
16. Wissenserwerb

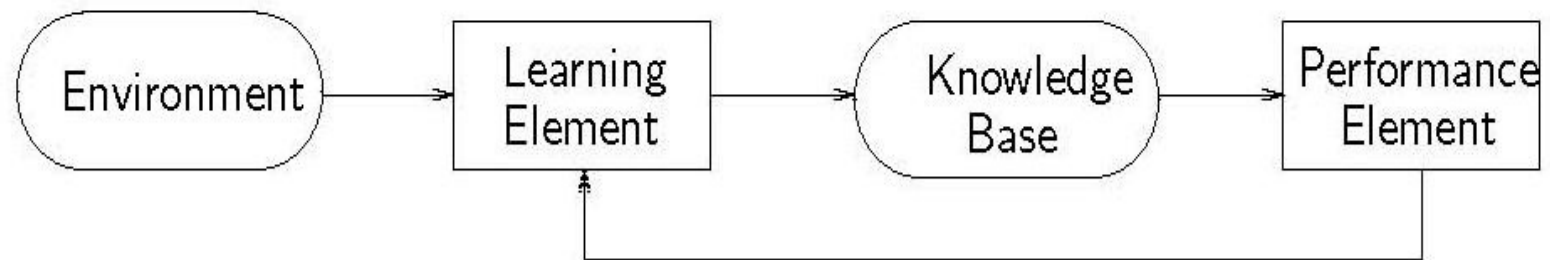
Das Material dieses Kapitels basiert
auf den Arbeiten von Frau Gabriele
Kern-Isberner.

Wissenserwerb

- Lernen bedeutet adaptive Systemveränderungen, die das System in die Lage versetzen, dieselbe Aufgabe beim nächsten Mal effizienter und effektiver zu erfüllen. [Simon 83]
- Lernen bedeutet die Konstruktion oder Modifikation von Repräsentationen unserer Erfahrungen [Michalski 86]
- Die Untersuchung und maschinelle Modellierung von Lernprozessen in ihren vielfältigen Ausprägungen stellen das Hauptthema des maschinellen Lernens dar [Carbonell, Michalski, Mitchell 83]
- Forschung im Bereich des maschinellen Lernens beinhaltet die Implementation von Computer-Programmen, die in der Lage sind, durch Informationsinput neues Wissen aufzubauen oder vorhandenes Wissen zu verbessern. [Michalski, Kodratoff 90].

Wissenserwerb

Schema eines Lernmodells



Wissenserwerb

Lernen

- durch direkte Eingabe von Wissen
- durch Deduktion
- durch Analogieschließen
- aus Beispielen (überwachtes Lernen)
 - ▶ von einer "Quelle" (Lehrer, Umgebung)
 - ▶ positive / negative Beispiele
 - ▶ inkrementell / alle auf einmal
- aus Beobachtungen & Experimenten (unüberwachtes Lernen)
 - ▶ Beobachten der Umgebung (passiv)
 - ▶ Ausführen von Experimenten (aktiv)

Wissenserwerb

- Überwachtes vs. unüberwachtes Lernen
 - ▶ Unüberwachte Klassifikation (Clustering): keine Klassifikation vorher bekannt, wird datengetrieben vorgenommen;
 - ▶ Überwachte Klassifikation (Diskriminanzanalyse): Bereits klassifizierte Beispiele vorgegeben; Aufgabe ist, ein neues Beispiel zu klassifizieren.
- Inkrementelles vs. nicht-inkrementelles Lernen:

Hier wird danach unterschieden, ob die Beispiele/Beobachtungen alle zu Beginn zur Verfügung stehen müssen oder nach und nach verarbeitet werden können.

Wissenserwerb

Typen des gelernten Wissens

- Parameter in algebraischen Ausdrücken
- Entscheidungsbäume
- formale Grammatiken
- Regeln (Assoziationsregeln)
- Konzepte
- Ontologien
- ...

Clustering als unüberwachtes Lernen

Unter Clustering oder Cluster-Analyse versteht man die unüberwachte Organisation einer Sammlung von Beobachtungen (Daten) in Gruppen.

Kriterien für die Aufteilung in Cluster:

- Ähnlichkeit innerhalb eines Clusters (intra-class similarity);
- Unähnlichkeit zwischen (verschiedenen) Clusters (inter-class dissimilarity).

Beobachtungen innerhalb eines Clusters sind zueinander ähnlicher als zu Beobachtungen aus anderen Clustern.

Clustering – Historie und Anwendungen

Clustering hat seine Wurzeln

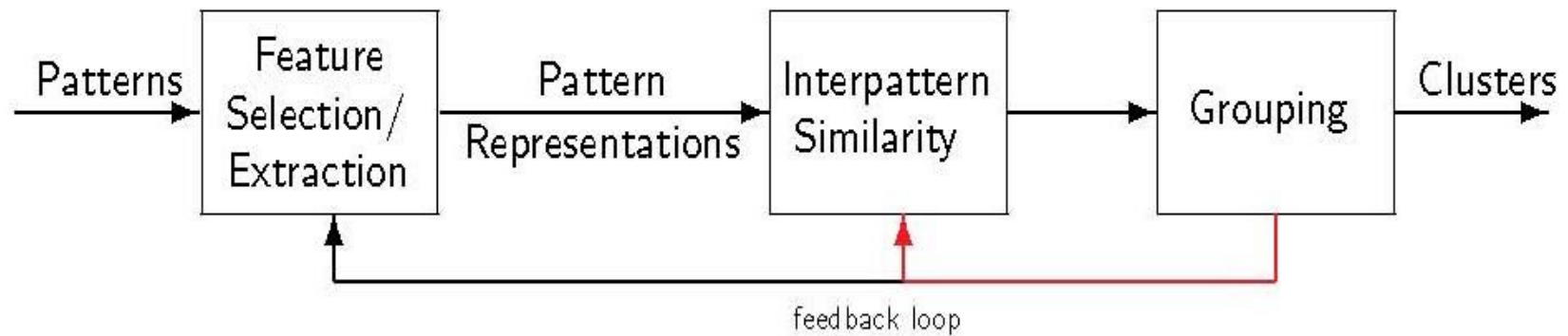
- in der Statistik und
- in der Entscheidungstheorie,
und wird benutzt (u.A.)
- in der Mustererkennung,
- im maschinellen Lernen,
- in der Entscheidungstheorie,
- im Information Retrieval
- ...

Wissenserwerb

Ein Clustering-Prozess besteht aus folgenden Schritten:

- Repräsentation von Beobachtungen (Daten)
- Definition eines Ähnlichkeitsmasses für Daten (passend zum Problembereich)
- den eigentlichen Clustering-Vorgang
- Datenabstraktion (bei Bedarf)
- Beurteilung des Outputs (bei Bedarf)

Wissenserwerb



Wissenserwerb

Repräsentation von Beobachtungen (Daten) umfasst die Festlegung

- der Anzahl von Clustern
- der Menge der verfügbaren Beobachtungen
- von Anzahl, Typen und Skalen der Merkmale zur Beschreibung der Beobachtungen.

Dabei werden oft die folgenden beiden Techniken verwendet:

- **Merkmalsselektion:** Auswahl eines (möglichst effektiven) Teils der ursprünglichen Merkmale;
- **Merkmalsextraktion:** Die ursprünglichen Merkmale werden transformiert, um neue aussagekräftige Merkmale zu definieren

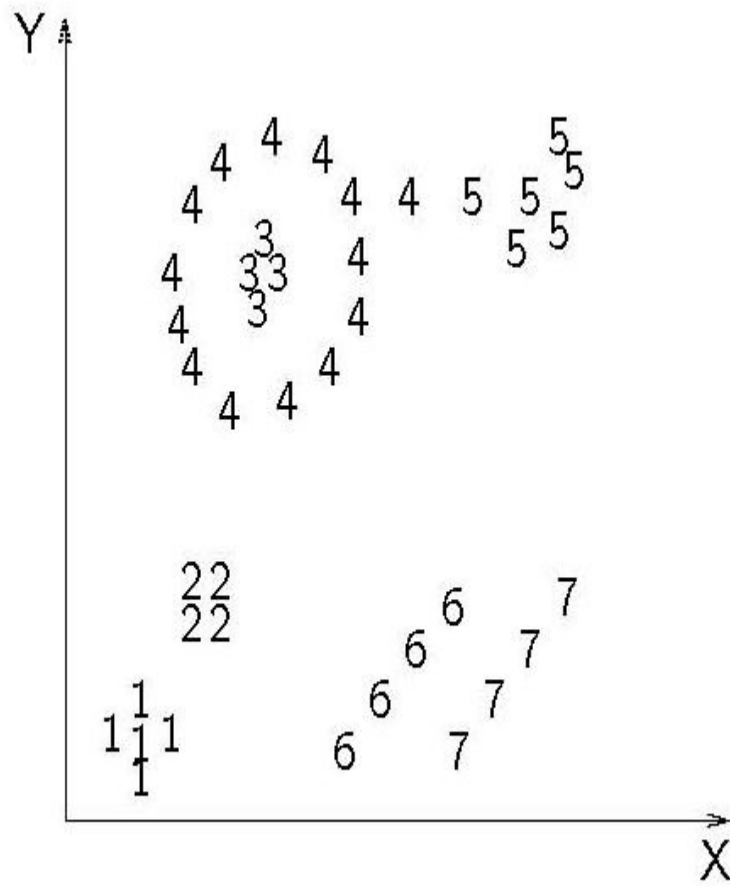
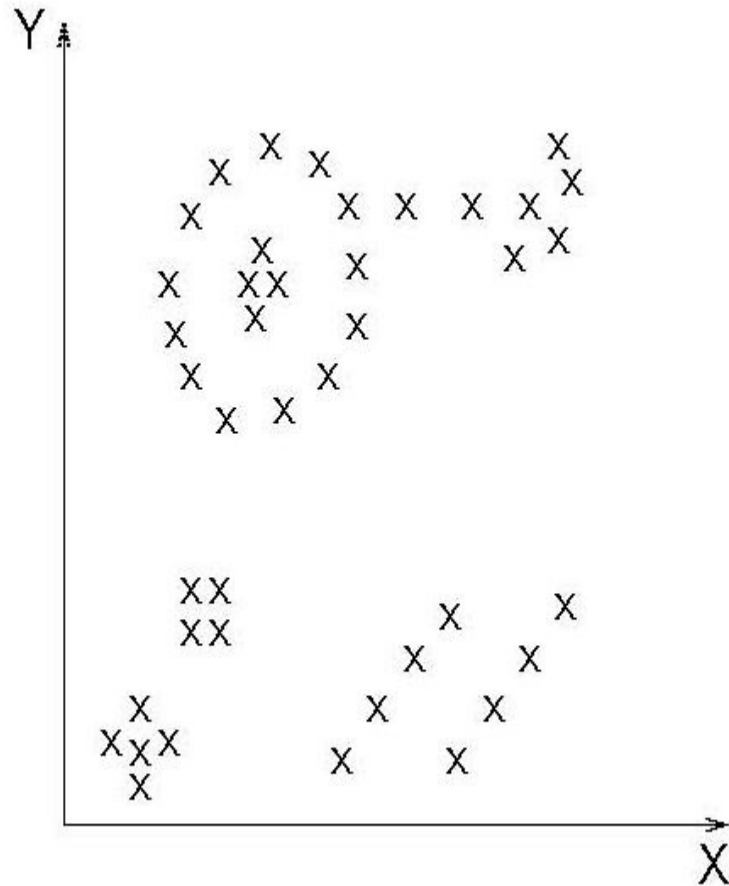
Wissenserwerb

- Datenähnlichkeit wird in der Regel definiert mit Hilfe einer Distanzfunktion, die Unterschiede zwischen Daten paarweise misst.
Beispiel: Euklidische Distanz
- In der Datenabstraktion wird eine möglichst einfache und kompakte Repräsentation des Clustering-Ergebnisses vorgenommen. Jeder Cluster wird kompakt beschrieben, z.B. durch prototypische Eigenschaften oder durch repräsentative Beobachtungen.

