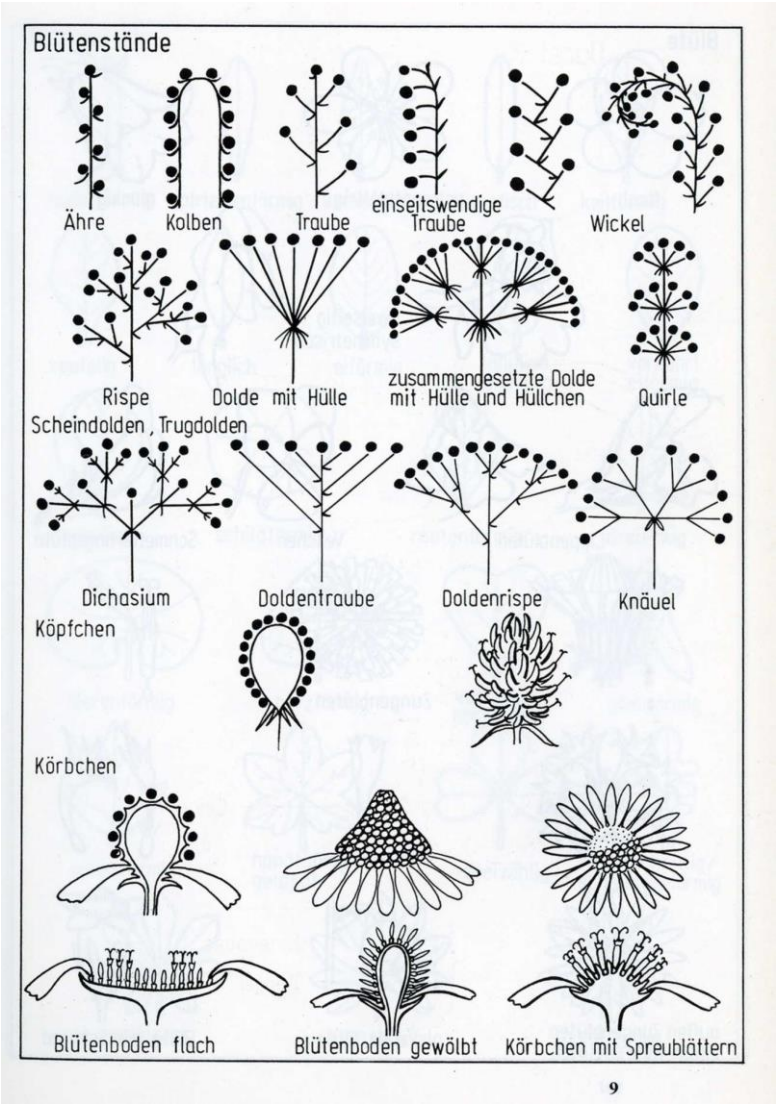
8. Klassifikation bei nominalen Merkmalen

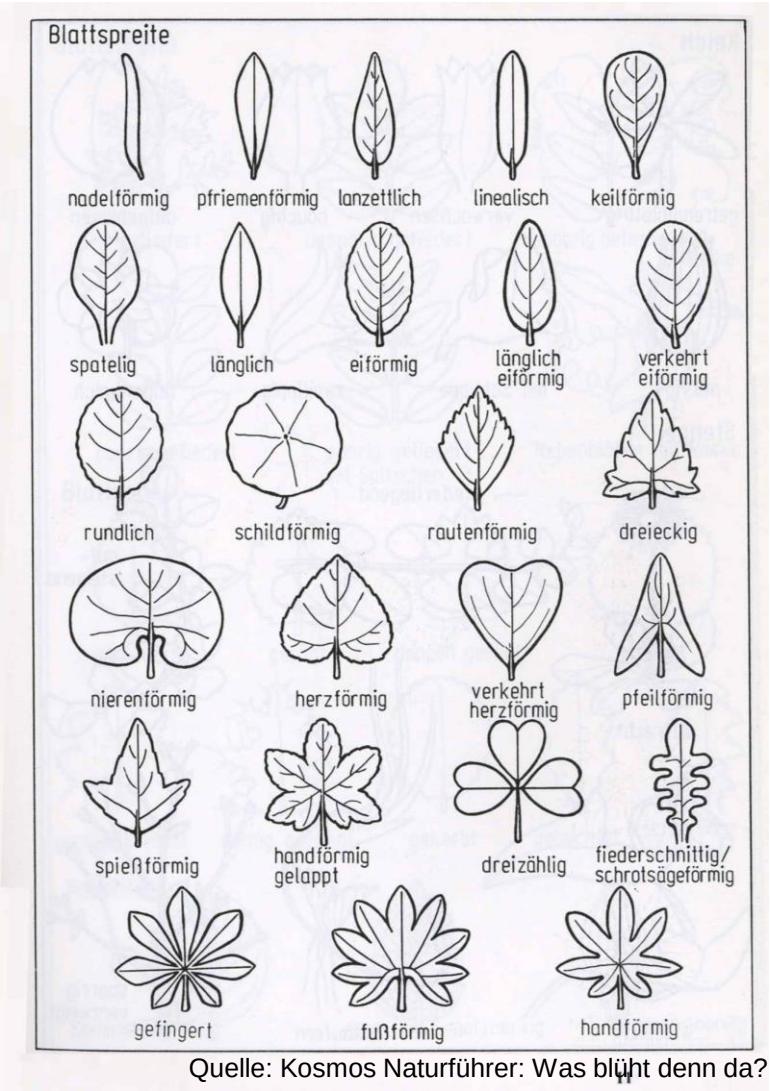
8. Klassifikation bei nominalen Merkmalen

