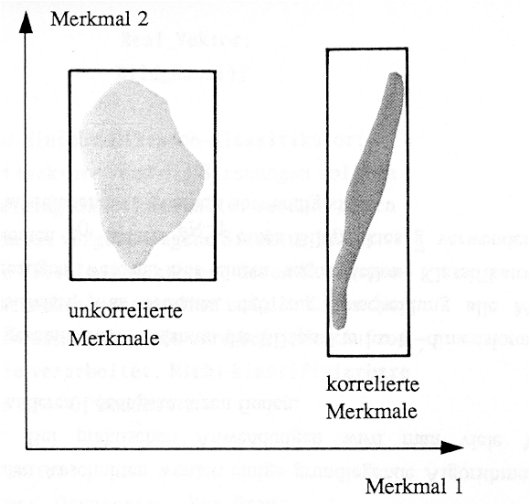
also bis auf konst. Faktor die quadrierte euklidische Distanz.

weitere Klassifikationsverfahren in Kurzübersicht

Quadermethode

- geometrischer Klassifikator
- achsenparalleler Quader wird um die Klasse gelegt
- sehr einfach zu implementieren und rechenzeitsparend





- Nachteil: Mehrdeutigkeit bei Überlappung der Quader
- Abhilfe: in diesem Fall nach einem anderen Verfahren klassieren
- prakt. Erfahrung: weniger als 1/3 der Bildpunkte liegen in Überlappungsbereichen $\Rightarrow$ Quadermethode als Vorstufe bringt immer noch Rechenzeitvorteil

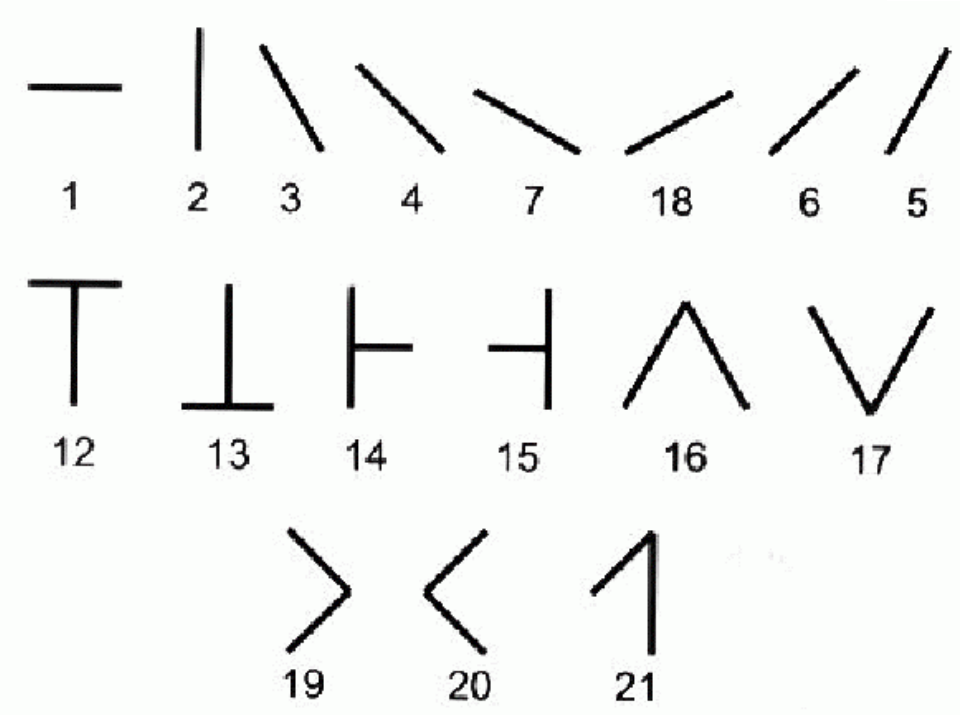
Entscheidungsbaum

- vorab berechnete Kontrollstruktur (Entscheidungs-Kaskade) für die Klassifikation
- anknüpfend an hierarchische Cluster-Verfahren
- oder explizit vom Designer des Systems entworfen bei kleinen, festen Datensätzen