Wissenserwerb

- Es gibt eine Vielzahl unterschiedlicher Clustering-Verfahren, aber es gibt keines, das universell brauchbar ist.
- Clustering-Verfahren basieren in der Regel auf impliziten Annahmen über die Form von Clustern, so dass nicht alle Clusterformen von einem Clustering-Verfahren gleich gut erkannt werden.
- Wissen über den Problembereich und über Details der Datenerhebung sind in der Regel von großem Nutzen beim Clustering.

Wissenserwerb



Wissenserwerb

Eine Beobachtung (Merkmalsvektor, Datum) $\mathbf{x}$ ist im Allg. ein Vektor $\mathbf{x}$ Merkmalswerten x_i :

$$\mathbf{x} = (x_1, \dots, x_d)$$

Merkmale können quantitativer oder qualitativer Natur sein:

- quantitative Merkmale mit
 - ▶ kontinuierlichen Werten (z.B. Gewicht, Größe, Temperatur);
 - ▶ diskreten Werten (z.B. Anzahl der Beschäftigten);
 - ▶ Intervallwerten (z.B. Zeitdauer);
- qualitative Merkmale mit
 - ▶ nominalen Werten (z.B. Farbe);
 - ▶ ordinalen Werten (z.B. qualitative Abstraktionen quantitativer Merkmalswerte wie "niedrig", "mittel", "hoch").

Wissenserwerb

Die Wahl geeigneter Merkmale und Merkmalswerte ist also von entscheidender Bedeutung für das Ergebnis des Clusterings!

Eine Datenmenge $\mathcal{B}$ besteht aus endlich vielen Beobachtungen $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ und wird manchmal auch als eine $n \times d$ Datenmatrix repräsentiert.

Eine Klasse wird als abstrakte Quelle für die Erzeugung von Beobachtungen gesehen. Das Ergebnis des Clusterings sollte die verschiedenen Beobachtungserzeugungsprozesse widerspiegeln.

Wissenserwerb

Der grundlegende Begriff für die Einteilung der Beobachtungen in Cluste ist die Ähnlichkeit zwischen Beobachtungen, die durch Ähnlichkeitsmaße quantifiziert wird. Dual dazu ist der Begriff der Unähnlichkeit, der durch Distanzmaße bestimmt wird.

Das meist gebrauchte Distanzmaß ist der Euklidische Abstand

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^d (x_{i,k} - x_{j,k})^2 \right)^{1/2} = \| \mathbf{x}_i - \mathbf{x}_j \|_2,$$

der ein Spezialfall ($p = 2$) einer Minkowski Metrik ist:

$$d_p(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^d (x_{i,k} - x_{j,k})^p \right)^{1/p}$$

Wissenserwerb

Um die unterschiedliche Relevanz von Merkmalen zu berücksichtigen, kann man dies noch mit Gewichten w_k versehen:

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^d w_k (x_{i,k} - x_{j,k})^2 \right)^{1/2}$$

Ein Distanzmaß, das statistische Varianzen und Korrelationen berücksichtigt, ist der Mahalanobis-Abstand

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j) \Sigma^{-1} (\mathbf{x}_i - \mathbf{x}_j)^T$$

wobei Σ hier die Kovarianz-Matrix der Beobachtungen ist.

Wissenserwerb

Einige Distanzmaße berechnen eine relative Ähnlichkeit, die von der Menge der umgebenden Beobachtungen (dem Kontext) abhängig ist. Ein Beispiel ist hier die gegenseitige Nachbar-Distanz (*mutual neighbor distance*, MND):

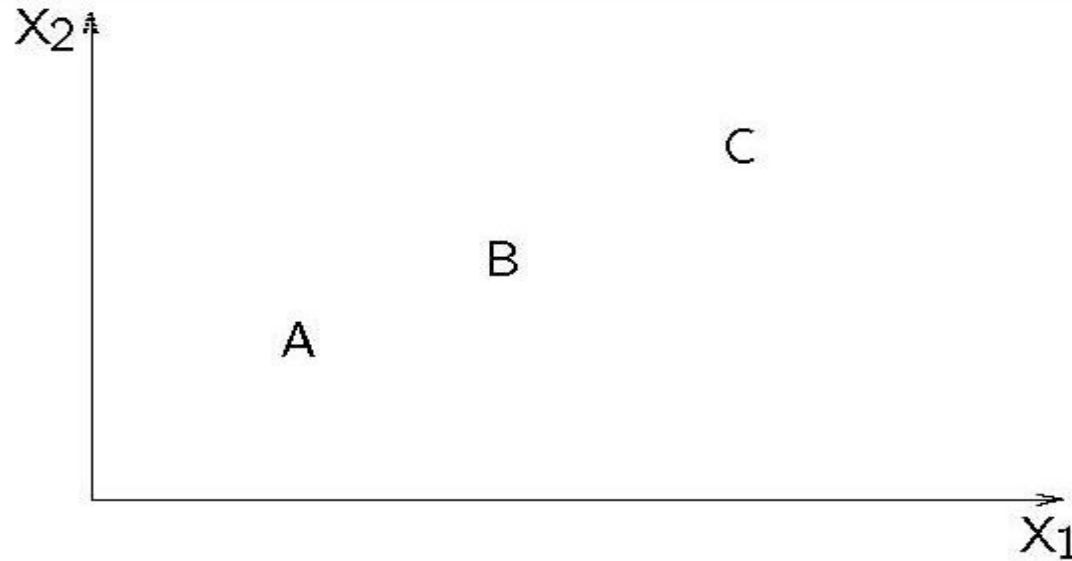
$$MND(\mathbf{x}_i, \mathbf{x}_j) = NN(\mathbf{x}_i, \mathbf{x}_j) + NN(\mathbf{x}_j, \mathbf{x}_i)$$

wobei

$NN(\mathbf{x}_i, \mathbf{x}_j) = \#$ nächste Nachbarn von $\mathbf{x}_j$ bezgl. $\mathbf{x}_i$

nicht symmetrisch!

Wissenserwerb



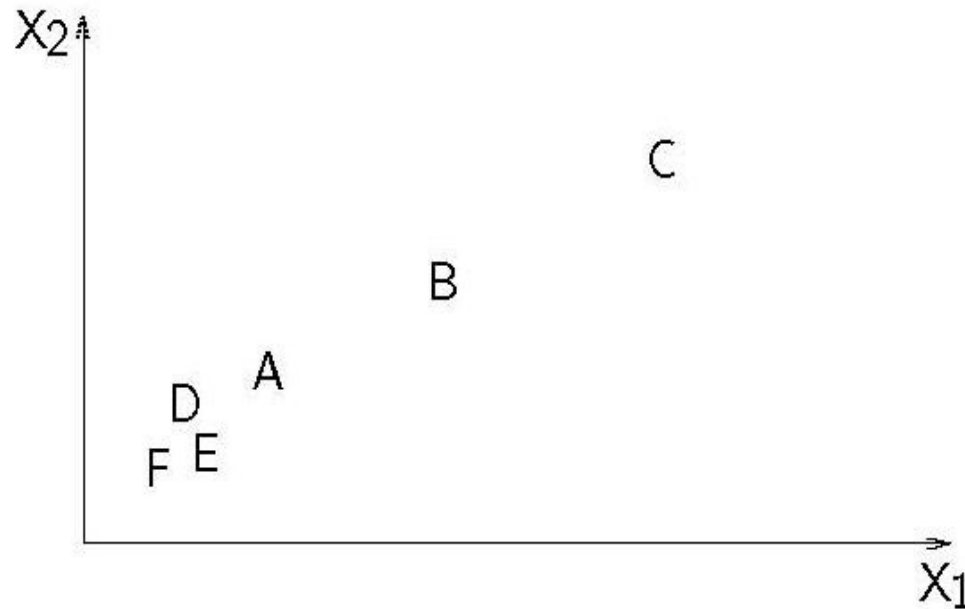
A und B sind jeweils füreinander nächste Nachbarn, d.h.

$$NN(A, B) = NN(B, A) = 1, \text{ also } MND(A, B) = 2$$

Es ist jedoch

$$NN(B, C) = 1, NN(C, B) = 2, \text{ also } MND(B, C) = 3$$

Wissenserwerb



Nun wurden die Punkte D, E, F hinzugefügt, und es ergeben sich folgende Abstandswerte:

$$MND(B, C) = NN(B, C) + NN(C, B) = 1 + 2 = 3$$

$$MND(A, B) = NN(A, B) + NN(B, A) = 1 + 4 = 5$$

Wissenserwerb

Man unterscheidet hierarchisches und partitionierendes Clustern:

- hierarchisches Clustern erzeugt eine Folge verschachtelter Partitionierungen der Beobachtungsmenge;
- partitionierendes Clustern erzeugt eine Partitionierung der Menge der Beobachtungen.

Wissenserwerb

Zusätzlich kann man Clustering-Techniken noch nach den folgenden Kriterien unterscheiden:

- aggregativ vs. divisiv:
 - ▶ Agglomeratives Clustering beginnt mit Clustern, in denen genau eine Beobachtung liegt, die so lange verschmolzen werden, bis ein vorgegebenes Qualitätskriterium erfüllt ist.
 - ▶ Divisives Clustering beginnt mit einem einzigen Cluster, in dem alle Beobachtungen liegen, und spaltet diesen Cluster so lange auf, bis ein vorgegebenes Qualitätskriterium erfüllt ist.

Wissenserwerb

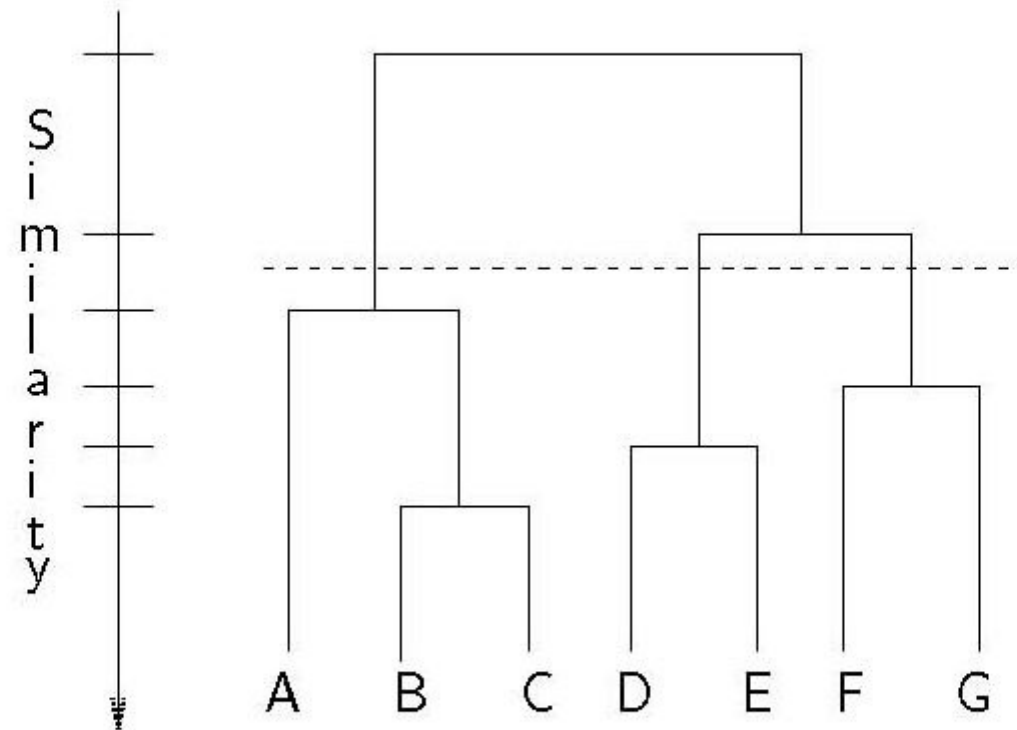
- Monothetisch vs. polythetisch: Hier wird danach unterschieden, ob bei der Bestimmung der Ähnlichkeit alle Merkmale gleichzeitig (polythetisch) oder nacheinander (monothetisch) berücksichtigt werden. Die meisten Algorithmen sind polythetisch.
- Exakt vs. Fuzzy:
 - ▶ Exaktes Clustering weist jede Beobachtung genau einem Cluster zu.
 - ▶ Fuzzy-Clustering weist jeder Beobachtung einen Zugehörigkeitsgrad zu jedem Cluster zu.