	Skala				
	qualitativ		metrisch		
	Nominal-	Ordinal-	Intervall-	Verhältnis-	Absolut-
Empirische Relationen	~ Äquivalenz	~ Äquivalenz ≻Ordnung	~ Äquivalenz ≻Ordnung ⊕Emp. Addition	~ Äquivalenz ≻Ordnung ⊕Emp. Addition ⊗Emp. Multipl.	~ Äquivalenz ≻Ordnung ⊕Emp. Addition ⊗Emp. Multipl.
Zulässige Transfor-mationen	$m' = f(m)$ $f(.)$ bijektiv	$m' = f(m)$ $f(.)$ streng monoton	$m' = am + b$ mit $a > 0$	$m' = am$ mit $a > 0$	$m' = m$
Beispiele zugehörige Merkmale	Telefonnum., Kfz-Kennz., Typen, PLZ, Geschlecht	Güteklassen, Härtegrad, Windstärke	Temp. in C°, F°, Kalenderzeit, geographische Höhe	Masse, Länge, el. Strom,	Quantenzahlen, Teilchenanzahl, Fehlerzahl
Werte von m	Zahlen, Namen, Symbole	i.d.R. natürliche Zahlen	i.d.R. reelle Zahlen	i.d.R. reelle Zahlen > 0	i.d.R. natürliche Zahlen
Aussage-kraft	gering	→	→	→	hoch

8. Klassifikation bei nominalen Merkmalen

Beispiel: Bestimmung (Klassifikation) von Blütenpflanzen





Quelle: Kosmos Naturführer: Was blüht denn da?

8. Klassifikation bei nominalen Merkmalen

Nominale Merkmale:

- Eigenschaften ohne quantitative Bedeutung und ohne Ordnung
- Einzige zulässige binäre Relation: ~ Äquivalenzrelation
- Nominale Merkmale m können nur diskrete „Werte“ (Ausprägungen) annehmen.

Bayessche Klassifikation:

$$P(\omega \mid m) = \frac{P(m \mid \omega)P(\omega)}{P(m)} = \frac{P(m \mid \omega)P(\omega)}{\sum_{i=1}^{c} P(m \mid \omega_i)P(\omega_i)}$$

$$\omega \in \{\omega_1, \ldots, \omega_c\} = \Omega / \sim$$

Bayessche Methodik der Kapitel 3 und 4 ist ohne Einschränkungen auch im Falle nominaler Merkmale und im Falle von gemischten Merkmals-vektoren einsetzbar.

8. Klassifikation bei nominalen Merkmalen

Weitere Ansätze für die Klassifikation auf der Basis nominaler Merkmale:

- Entscheidungsbäume (Decision Trees)
- String Verfahren
- Grammatiken (Syntaktische Mustererkennung)

8.1. Entscheidungsbäume

Beispiel Pflanzenbestimmung: Eine Sequenz von Fragen nach unterschiedlichen nominalen Eigenschaften (Blattform, Blattstand, Blütenform, Blütenfarbe usw.) engt nach und nach die Klassenzugehörigkeit der zu bestimmenden Pflanze ein, bis schließlich die Zuweisung zu einer Klasse erfolgt.

Typisch hierbei: die als nächstes gestellte Frage hängt i.d.R. von der Antwort auf die vorher gestellte Frage ab.

Formal kann diese Fragetechnik durch Entscheidungsbäume beschrieben werden. → In diesem Sinne ist ein Pflanzenbestimmungsbuch ein „manueller“ Entscheidungsbaumklassifikator.

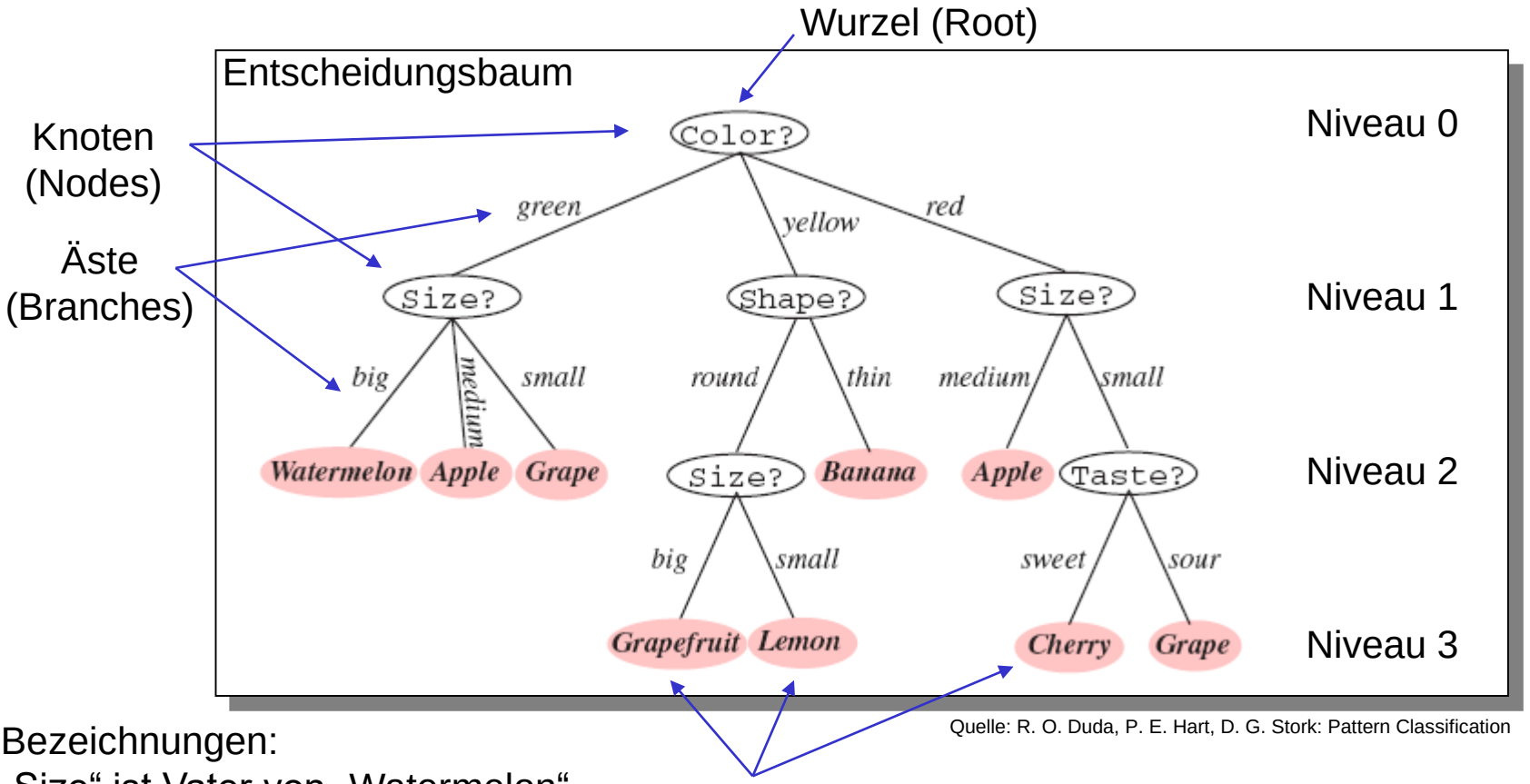
Weitere Beispiele: Ärztliche Diagnosen, Fehlersuchvorschriften für Geräte