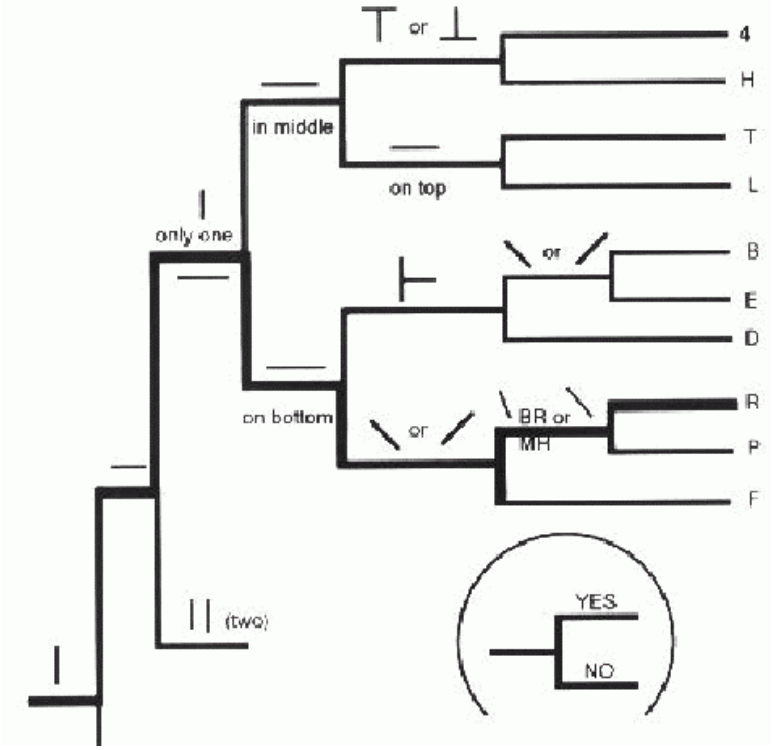
Beispiel

OCR - Optical Character Recongnition

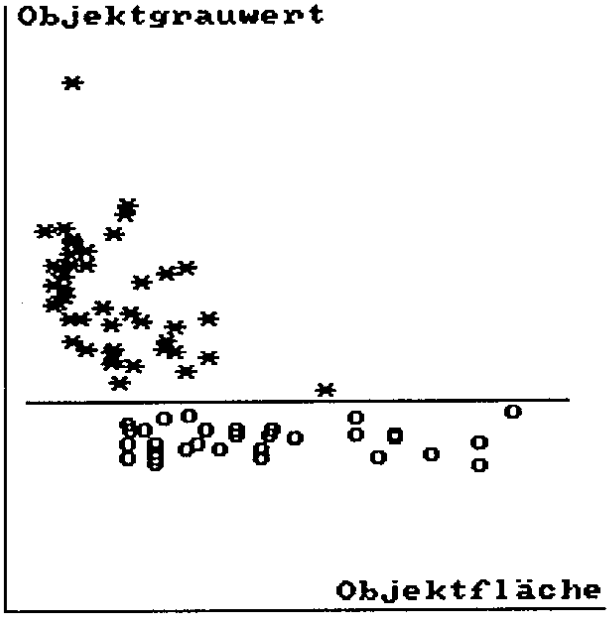
- Zerlegung der Buchstaben in einzelne Striche
- Klassifikation aufgrund des Auftretens der Striche (Entscheidungsbaum)



Entscheidungsbaum für die Schrifterkennung (Ausschnitt)



Lineare Klassifikatoren:

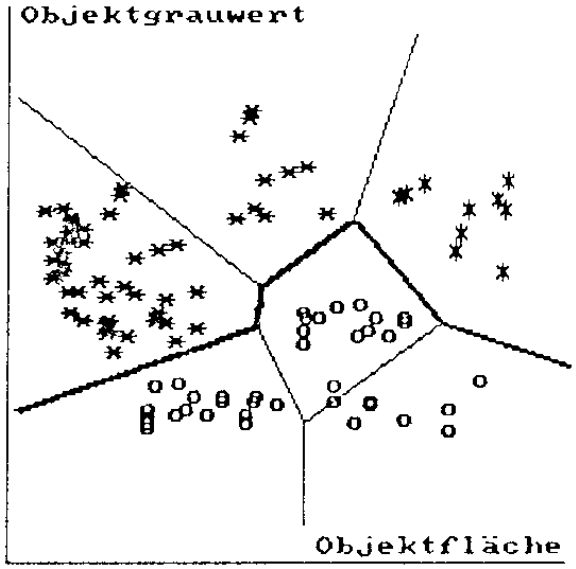


- Teilung des *M*-dim. Merkmalsraumes durch eine (*M*–1)-dimensionale Hyperebene
- einfachste Art der Bisektion