Fuzzy-Clustering kann in exaktes Clustering überführt werden, indem man jeder Beobachtung den Cluster mit dem höchsten Zugehörigkeitsgrad zuweist.

Wissenserwerb

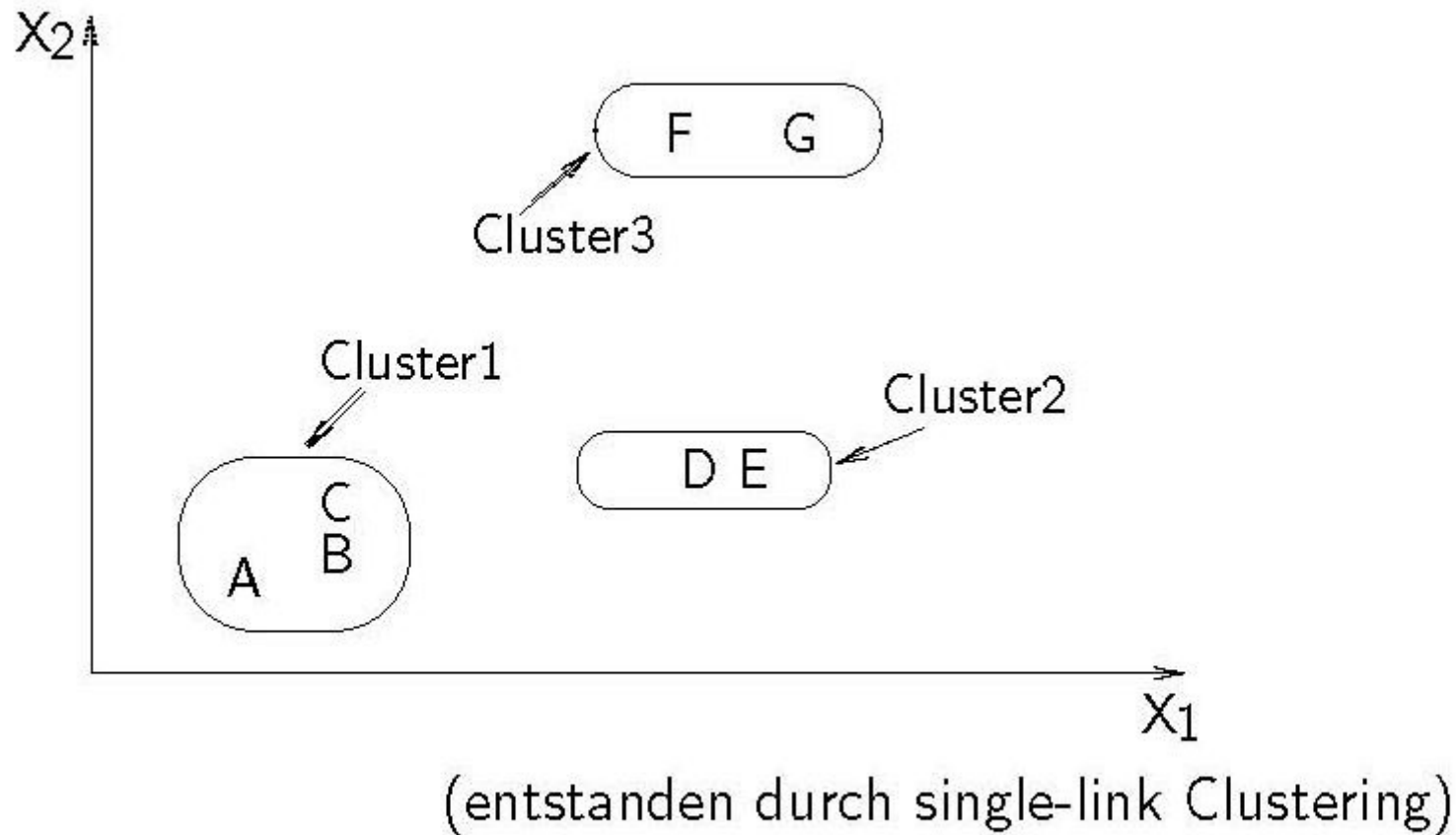
- 1 Bilde Cluster mit jeweils einem Element. Berechne die Distanzen zwischen jeweils zwei verschiedenen Daten und ordne diese Distanzen d_k in aufsteigender Ordnung.
- 2 Bilde für jede Distanz d_k einen Graph auf den Beobachtungen, so dass jeweils Beobachtungen mit einem Abstand kleiner als d_k durch eine Kante verbunden werden.
- 3 Wiederhole Schritt 2 so lange, bis der entstehende Graph **einfach zusammenhängend** ist.
- 4 Die Cluster entstehen jeweils aus den **einfachen Zusammenhangskomponenten** der Graphen, wobei sich Hierarchien in Abhängigkeit von der gewählten Unähnlichkeit (Distanz) ergeben.

Wissenserwerb



Das zugehörige Dendrogramm

Wissenserwerb



Wissenserwerb

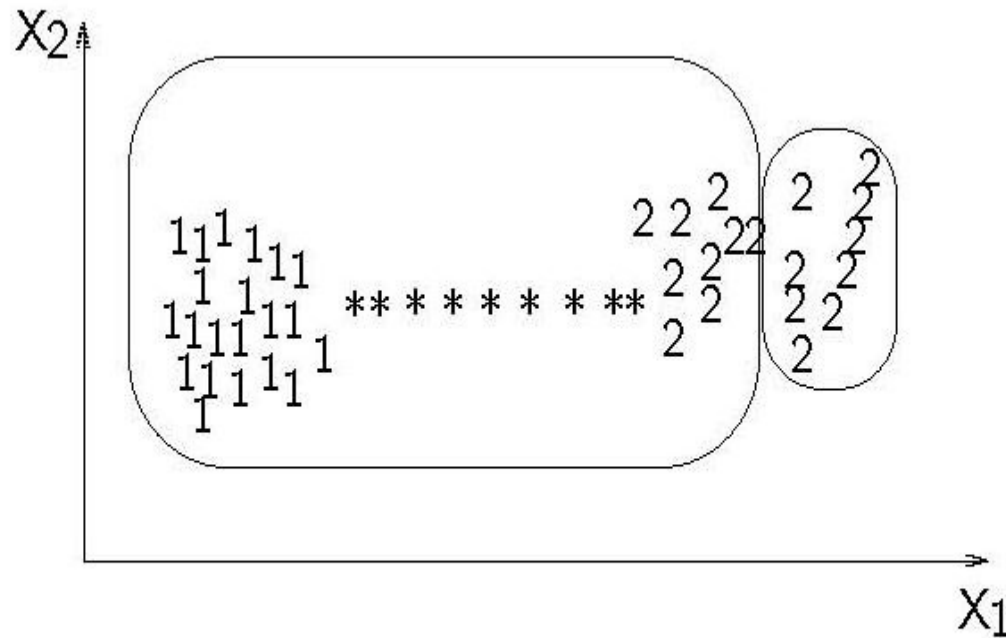
- 1 Bilde Cluster mit jeweils einem Element. Berechne die Distanzen zwischen jeweils zwei verschiedenen Beobachtungen und ordne diese Distanzen d_k in aufsteigender Ordnung.
- 2 Bilde für jede Distanz d_k einen Graph auf den Beobachtungen, so dass jeweils Beobachtungen mit einem Abstand kleiner als d_k durch eine Kante verbunden werden.
- 3 Wiederhole Schritt 2 so lange, bis der entstehende Graph **vollständig zusammenhängend** ist.
- 4 Die Cluster entstehen jeweils aus den **vollständigen Zusammenhangskomponenten** der Graphen, wobei sich Hierarchien in Abhängigkeit von der gewählten Unähnlichkeit ergeben.

Wissenserwerb

- Beim single-link Clustering berechnet sich der Abstand zweier Cluster als Minimum der Abstände der Paare aus verschiedenen Clustern.
- Beim complete-link Clustering berechnet sich der Abstand zweier Cluster als Maximum der Abstände der Paare aus verschiedenen Clustern.

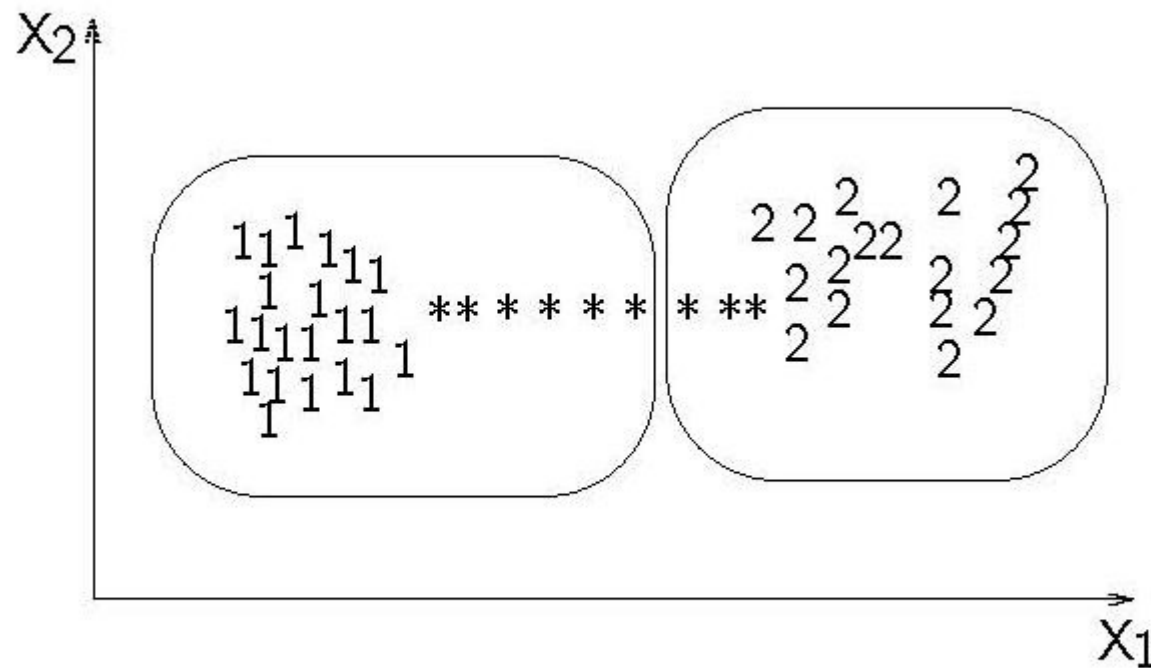
Complete-link Clustering erzeugt tendenziell eher kompakte Cluster, während die von single-link Clustering erzeugten Cluster häufig länglich sind.