8.1. Entscheidungsbäume

Beispiel: Klassifikation von Früchten

$$\omega \in \Omega / \sim \; = \{\omega_1, \ldots, \omega_7\} = \{\text{Apple, Watermelon, Grape, Grapefruit, Lemon, Cherry, Banana}\}$$

$$\mathbf{m} \in \mathbb{M} \subset \mathbb{M}_1 \times \mathbb{M}_2 \times \mathbb{M}_3 \times \mathbb{M}_4 =$$
$$\{\text{green, yellow, red}\} \times \{\text{big, medium, small}\} \times \{\text{round, thin}\} \times \{\text{sweet, sour}\}$$



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

Bezeichnungen:
„Size“ ist Vater von „Watermelon“.
„Watermelon“ ist Kind von „Size“.

8.1. Entscheidungsbäume

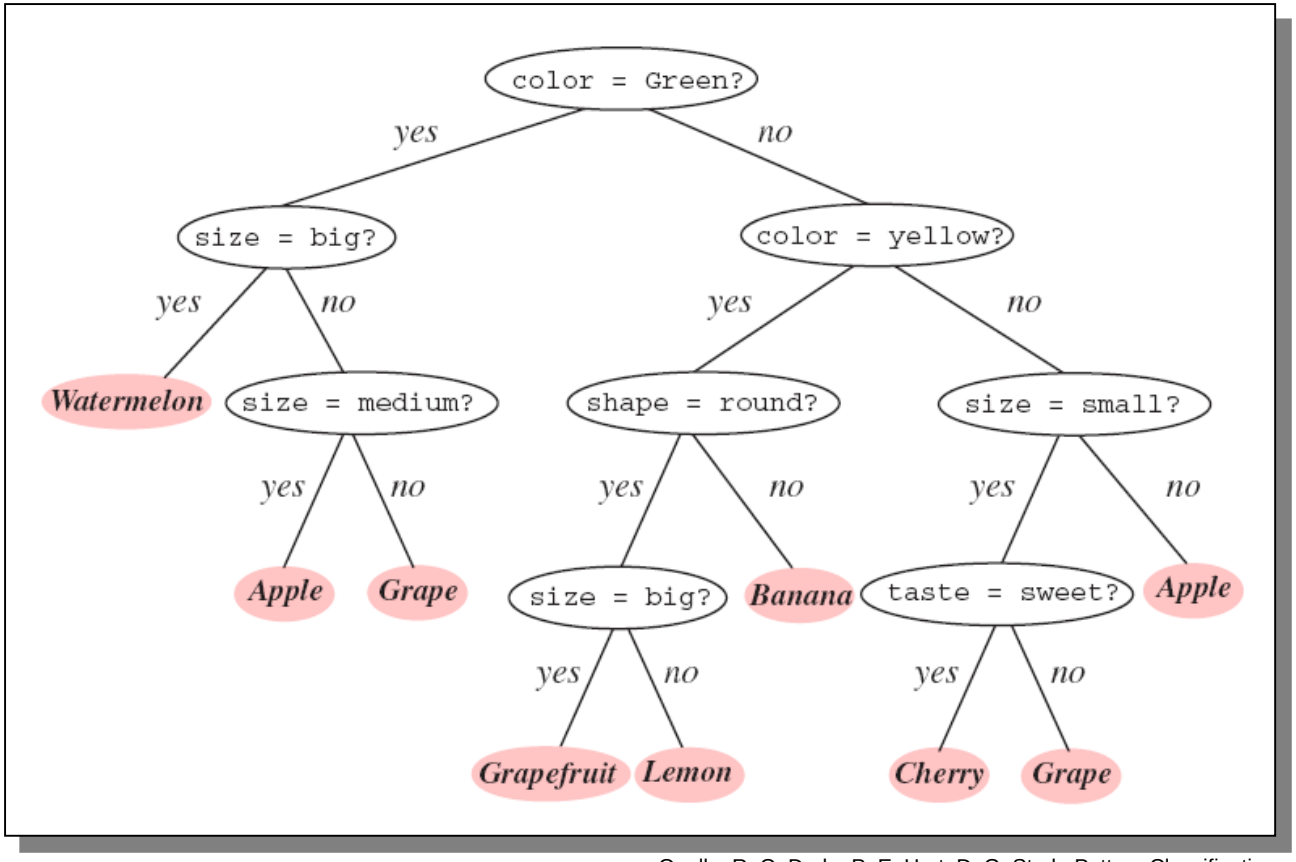
Bemerkungen:

- Entscheidungsbäume sind anschaulich interpretierbar.
- Äste eines Knotens müssen sich gegenseitig ausschließen und müssen erschöpfend sein. → Frage muss eindeutig beantwortet werden können.
- A Priori Wissen über die Mustererkennungsaufgabe kann sehr einfach eingebracht werden.
- Klassifikation geschieht durch sequentielle Entscheidungen entlang eines Pfades durch den Baum bis ein Blatt erreicht wird;
  Blätter stehen für Klassen.
- Gut strukturierte Entscheidungsbäume erlauben i.d.R. eine schnelle Klassifikation.
- Dieselbe Frage kann an mehreren Stellen im Baum auftreten.
- Dieselbe Frage kann an unterschiedlichen Plätzen im Baum unterschiedlich viele Äste (Antwortmöglichkeiten) haben.
- Mehrere Blätter können für dieselbe Klasse stehen.
- Entscheidungsbäume lassen sich auch auf Merkmale mit höherem Skalenniveau anwenden.

8.1. Entscheidungsbäume

Jeder Entscheidungsbaum kann in einen solchen mit binär entscheidbaren Fragen (Ja/Nein-Fragen) umgewandelt werden). Im Folgenden werden daher nur noch Bäume mit binären Entscheidungen betrachtet.