- optimale Anpassung der Hyperebene an die Trainingsdaten durch iterativen Prozess: *Fehlerkorrekturalgorithmus*, *Perzeptron-Algorithmus* (s. Voss & Süße 1991)
- Vorbild für Error Backpropagation bei neuronalen Netzwerken, siehe unten

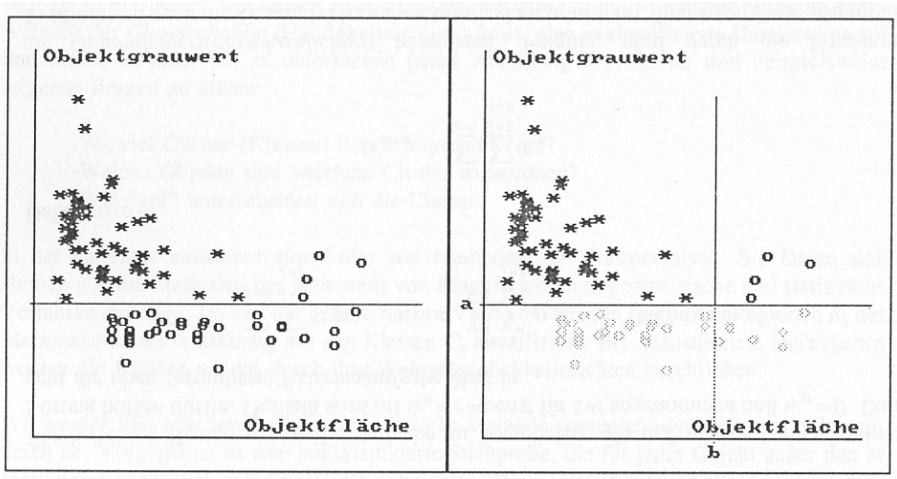
Nächster-Nachbar-Klassifikator

Distanz wird nicht zum Zentrum eines Clusters gebildet, sondern zu allen (bekannten) Elementen (oder zu einer festen Menge ausgewählter Repräsentanten), und davon wird das Minimum für die Entscheidung benutzt:



(aus Voss & Süße 1991)

Hierarchische Klassifikation mit achsenparallelen Hyperebenen:



Qualitäts- (Performance-) Bewertung bei Klassifikations-verfahren

- Grundlage für die Klassifikator-Auswahl und -Verbesserung

Genauigkeitsabschätzung aus den Lerndaten: Fehlklassifikationstabelle

TABLE 8.1. Misclassification Table

		Predicted Class						
		1	2	.	.	.	g	Total
Correct Class	1	n_{11}	n_{12}	.	.	.	n_{1g}	n_1
	2	.	.				.	n_2
	.	.	.				.	.
	.	.	.				.	.
	.	.	.				.	.
	g	n_{g1}	.	.	.	.	n_{gg}	n_g/N

"Holdout"-Verfahren:

- Aufteilung der Referenzdaten in Lerndaten und Testdaten (zufällige Auswahl!)
- Fehlerschätzung aufgrund der Testdaten
- NT: nicht alle Referenzdaten werden zum Training verwendet
- liefert pessimistische Schätzung

Verwendung von *Standard-Objektmengen*, an denen neue Verfahren evaluiert werden (*benchmarking*)

Beispiel:

- OCR: Standardbuchstaben wurden vereinbart - OCR-A, OCR-B
- Fehlerraten auf solchen Dokumenten werden zu Vergleichszwecken herangezogen

OCR-A Testbuchstabensatz:

Postprocessing
(Nachbearbeitung des Klassifikationsergebnisses):

- Entscheidung des Klassifikators nicht fehlerfrei
  - keine Entscheidung (Reject)
  - mehrdeutige Entscheidung
  - Liste von Zugehörigkeits-wahrscheinlichkeiten

B oder 13?

- Postprozessor muß mit dieser Information umgehen können
- z.B. kontextabhängige Entscheidung (Spell-Checker bei OCR)