Wissenserwerb



Single-Link-Clustering einer Beobachtungsmenge

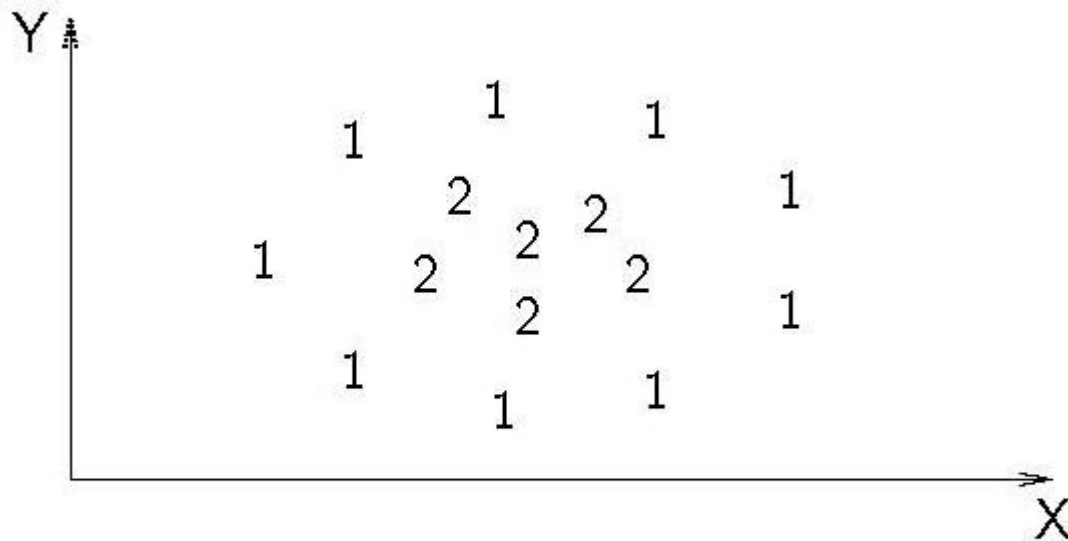
Wissenserwerb



Complete-Link-Clustering derselben Beobachtungsmenge

Wissenserwerb

Andererseits erweist sich Single-Link-Clustering oft als vielseitiger:



Single-Link-Clustering erkennt hier die beiden konzentrischen Cluster, Complete-Link-Clustering jedoch nicht.

Wissenserwerb

Das am weitesten verbreitete globale Kriterien zur Beurteilung der Güte einer Partitionierung ist der quadratische Fehler (*squared error*), der bei einer Partitionierung $\mathcal{P}$ (mit K Clustern) der Datenmenge $\mathcal{B}$ wie folgt berechnet wird:

$$e^2(\mathcal{B}, \mathcal{P}) = \sum_{j=1}^K \sum_{i=1}^{n_j} \| \mathbf{x}_i^{(j)} - \mathbf{c}_j \|^2$$

wobei $\mathbf{x}_i^{(j)}$ die i -te Beobachtung des j -ten Clusters und $\mathbf{c}_j$ das Zentroid des j -ten Clusters ist.

Der populärste Clustering-Algorithmus, der das Quadratfehler-Kriterium verwendet, ist das sog. *k-means-Clustering*.

Wissenserwerb

- Wähle k beliebige Beobachtungen (oder Punkte) als Zentroiden von k Clustern.
- Ordne jede Beobachtung dem nächsten Zentroid zu.
- Berechne neue Zentroide der so entstehenden Cluster (z.B. als Mittelwerte).
- Wiederhole Schritt 2 und 3 so lange, bis ein Konvergenzkriterium erfüllt ist.

Typische Konvergenzkriterien sind:

- die Partitionierung bleibt stabil;
- der Quadratfehler verändert sich nicht.

Wissenserwerb

Vorteile von k -means:

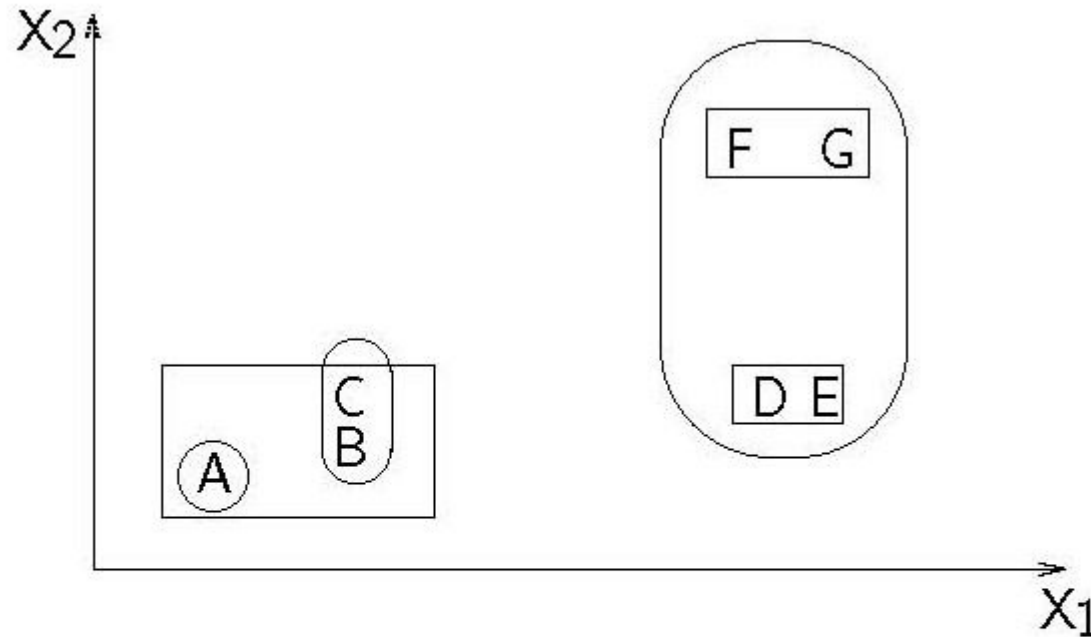
- leicht zu implementieren;
- lineare Komplexität in der Zahl der Beobachtungen (bei konstanter Zahl der Cluster und der Iterationsschritte).

Nachteile von k -means:

- Güte der Partitionierung hängt stark ab von der initialen Partitionierung;
- kann in lokales Optimum konvergieren.

Deswegen werden oft mehrere Läufe mit zufälligen Start-Zentroiden durchgeführt.

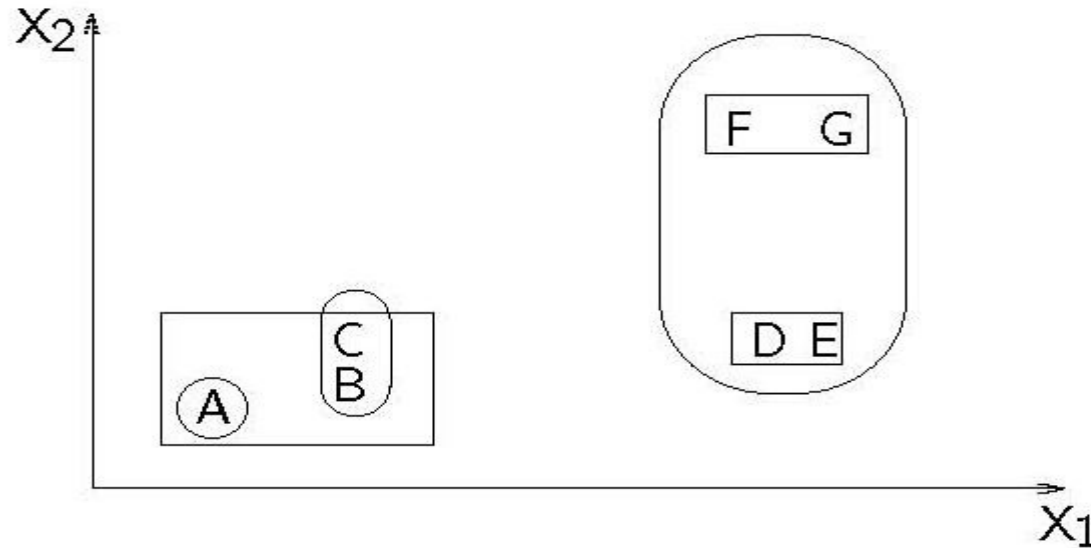
Wissenserwerb



Wählen wir hier als initiale Zentroide die Beobachtungen A, B, C , so erhalten wir die Ellipsen-Partitionierung

$$\{A\}, \{B, C\}, \{D, E, F, G\}$$

Wissenserwerb



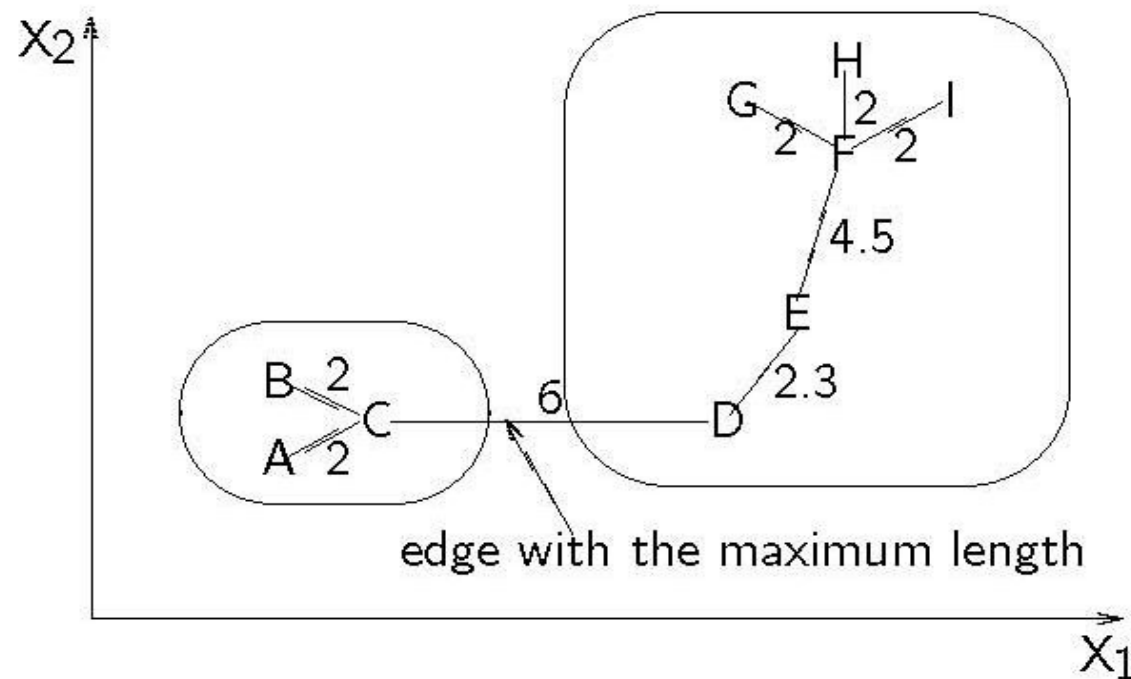
Das globale Minimum des Quadratfehlers (bei drei Clustern) wird bei der Rechteck-Partitionierung

$$\{A, B, C\}, \{D, E\}, \{F, G\}$$

angenommen.

Wissenserwerb

Das bekannteste graphentheoretische, divisive Clustering-Verfahren benutzt den minimalen Spannbaum (minimal spanning tree, MST), der alle Beobachtungen verbindet. Durch Entfernen der jeweils längsten Kanten entstehen Cluster.



Wissenserwerb

COBWEB ist ein inkrementelles, hierarchisches Clustering-Verfahren, das das heuristische Qualitätsmaß Kategorie-Nutzen (*category utility, CU*) verwendet.

Sei A_i ein Merkmal, V_{ij} ein Merkmalswert, und C_k ein Cluster; die Wahrscheinlichkeitsfunktion P werde als relative Häufigkeit berechnet.