Beispiel: Klassifikation von Früchten



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

8.1. Entscheidungsbäume

Lernen eines Entscheidungsbaumes entspricht dem Aufbauen des Baumes anhand der Lernstichprobe $\mathbb{D}$.

Jede Verzweigung entspricht einem disjunkten Splitten einer Teilmenge der Lernstichprobe.

Overfitting (Überanpassung) im Kontext von Entscheidungsbäumen bedeutet eine zu feine Verzweigung des Baumes. Dann entfallen in der Lernphase auf Entscheidungen an Knoten und auf die Blätter zu wenige Elemente der Lernstichprobe.

8.1. Entscheidungsbäume

Wunsch: Einfacher kompakter Baum mit wenigen Knoten

Ansatz: Frage an jedem Knoten so stellen, dass die entstehenden Teilmengen der Lernstichprobe möglichst „sortenrein“ (*pure*) sind.

→ Verzweigungen anhand von Heterogenitätsmaßen festlegen.

Heterogenitätsmaße i (*impurity measures*):

Anforderungen:

i minimal bei vollständiger Konzentration auf eine Klasse

i maximal bei Gleichverteilung über alle Klassen

Heterogenitätsmaße i (impurity measures):

- Entropie-Heterogenitätsmaß:

$$i(n) = -\sum_j \hat{P}_n(\omega(\mathbf{m}) = \omega_j) \log_2 \left(\hat{P}_n(\omega(\mathbf{m}) = \omega_j) \right)$$

n: Nummer des Knotens

$$\hat{P}_n(\omega(\mathbf{m}) = \omega_j) := \frac{N_n^j}{N_n}$$

N_n : Umfang der Teilmenge von D am Knoten Nummer n

N_n^j : Umfang der Teilmenge von D der Klasse ω_j am Knoten Nummer n

- Gini-Heterogenitätsmaß: ist gleich dem Erwartungswert der Fehlerwahrscheinlichkeit, wenn die Klasse anhand der Verteilung am Knoten n zufällig festgelegt wird.

$$i(n) = \sum_{i \neq j} \hat{P}_n(\omega(\mathbf{m}) = \omega_i) \hat{P}_n(\omega(\mathbf{m}) = \omega_j) = 1 - \sum_j \hat{P}_n^2(\omega(\mathbf{m}) = \omega_j)$$

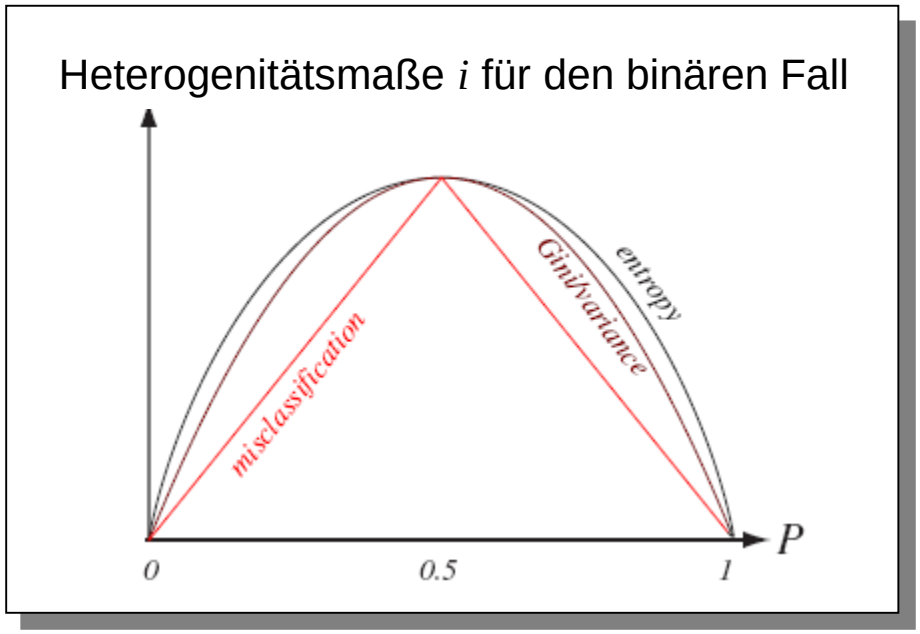
8.1. Entscheidungsbäume

Gini-Heterogenitätsmaß, binärer Fall: $c = 2$

$$i(n) = 2\hat{P}_n(\omega(\mathbf{m}) = \omega_1)\ \hat{P}_n(\omega(\mathbf{m}) = \omega_2)$$

- Fehlklassifikations-Heterogenitätsmaß: Fehlerwahrscheinlichkeit einer Mehrheitsentscheidung zugunsten der am Knoten n dominanten Klasse.

$$i(n) = 1 - \max_j \{\hat{P}_n(\omega(\mathbf{m}) = \omega_j)\}$$



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

8.1. Entscheidungsbäume

Vorgehensweise: Man stellt am Knoten n die Frage, welche die Heterogenität am meisten vermindert, die also

$$\Delta i(n) := i(n) - \hat{P}_{\text{Ja}}\, i(n_{\text{Ja}}) - \hat{P}_{\text{Nein}}\, i(n_{\text{Nein}})$$

maximiert.

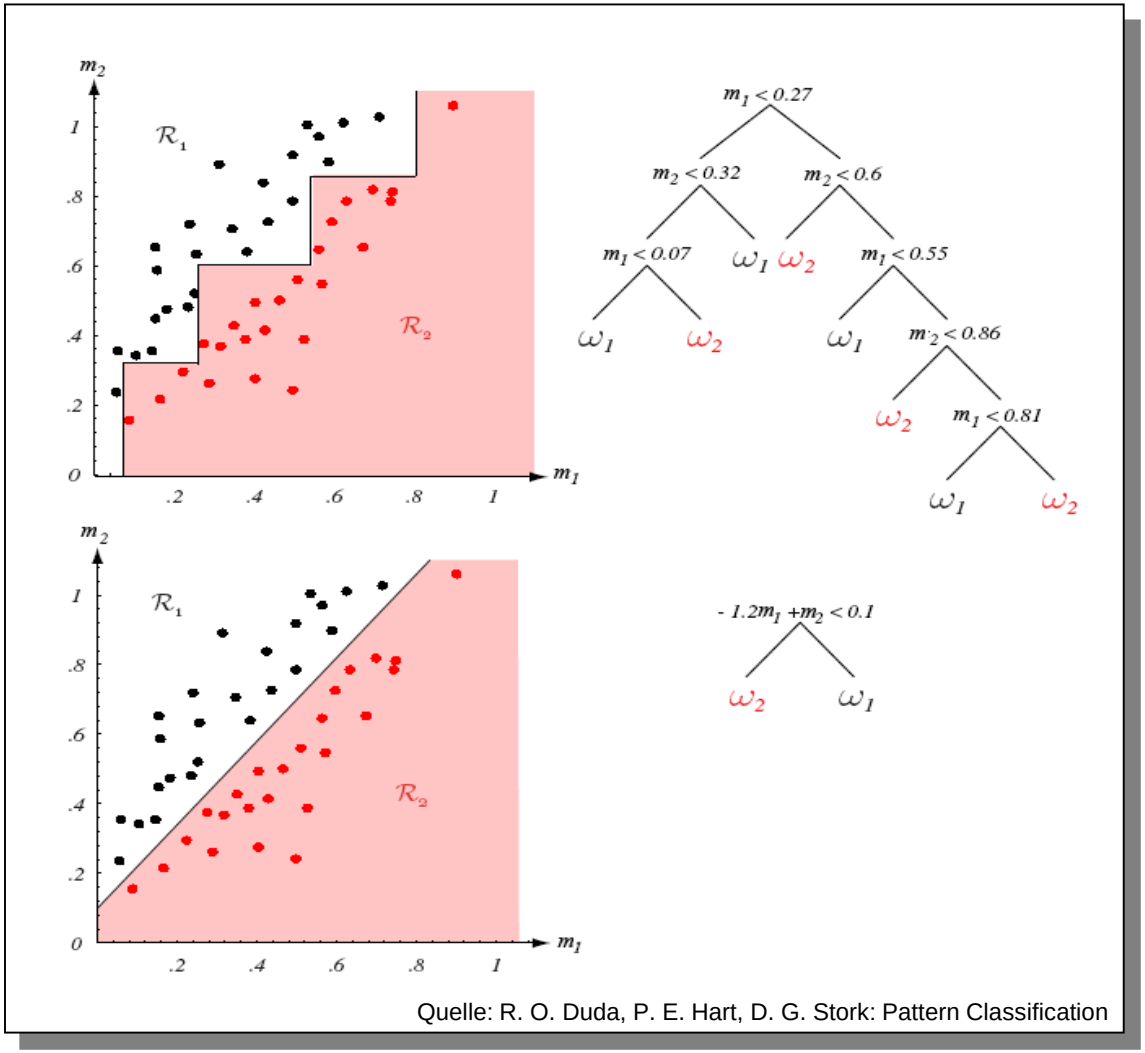
Abbruchbedingung: Man verzweigt so lange bis die Heterogenität einen vorgegebenen Schwellwert unterschreitet oder die Zahl der zur Verfügung stehenden Stichprobenelemente eine vorgegebene Anzahl unterschreitet.

Bemerkungen:

- Die Optimierung an den Verzweigungen ist nur lokal → keine Garantie, dass das globale Optimum gefunden wird.
- Bei nicht zu umfangreichen Aufgabenstellungen kann der optimale Entscheidungsbaum auch durch vollständige Suche gefunden werden.
- Bäume können noch nachbearbeitet werden, um ihre Eigenschaften zu verbessern (Merging, Pruning).

Einfluss der gewählten Merkmale:

Beispiel: quantitative Merkmale, $d = 2$, $c = 2$



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

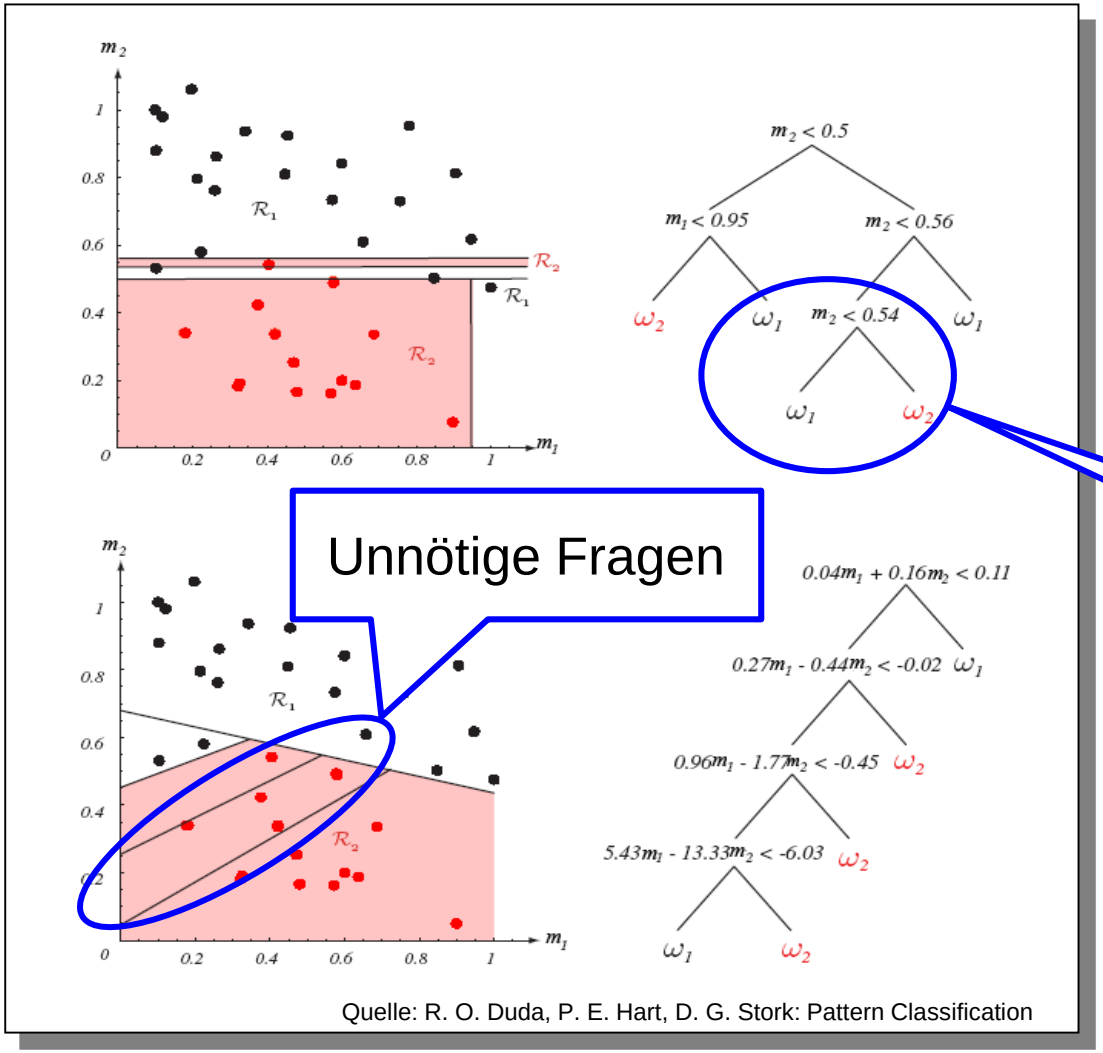
Merkmale m_1 und m_2 sind ungünstig.
Baum ist unnötig groß.