- Die Intra-Klassen-Ähnlichkeit wird gemessen mittels

$$P(A_i = V_{ij} | C_k)$$

- Die Inter-Klassen-Unähnlichkeit wird gemessen mittels

$$P(C_k | A_i = V_{ij})$$

Wissenserwerb

Ein gewichtetes Maß aus Ähnlichkeiten und Unähnlichkeiten für eine Partitionierung $C_1, \dots, C_n$ ergibt sich dann als

$$\begin{aligned} & \sum_{k=1}^n \sum_i \sum_j P(A_i = V_{ij}) P(C_k | A_i = V_{ij}) P(A_i = V_{ij} | C_k) \\ &= \sum_{k=1}^n P(C_k) \sum_i \sum_j P(A_i = V_{ij} | C_k)^2 \end{aligned}$$

dies entspricht der erwarteten Anzahl von Attributwerten, die korrekt für eine zufällige Beobachtung aus einem Cluster geraten werden können, aufsummiert über alle Cluster.

Wissenserwerb

Der Kategorie-Nutzen CU ergibt sich nun als der informative Gewinn, der aus der Clusterung gezogen werden kann, nämlich als die Differenz zwischen diesem Erwartungswert bei gegebener Partitionierung und dem entsprechenden Erwartungswert **ohne Partitionierung** (i.e. $\sum_i \sum_j P(A_i = V_{ij})^2$). Damit ist

$$CU(\{C_1, \dots, C_n\}) = \frac{\sum_{k=1}^n P(C_k) [\sum_i \sum_j P(A_i = V_{ij} | C_k)^2 - \sum_i \sum_j P(A_i = V_{ij})^2]}{n}$$

Wissenserwerb

Ein **probabilistisches Konzept** ist eine probabilistische Repräsentation einer Klasse, bei der die Wahrscheinlichkeiten aller Merkmalswerte aufgelistet werden.

Beispiel: Es seien die folgenden Beschreibungen von Tieren gegeben:

<i>Name</i>	<i>BodyCover</i>	<i>HeartChamber</i>	<i>BodyTemp</i>	<i>Fertilization</i>
'mammal'	hair	four	regulated	internal
'bird'	feathers	four	regulated	internal
'reptile'	cornified-skin	imperfect-four	unregulated	internal
'amphibian'	moist-skin	three	unregulated	external
'fish'	scales	two	unregulated	external

Wissenserwerb

Ein probabilistisches Konzept als probabilistische Beschreibung der Klasse {fish, amphibian, mammal} ergibt sich dann wie folgt:

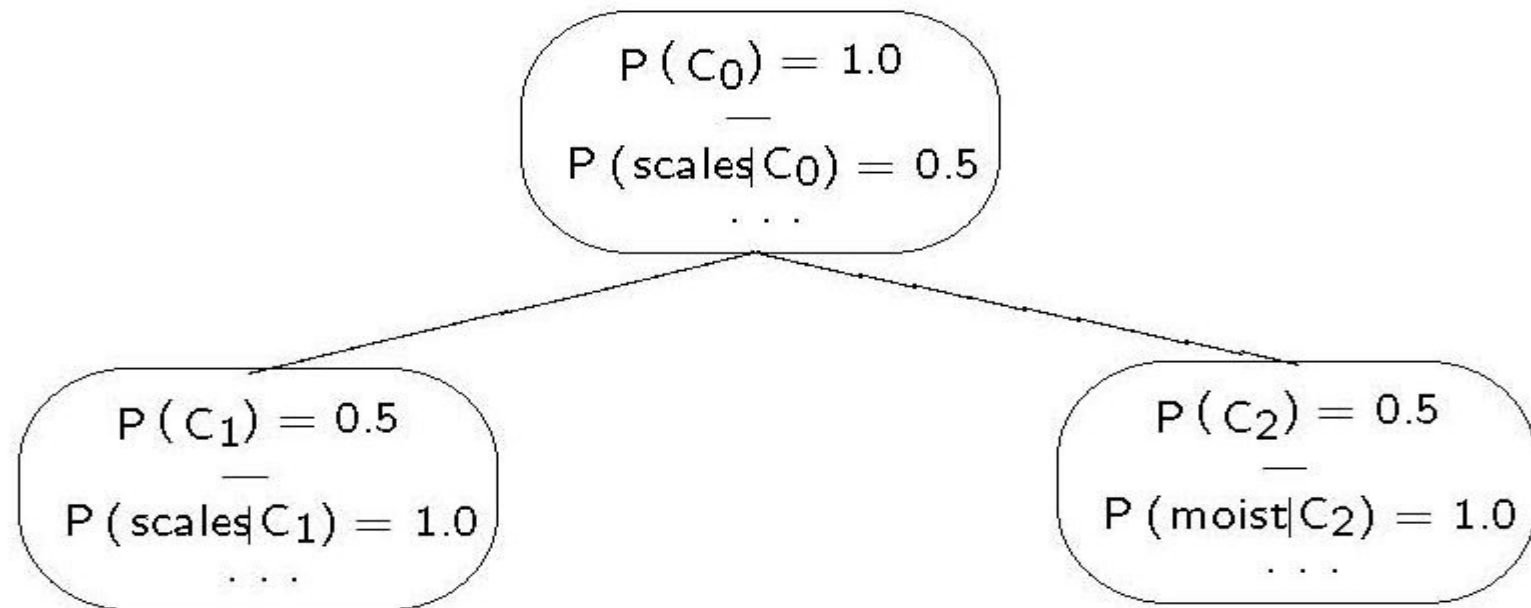
<i>Attributes</i>	<i>Values and probabilities</i>
BodyCover	scales [0.33], moist-skin [0.33], hair [0.33]
HeartChamber	two [0.33], three [0.33], four [0.33]
BodyTemp	unregulated [0.67], regulated [0.33]
Fertilization	external [0.67], internal [0.33]

Wissenserwerb

COBWEB ordnet Objekte inkrementell in einen Klassifikationsbaum ein, dessen Knoten aus probabilistischen Konzepten bestehen, die Objektklassen repräsentieren. Dabei wird auf jedem Level eine der folgenden Operationen vorgenommen:

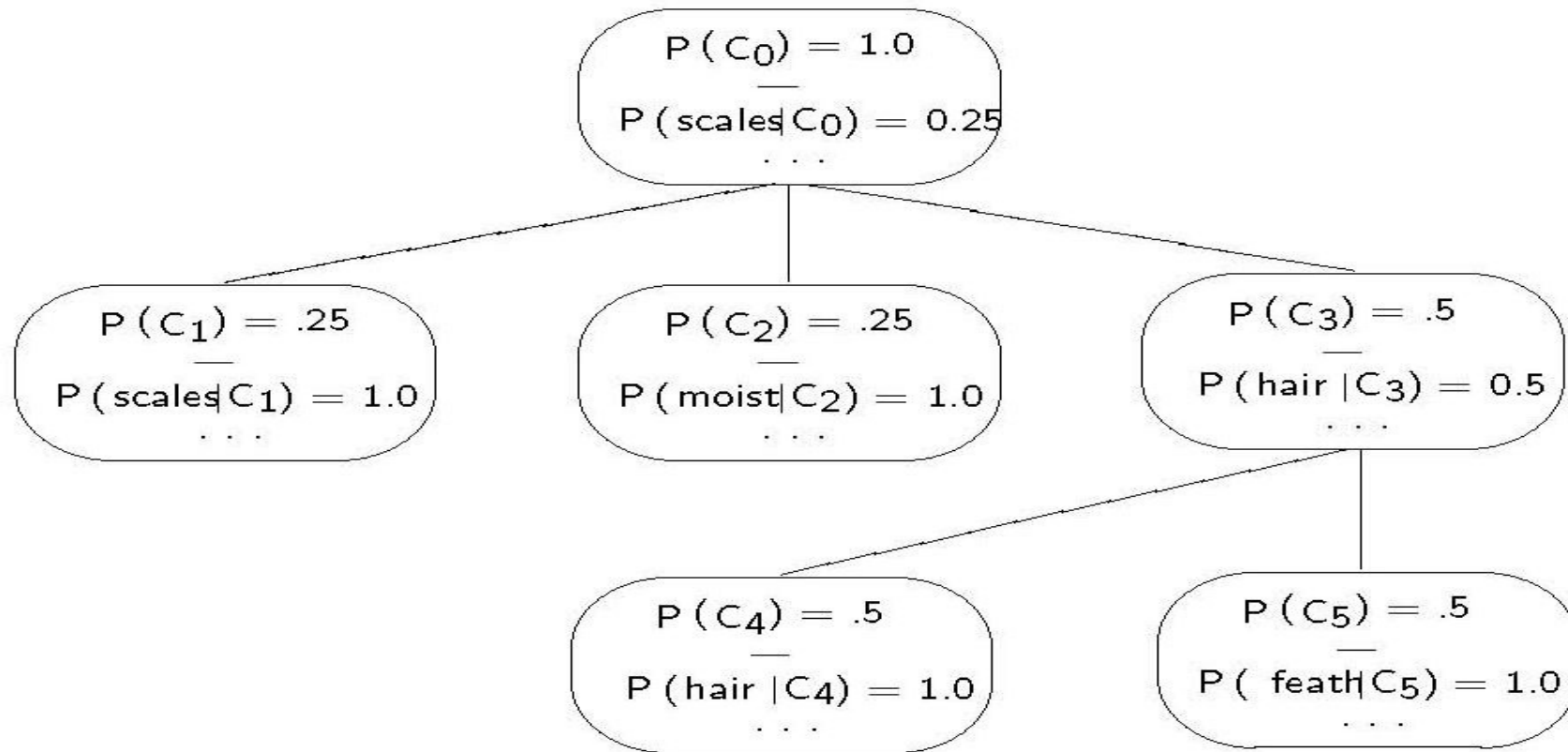
- Einordnen des Objektes in eine bestehende Klasse: COBWEB ordnet das Objekt versuchsshalber in jede Klasse ein, und die entstehende Gesamtqualität wird mittels *CU* berechnet. Schließlich wird das Objekt dem Knoten zugeordnet, der zu der besten *CU* führt.
- Einfügen einer neuen Klasse: Der *CU* der besten Positionierung wird verglichen mit dem *CU*, der sich aus dem Einfügen eines neuen Knotens mit dem neuen Objekt ergibt; ist der letztere besser, so wird ein neuer Cluster angelegt.

Wissenserwerb



Ein durch die *fish*- und *amphibian*-Objekte geschaffener Klassifikationsbaum.

Wissenserwerb



Wissenserwerb

COBWEB arbeitet mit einer variablen Zahl von Clustern – die Zahl der Cluster wird nicht von vorneherein vorgegeben, sondern durch die Beobachtungsmenge bestimmt.

Die Güte des Klassifikationsbaums kann während des Verfahrens noch verbessert werden, indem Geschwisterknoten miteinander verschmolzen werden oder Knoten aufgesplittet werden.

Entscheidungsbäume

- Wozu benutzt man Entscheidungsbäume?
 - ▶ Entscheidungsbäume dienen der Klassifikation von Objekten, die durch (Attribut, Wert)-Paare beschrieben sind;
 - ▶ hier: binäre Klassifikation, d.h. Ja/Nein-Entscheidung
- Was sind Entscheidungsbäume?
- Wie benutzt man Entscheidungsbäume?
- Wie baut man Entscheidungsbäume auf?

Wissenserwerb

- Wurzel und innere Knoten des Baumes sind mit Attributen markiert und repräsentieren Abfragen, welchen Wert das betrachtete Objekt für das jeweilige Attribut hat.
- Die von einem Knoten ausgehenden Kanten sind mit den zugehörigen möglichen Attributwerten markiert.
- Die Blätter enthalten die Klassifikation.

Objekte werden durch vollständige Pfade durch den Baum klassifiziert.