Komplexere Fragen können den Baum vereinfachen.

Entspricht hier einer vorgeschalteten Merkmalstransformation: $m' := -\frac{6}{5} m_1 + m_2$ und Schwellwertvergleich.

Einfluss der gewählten Merkmale:

Beispiel: quantitative Merkmale, $d = 2$, $c = 2$



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

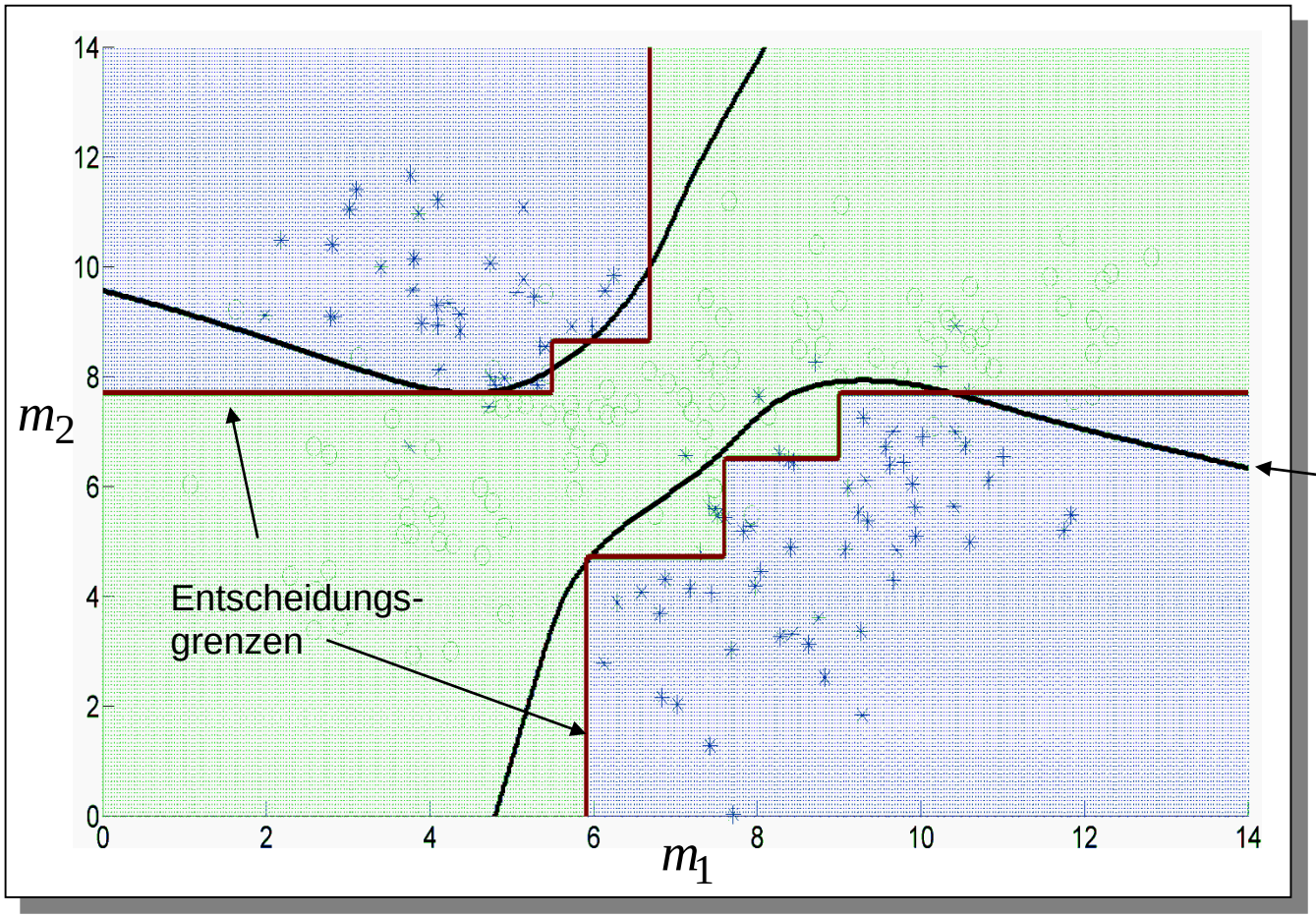
Entscheidungsbaum mit schlechten Generalisierungs-eigenschaften

Overfitting!