Wissenserwerb

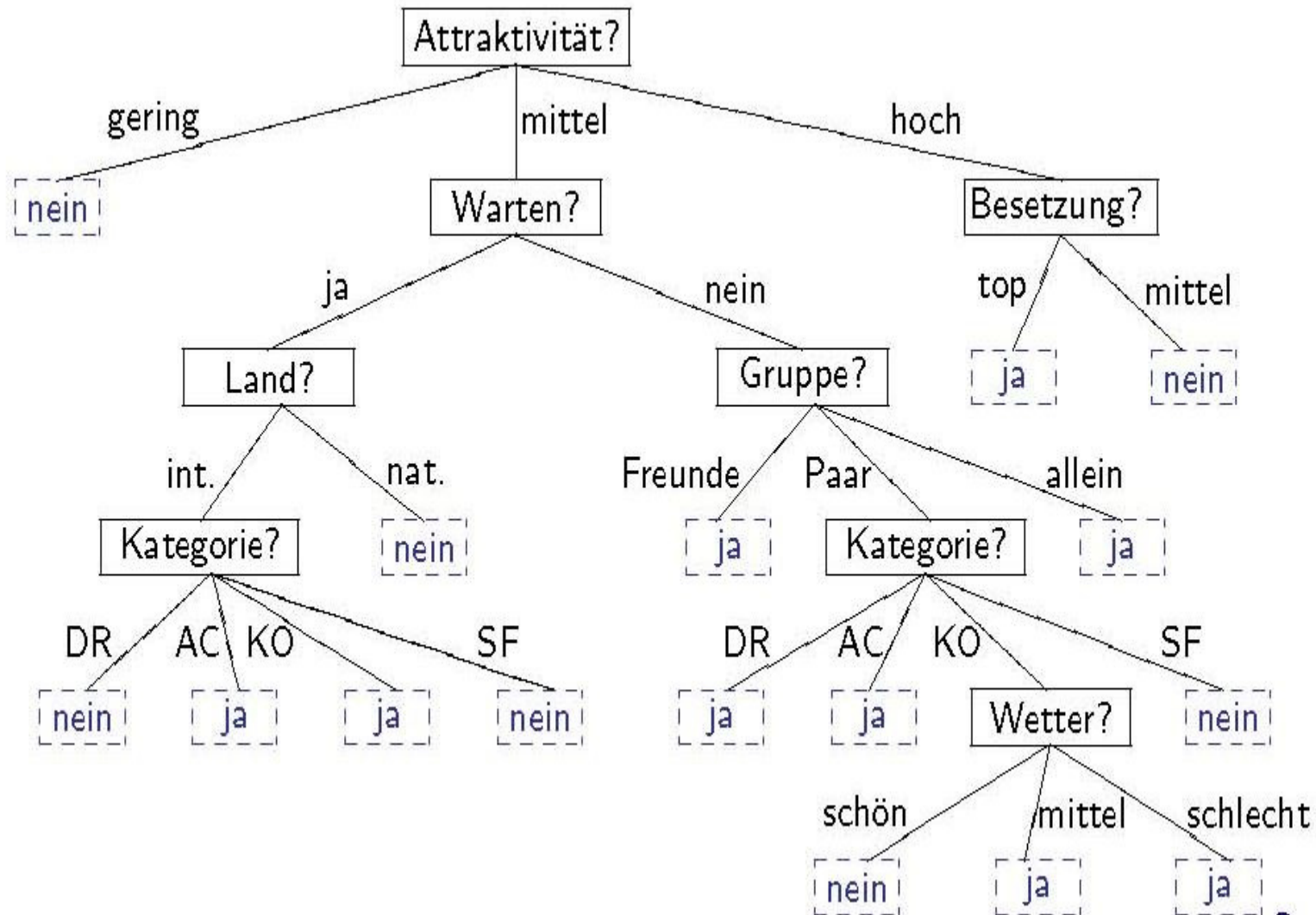
Entscheidungssituation: "Kino – ja oder nein?"

zu klassifizierende Objekte: Situationen

relevante Attribute:

Attribut	Werte
<i>Attraktivität</i>	<i>hoch, mittel, gering</i>
<i>Preis</i>	normal (\$) oder mit Zuschlag (\$\$)
<i>Loge</i>	verfügbar (<i>ja</i>) oder nicht (<i>nein</i>)
<i>Wetter</i>	<i>schön, mittel, schlecht</i>
<i>Warten</i>	Wartezeit (<i>ja</i>) oder nicht (<i>nein</i>)
<i>Besetzung</i>	Cast und Regie sind <i>top, mittel</i> (mäßig)
<i>Kategorie</i>	<i>Action (AC), Komödie (KO), Drama (DR), SciFi (SF)</i>
<i>Reservierung</i>	besteht (<i>ja</i>) oder nicht (<i>nein</i>)
<i>Land</i>	<i>nationale (N) oder internationale (I)</i> Produktion
<i>Gruppe</i>	mit <i>Freunde(n)</i> , als <i>Paar</i> , oder <i>allein</i>

Wissenserwerb



Wissenserwerb

Lernaufgabe:

Erzeuge Entscheidungsbaum aus Trainingsmenge (= Menge von **klassifizierten** Beispielen), so dass

- Beispiele der Trainingsmenge korrekt klassifiziert werden und
- sich der Entscheidungsbaum auch für andere Beispiele generalisieren lässt

→ **induktives Lernen**

Wissenserwerb

<i>Beisp.</i>	<i>Attr.</i>	<i>Preis</i>	<i>Loge</i>	<i>Wetter</i>	<i>Warten</i>	<i>Bes.</i>	<i>Kat.</i>	<i>Land</i>	<i>Res.</i>	<i>Gruppe</i>	<i>Kino?</i>
X_1	hoch	\$\$	ja	schlecht	ja	top	AC	int.	ja	Freunde	ja
X_2	mittel	\$	ja	mittel	nein	mittel	KO	int.	nein	Paar	ja
X_3	mittel	\$	nein	mittel	ja	mittel	DR	int.	nein	Freunde	nein
X_4	gering	\$	ja	mittel	ja	mittel	SF	int.	nein	allein	nein
X_5	mittel	\$	ja	mittel	nein	mittel	DR	int.	nein	Paar	ja
X_6	hoch	\$\$	ja	schön	nein	top	SF	int.	ja	Freunde	ja
X_7	mittel	\$	ja	schlecht	nein	mittel	KO	nat.	nein	Freunde	ja
X_8	mittel	\$	nein	schlecht	ja	mittel	AC	int.	nein	Freunde	ja
X_9	gering	\$	ja	schön	nein	mittel	KO	nat.	nein	Freunde	nein
X_{10}	mittel	\$	ja	schön	nein	mittel	KO	int.	nein	Paar	nein
X_{11}	hoch	\$	ja	mittel	ja	top	DR	int.	nein	Paar	ja
X_{12}	mittel	\$	nein	schlecht	ja	mittel	AC	nat.	nein	allein	nein
X_{13}	hoch	\$\$	ja	mittel	ja	mittel	SF	int.	nein	allein	nein
X_{14}	mittel	\$	ja	schön	ja	top	DR	int.	ja	Freunde	nein
X_{15}	mittel	\$	ja	schlecht	nein	mittel	AC	int.	nein	Paar	ja

Wissenserwerb

Ein wichtiger Vorteil eines Entscheidungsbaumes ist die Möglichkeit des Generalisierens: Es soll ein Muster aus den Beispielen extrahiert werden, das sich auf neue Situationen verallgemeinern lässt. Ziel des Lernens ist es daher, einen möglichst kompakten Entscheidungsbaum zu generieren. Dieser Überlegung liegt ein generelles Prinzip des induktiven Lernens zugrunde:

Occam's Razor

Bevorzuge die einfachste Hypothese, die konsistent mit allen Beobachtungen ist.

Wissenserwerb

- ① Wähle ein Attribut **a** für den nächsten Knoten.
- ② Für jeden Wert von **a** erzeuge einen Nachfolgeknoten; markiere die zugehörige Kante mit diesem Wert.
- ③ Verteile die aktuelle Trainingsmenge auf die Nachfolgeknoten, entsprechend den jeweiligen Werten von **a**.
- ④ wende TDIDT auf die neuen Blattknoten an (Rekursion)

Wissenserwerb

An den (aktuellen) Blattknoten können vier verschiedene Fälle auftreten

- ① alle Beispiele haben die gleiche Klassifikation $C \rightarrow$ Klassifikation C
- ② Beispielmenge ist leer $\rightarrow$ Default-Klassifikation
- ③ es gibt noch positive und negative Beispiele,
aber es sind keine Attribute mehr übrig $\rightarrow$ Inkonsistenz
(es gibt Beispiele mit genau denselben Attributwerten, aber
unterschiedlicher Klassifikation)
- ④ es gibt noch positive und negative Beispiele,
die aktuelle Menge der Attribute ist nicht leer
 $\rightarrow$ nächster Rekursionsschritt

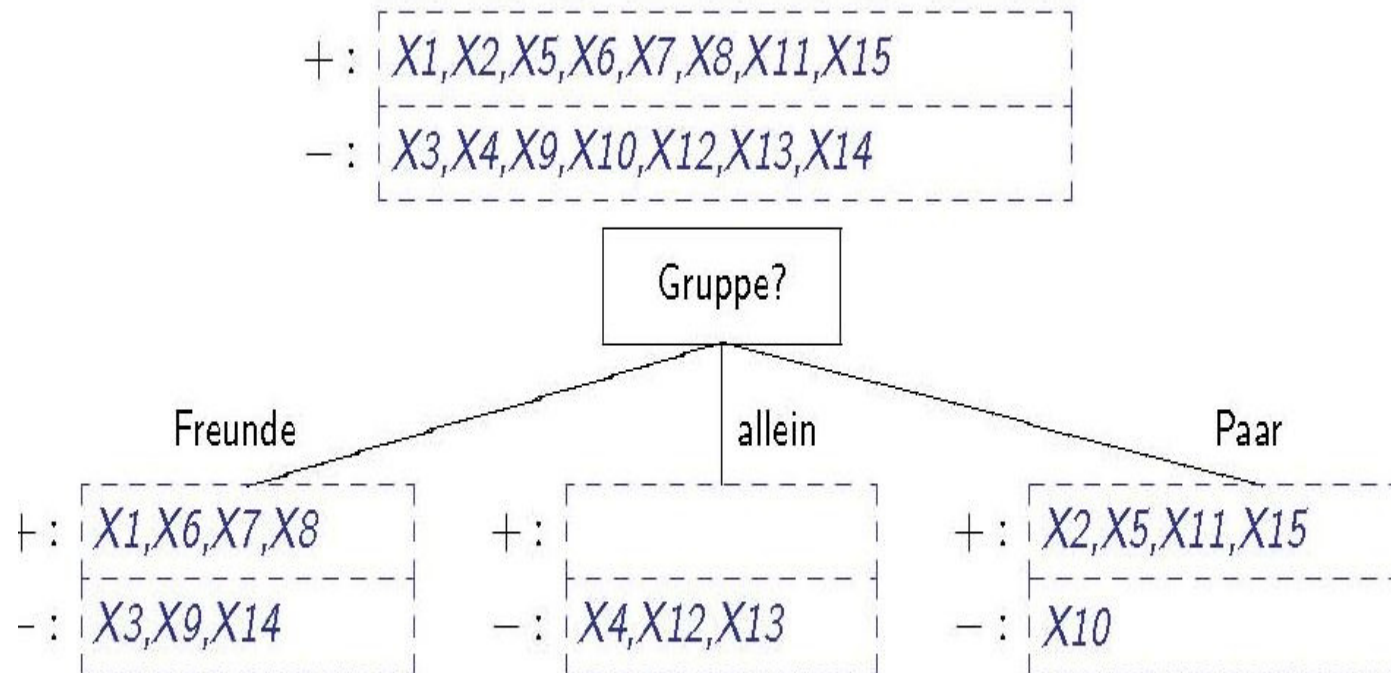
Zentrales Problem: Wie findet man das (jeweils nächste) **beste Attribut**,
um den Entscheidungsbaum aufzubauen ?

Wissenserwerb

Welches Attribut **a** soll als nächstes gewählt werden?

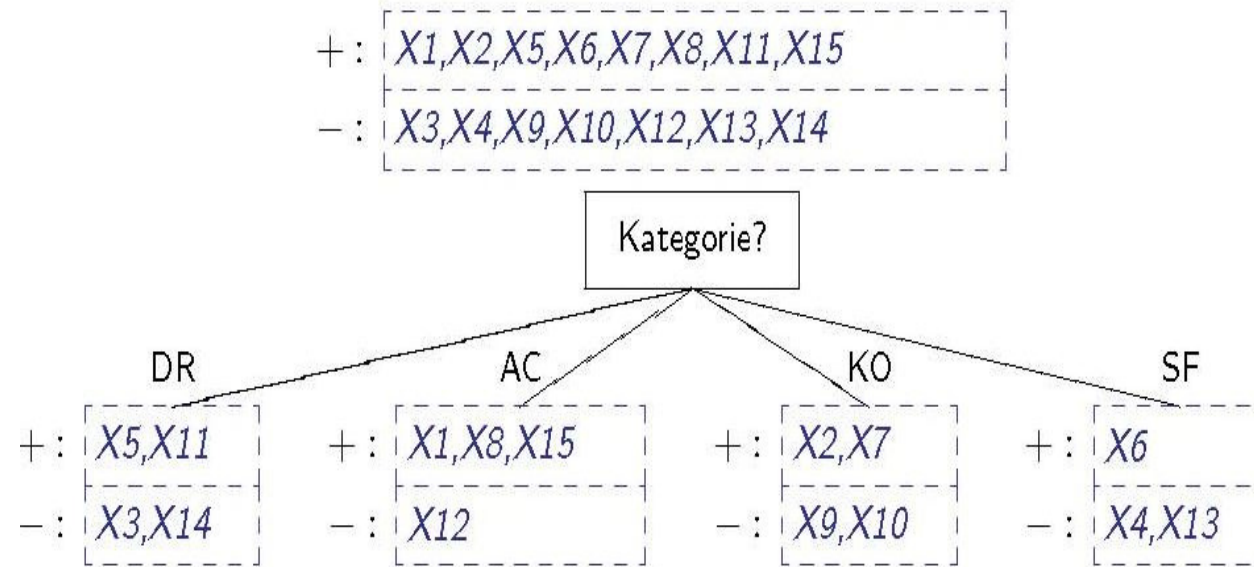
- Wähle dasjenige Attribut, das am wichtigsten ist, d.h. das
 - ▶ soviel Beispiele wie möglich klassifiziert
(→ Kardinalitätskriterium);
 - ▶ die meiste Information enthält (→ ID3, C4.5).

Wissenserwerb



Beim Wert *Gruppe* = *allein* werden drei Beispiele vollständig klassifiziert.

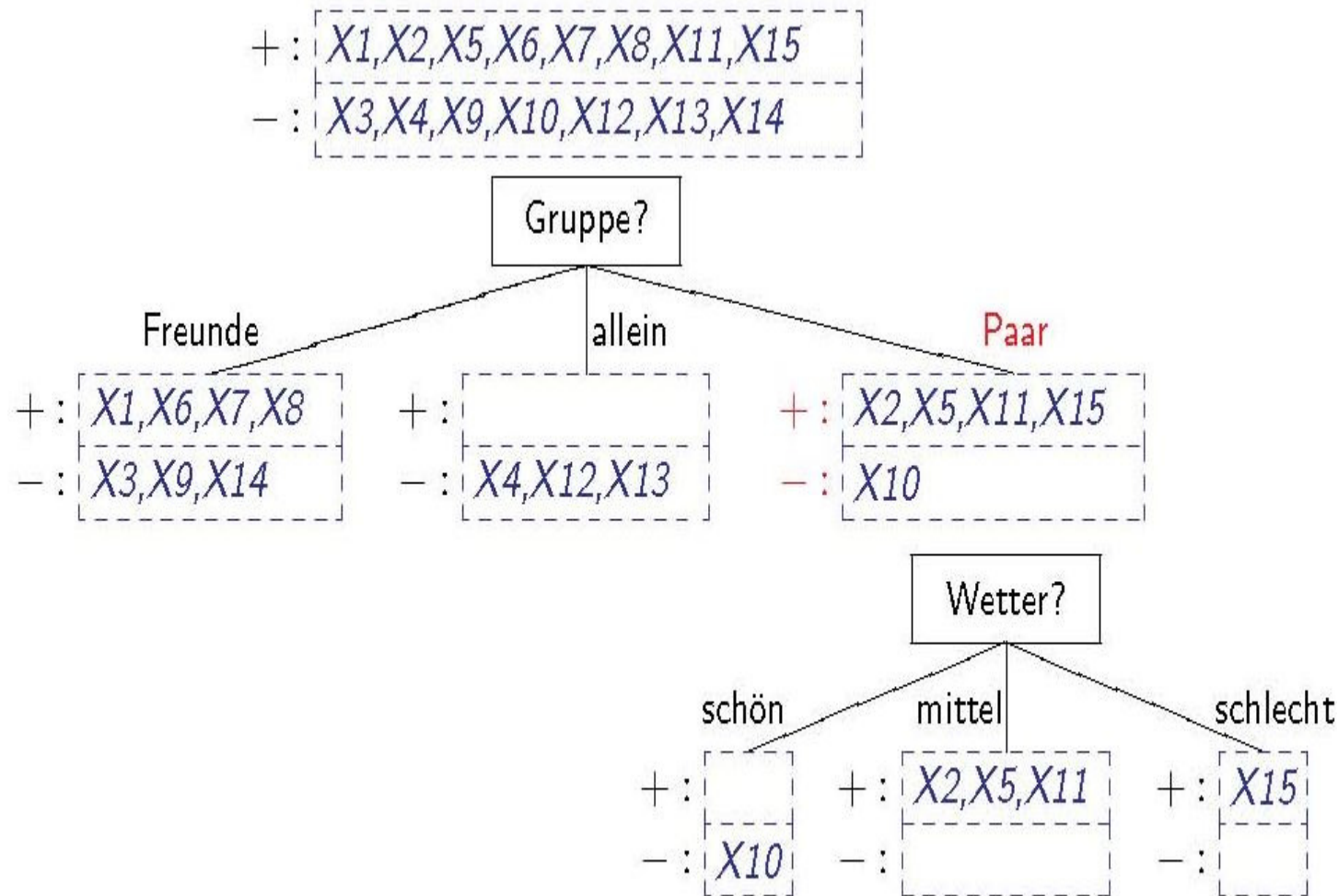
Wissenserwerb



Das Attribut *Kategorie* kann kein einziges Trainingsbeispiel mit nur einem Test klassifizieren.

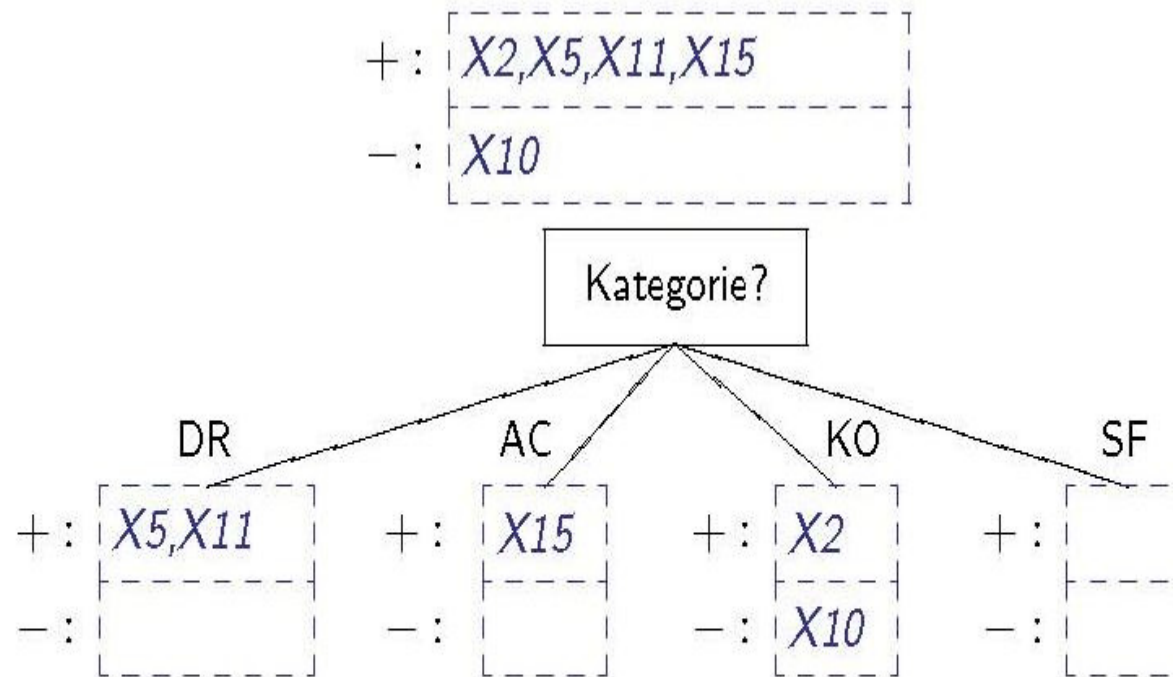
Gruppe ist also nach dem Kardinalitätskriterium als erstes Attribut des Entscheidungsbaumes besser geeignet als *Kategorie*.

Wissenserwerb



Wetter klassifiziert alle Beispiele der Menge $E_{Gruppe=Paar}$ vollständig.

Wissenserwerb



Kategorie kann hier zwei Beispiele nicht eindeutig klassifizieren.

Folglich ist *Wetter* als zweites Attribut an dieser Stelle besser geeignet.

Wissenserwerb

Die Wichtigkeit eines Attributes ist jedoch ein relativer Begriff – sie hängt stark von der aktuellen Beispielmenge ab!

Beispiel: Das Attribut *Kategorie* ist als erste Abfrage für die gesamte Beispielmenge $\{X_1, \dots, X_{15}\}$ ziemlich nutzlos – andererseits hat gerade dieses Attribut für die Beispielmenge

$$E_{Gruppe=Freunde} = \{X_1, X_3, X_6, X_7, X_8, X_9, X_{14}\}$$

unter allen übrig gebliebenen Attributen die größte Wichtigkeit (es klassifiziert 5 Beispiele vollständig).