Unnötig komplizierter Entscheidungsbaum

8.1. Entscheidungsbäume

Beispiel: Entscheidungen im Baum an Trainingsdaten angepasst

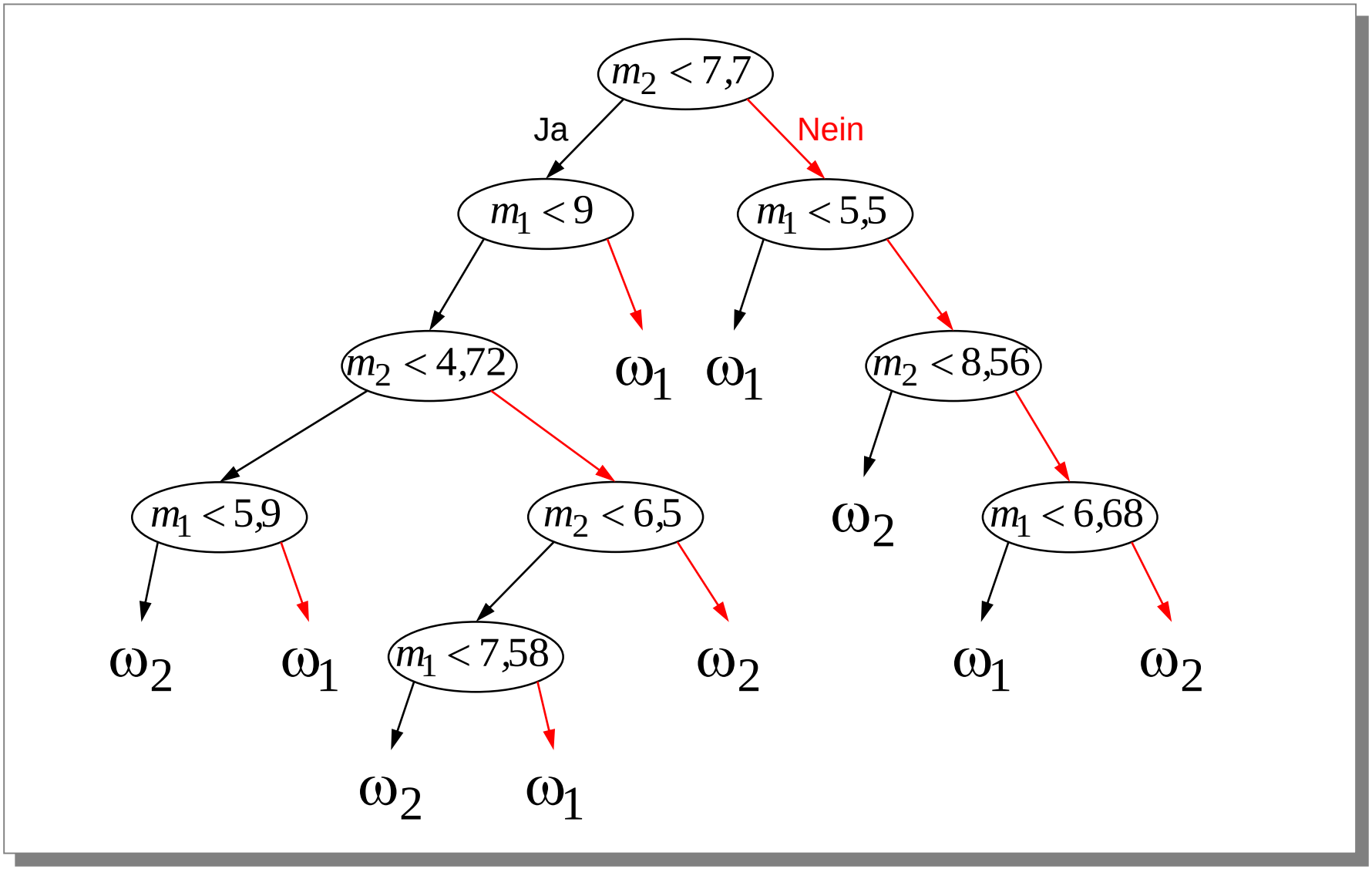


Parameter anhand der Lernstichprobe geschätzt. Testfehler = 11%

Asymptotischer Testfehler ≈ 13,55%

8.1. Entscheidungsbäume

Beispiel: Entscheidungen im Baum an Lernstichprobe angepasst



8.1. Entscheidungsbäume

Bemerkung:
Entscheidungsbäume können auch als Metaverfahren aufgefasst werden. An jedem Knoten kann ein beliebiger Klassifikator platziert werden, sodass der Baum Teilklassifikatoren zu einem Gesamtklassifikator zusammenschaltet.

8.2. Random Forests

Idee:

- Kombination mehrerer Klassifikatoren (hier: Entscheidungsbäume) zu einem Ensemble (hier: Wald).
- Ergebnis der Klassifikation $w(\mathbf{m})$ als gewichtete Summe der einzelnen Klassifikatoren $b_i(\mathbf{m})$:

$$w(\mathbf{m}) = \sum_{i=1}^{B} \alpha_i b_i(\mathbf{m})$$

Vorgehensweise bei sog. Random Forests:

- Aus einer Lernstichprobe D wird eine Menge von Entscheidungsbäumen aufgebaut: $\{b_i(\mathbf{m}): i = 1 \ldots B\}$.
- Bei der Klassifikation werden die Ergebnisse der einzelnen Bäume gleichgewichtet: ($\alpha_i = \frac{1}{B}$).
- Für das Training eines Entscheidungsbaumes $b_i(\mathbf{m})$ werden zufällig Merkmalsvektoren und Merkmale aus der Lernstichprobe ausgewählt.

L. Breiman, "Random Forests". Machine Learning 45 (1): 5–32, 2001.

Training:

Gegeben sei eine Lernstichprobe

$$\mathrm{D} = \{\mathbf{m}_1, ..., \mathbf{m}_N\} \text{ mit } \mathbf{m} := (m_1, ..., m_d)^{\mathrm{T}} \text{ und } \omega(\mathbf{m}_i) \text{ für } i = 1, ..., N \text{ bekannt.}$$

Für jeden Baum $b_i(\mathbf{m})$ des Waldes:

(1) Ziehe zufällig und mit Zurücklegen n Merkmalsvektoren aus D.
(2) Konstruiere aus dieser Stichprobe den Baum. Führe hierzu rekursiv die folgenden Schritte für jedes derzeitige Blatt des Baumes durch:

a) Wähle zufällig $d' < d$ Merkmale aus.
b) Wähle unter Berücksichtigung nur dieser d' Merkmale eine Verzweigung so, dass die Menge der Merkmalsvektoren möglichst „gut“ auf die zwei resultierenden Kinder aufgeteilt wird (Heterogenitätsmaß).

Abbruchbedingung: Die Mindestanzahl der Stichprobenelementen die in einem Blatt zur Verfügung stehen müssen wird erreicht.