Wissenserwerb

function $DT(E, A, default)$

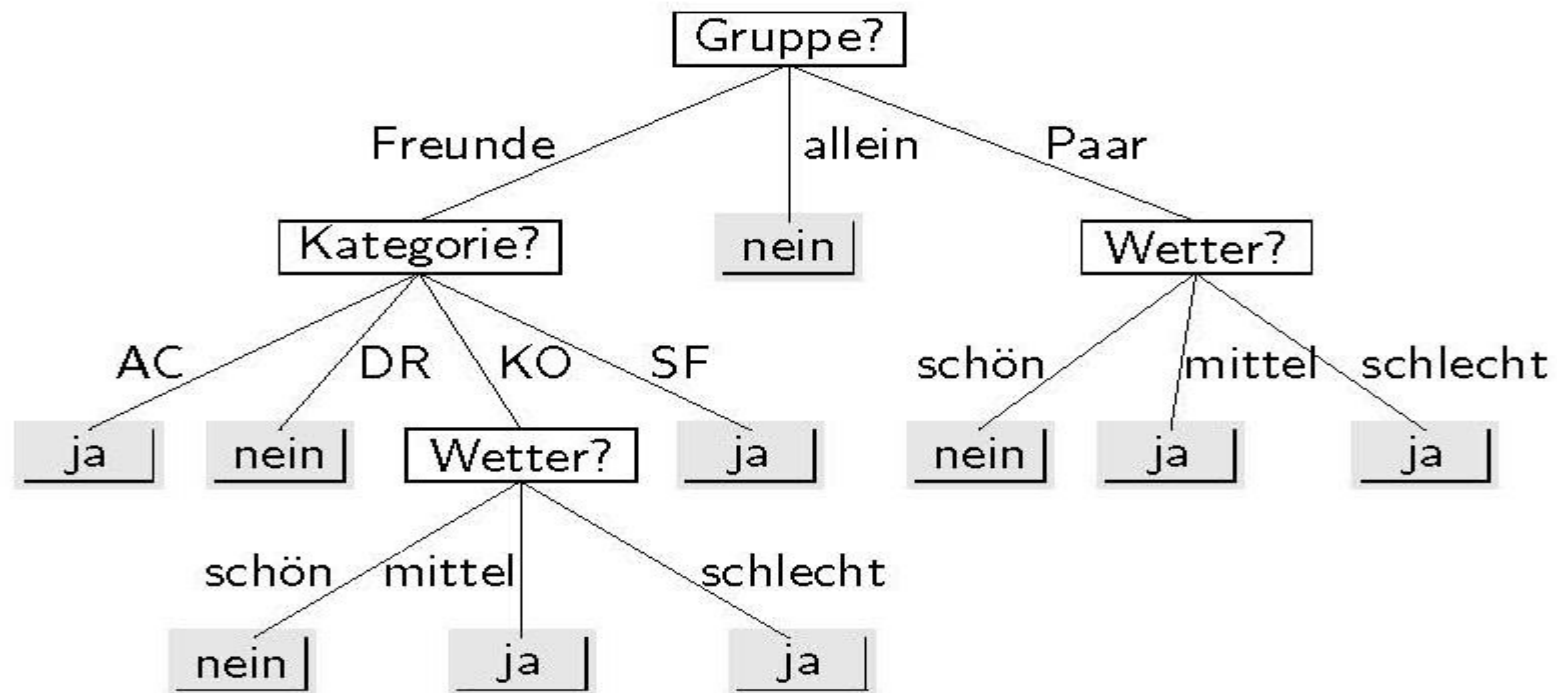
Eingabe: E Menge von Beispielen
 A Menge von Attributen
 $default$ Default-Klassifikation

Ausgabe: Entscheidungsbaum

Wissenserwerb

```
if  $E = \emptyset$ 
  then return default
else if alle Elemente in  $E$  haben die gleiche Klassifikation  $c \in \{ja, nein\}$ 
  then return  $c$ 
  else if  $A = \emptyset$ 
    then Fehler "gleiche Beispiele mit unterschiedlicher
           Klassifikation"
  else  $a := ChooseAttribute(A, E)$ 
        $T :=$  neuer Entscheidungsbaum
           mit Wurzelmarkierung  $a$ 
       for each Attributwert  $w_i$  von  $a$  do
          $E_i := \{e \in E \mid a(e) = w_i\}$ 
          $T_i := DT(E_i, A \setminus \{a\}, MajorityVal(E))$ 
         hänge an den Wurzelknoten von  $T$ 
         eine neue Kante mit Markierung  $w_i$ 
         und Unterbaum  $T_i$  an
       end
  return  $T$ 
```

Wissenserwerb



Der fertige Entscheidungsbaum

Wissenserwerb

Der Kern eines TDIDT-Verfahrens ist die Attributauswahl mit dem Ziel, den Entscheidungsbaum möglichst klein zu halten.

Die etabliertesten TDIDT-Systeme sind ID3 und seine Weiterentwicklung C4.5, die beide auf Quinlan zurückgehen.

Zur Bestimmung des jeweils besten Attributes verwenden diese beiden Systeme grundsätzlich den Informationsgehalt eines Attributes, der durch den jeweiligen Informationsgewinn bestimmt wird.

Wissenserwerb

Die Menge E der Beispiele enthalte p positive und n negative Beispiele; dann beträgt die Wahrscheinlichkeit, ein positives bzw. negatives Beispiel auszuwählen, $\frac{p}{p+n}$ bzw. $\frac{n}{p+n}$.

Der Informationsgehalt $I(E)$ der Antwort auf die Frage

"Handelt es sich bei einem beliebigen Beispiel aus E um ein positives oder ein negatives Beispiel ?"

beträgt daher

$$\begin{aligned} I(E) &:= H\left(\frac{p}{p+n}; \frac{n}{p+n}\right) \\ &= -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n} \quad [bit] \end{aligned}$$

Wissenserwerb

Beispiel: Bei der Trainingsmenge des Kinoproblems beträgt dieser Informationsgehalt

$$H\left(\frac{p}{p+n}; \frac{n}{p+n}\right) = H\left(\frac{8}{15}; \frac{7}{15}\right) \approx 0.9968 \text{ bit}$$

Bei der Attributauswahl soll nun berücksichtigt werden, welchen Informationsgewinn man erhält, wenn man den Wert eines Attributs kennt.

Wissenserwerb

Informell errechnet sich also der Informationsgewinn als Differenz

$$\begin{aligned} \text{Informationsgewinn} = & \\ & \text{Informationsgehalt vor Attributauswahl} \\ & - \text{Informationsgehalt nach Attributauswahl} \end{aligned}$$

wobei der Informationsgehalt einer Trainingsmenge als die Entropie der zugehörigen Verteilung definiert wird.

Wissenserwerb

Jedes Attribut a teilt die Trainingsmenge E in Teilmengen $E_1, \dots, E_k$ auf, wobei jedes E_i die zum Wert w_i von a gehörigen Beispiele enthält und k die Anzahl der verschiedenen Werte $w_1, \dots, w_k$ ist, die a annehmen kann.

Jedse E_i habe p_i positive und n_i negative Beispiele.

Wenn wir $a = w_i$ wissen, benötigen wir also noch

$$I(E_i) = H\left(\frac{p_i}{p_i + n_i}; \frac{n_i}{p_i + n_i}\right) \text{ bit}$$

an Information.

Wissenserwerb

Durch die Berücksichtigung aller Attributwerte $a = w_i$ ergibt sich nach dem Test des Attributes a die folgende bedingte mittlere Information:

$$\begin{aligned} I(E \mid a \text{ bekannt}) &= \sum_{i=1}^k P(a = w_i) \cdot I(E_i) \\ &= \sum_{i=1}^k \frac{p_i + n_i}{p + n} \cdot H\left(\frac{p_i}{p_i + n_i}; \frac{n_i}{p_i + n_i}\right) \end{aligned}$$

Wissenserwerb

Der Informationsgewinn des Attributes a ist nun :

$$\text{gain}(a) = I(E) - I(E \mid a \text{ bekannt})$$

(ist also nichts anderes als die gegenseitige Information zwischen dem Attribut a und der Ausprägung *positiv/negativ*.)

Das Lernsystem ID3 wählt als nächstes zu testendes Attribut a dasjenige aus, bei dem $\text{gain}(a)$ maximal ist.

Wissenserwerb

Der Informationsgewinn des Attributes a ist nun :

$$\text{gain}(a) = I(E) - I(E \mid a \text{ bekannt})$$

(ist also nichts anderes als die gegenseitige Information zwischen dem Attribut a und der Ausprägung *positiv/negativ*.)

Das Lernsystem ID3 wählt als nächstes zu testendes Attribut a dasjenige aus, bei dem $\text{gain}(a)$ maximal ist.

Wissenserwerb

Für das Kinoproblem errechnet man für die Wurzel des Entscheidungsbaumes:

$$\begin{aligned} \text{gain}(\text{Gruppe}) &= I(E) - I(E \mid \text{Gruppe bekannt}) \\ &\approx 0.9968 - \left[\frac{7}{15} H\left(\frac{4}{7}; \frac{3}{7}\right) + \frac{5}{15} H\left(\frac{4}{5}; \frac{1}{5}\right) + \frac{3}{15} H(1; 0) \right] \\ &\approx 0.2964 \text{ bit} \end{aligned}$$

$$\begin{aligned} \text{gain}(\text{Kategorie}) &= I(E) - I(E \mid \text{Kategorie bekannt}) \\ &\approx 0.9968 - \left[\frac{4}{15} H\left(\frac{3}{4}; \frac{1}{4}\right) + \frac{4}{15} H\left(\frac{1}{2}; \frac{1}{2}\right) + \frac{4}{15} H\left(\frac{1}{2}; \frac{1}{2}\right) \right. \\ &\quad \left. + \frac{3}{15} I\left(\frac{1}{3}; \frac{2}{3}\right) \right] \\ &\approx 0.0634 \text{ bit} \end{aligned}$$

Es zeigt sich, dass $\text{gain}(\text{Gruppe})$ maximal ist und daher von ID3 als erstes Attribut ausgewählt würde

Wissenserwerb

Der (absolute) Informationsgewinn $gain(a)$ hat den Nachteil, dass er Attribute mit zahlreichen Werten bevorzugt, was im Extremfall zu unsinnigen Ergebnissen führen kann.

Beispiel: Bei einer medizinischen Diagnose werde als eines der Attribute die persönliche Identifikationsnummer (PIN) eines Patienten benutzt. Dieses Attribut hat so viele Werte, n , wie es Patienten in der Datei gibt, und partitioniert daher die Beispielmenge, E , in eben so viele Teilmengen mit je einem Element.

In diesem Fall ist die bedingte mittlere Information

$$I(E \mid PIN \text{ bekannt}) = \sum_{i=1}^n \frac{1}{n} H(0; 1) = 0 \text{ bit}$$

der Informationsgewinn also maximal. Für die Diagnose selbst jedoch ist das Attribut PIN nutzlos.

Wissenserwerb

Das System C4.5 benutzt statt des absoluten Informationsgewinns einen normierten Informationsgewinn

$$\text{gain ratio}(a) = \frac{\text{gain}(a)}{\text{split info}(a)}$$

wobei $\text{split info}(a)$ die Entropie des Attributes a ist:

$$\text{split info}(a) = H(a) = - \sum_{i=1}^k P(a = w_i) \log_2 P(a = w_i)$$

Wissenserwerb

Beispiel (Forts.): Die durch PIN induzierte Verteilung ist eine Gleichverteilung ($k = n$ und $P(PIN = w_i) = \frac{1}{n}$), also ist der Normierungsfaktor $H(PIN) = \log_2 n$ und damit maximal. Damit ist

$$gain\ ratio(PIN) = \frac{I(E)}{\log_2 n},$$

der Informationsgewinn durch PIN ist also minimal.

Das System C4.5 wählt als nächstes Attribut dasjenige mit dem maximalen *gain ratio*-Wert aus (*gain ratio*-Kriterium).

Wissenserwerb

Aus Entscheidungsbäumen kann man direkt Regeln ableiten:

Jeder Pfad von der Wurzel zu einem Blattknoten entspricht einer **if-then**-Regel.

Kino-Beispiel: Aus dem Entscheidungsbaum des Kino-Beispiels kann man z.B. die folgenden Regeln gewinnen:

if *Gruppe = Paar* **and** *Wetter = schlecht* **then**
Kinobesuch = ja

if *Gruppe = Freunde* **and** *Kategorie = SF* **then**
Kinobesuch = ja

if *Gruppe = Freunde* **and** *Kategorie = Komödie* **and**
Wetter = schön **then** *Kinobesuch = nein*