T. Hastie, R. Tibshirani, J. Friedman: The Elements of Statistical Learning, S.587ff (mod.)

8.2. Random Forests

Klassifikation:

- Jeder Baum $b_i(\mathbf{m})$ des Waldes erhält den zu klassifizierenden Merkmalsvektor $\mathbf{m}$ und bildet diesen auf eine Klasse $\omega_j \in \Omega/\sim$ ab.
- Die Wahrscheinlichkeit für das Vorliegen einer Klasse ω_j ergibt sich aus:

$$\hat{P}(\omega(\mathbf{m}) = \omega_j) = \frac{1}{B}\sum_{i=1}^{B}\left[b_i(\mathbf{m}) = \omega_j \right]$$

Variante 1: Mehrheitsentscheid nach Breiman. Mit [·] als Prädikatabbildung, Spezialfall: $\delta_{ij} = [i = j]$.

$$\hat{P}(\omega(\mathbf{m}) = \omega_j) = \frac{1}{B}\sum_{i=1}^{B}\hat{P}_i(\omega(\mathbf{m}) = \omega_j)$$

Variante 2: Mittelung der geschätzten Wahrscheinlichkeiten über den Klassen nach Ho.

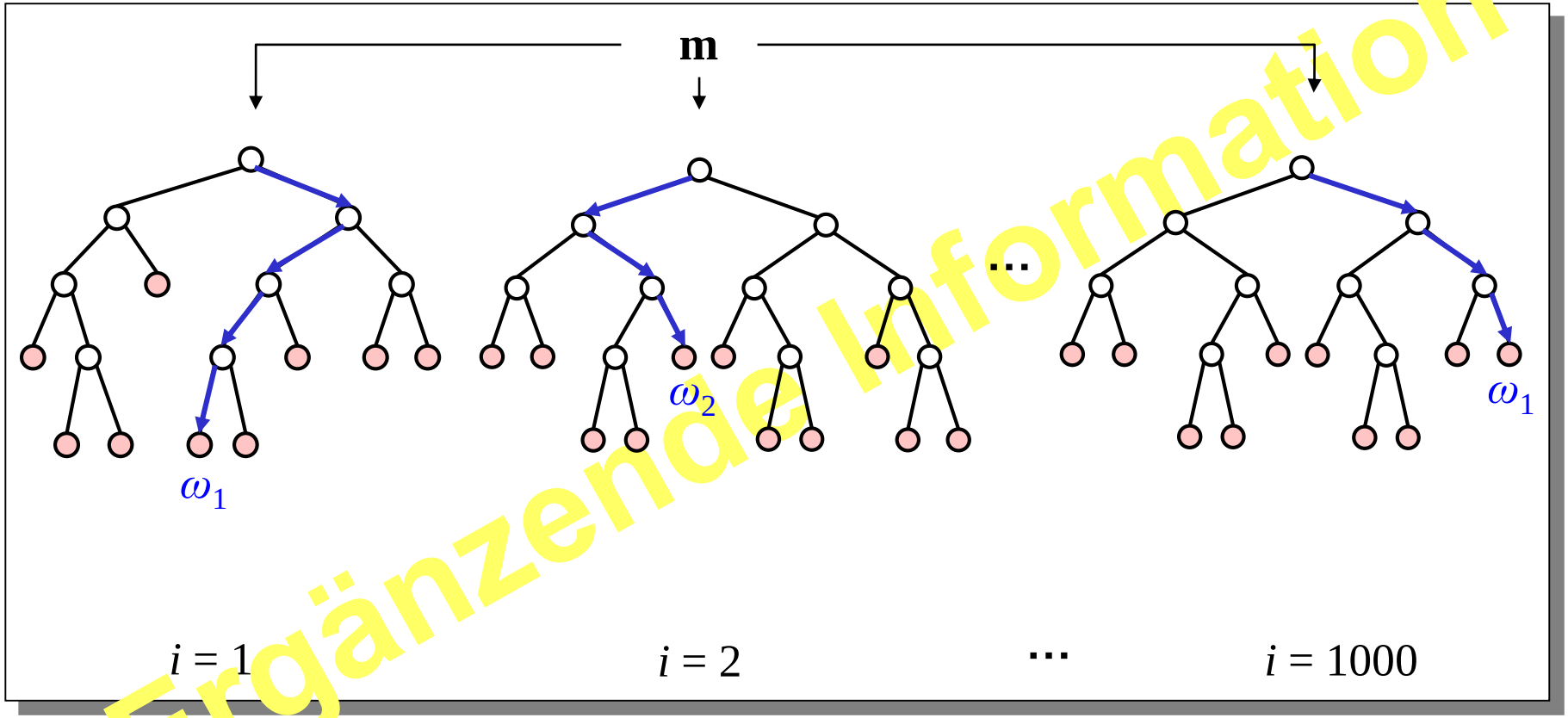
- Die geschätzte Klasse des Merkmalsvektors **m** ist gegeben durch:

$$\hat{\omega}(\mathbf{m}) = \arg\max_{\omega_j \in \Omega/\sim} \hat{P}\left(\omega(\mathbf{m}) = \omega_j\right)$$

L. Breiman, "Random Forests". Machine Learning 45 (1): 5–32, 2001.
T. Ho, “Random Decision Forests”, Proceedings of the 3rd International Conf. on Document Analysis and Recognition, Montreal, QC, 1995

Beispiel:

Mehrheitsentscheid mit $\{b_i(\mathbf{m}): i = 1 \ldots 1000\}$



$$\hat{P}(\omega(\mathbf{m}) = \omega_j) = \frac{1}{B}\sum_{i=1}^{B}\left[b_i(\mathbf{m}) = \omega_j \right]$$

Bemerkungen:

- Einlernen der Bäume ist i.d.R. schnell und simpel (es müssen nur $d' < d$ Merkmale in einer Verzweigung geprüft werden; häufig: $d' = \sqrt{d}$).
- Ensemble Methode: Gute Generalisierungsfähigkeit des Waldes gegenüber einem einzelnen Baum durch zufällige Wahl der Merkmale und Merkmalsvektoren beim Lernen.
- Klassifikation als auch Lernen sind leicht parallelisierbar.
- Random Forests können für Klassifikation, Regression und Cluster-analyse eingesetzt werden.
- Random Forests lassen sich auch auf Merkmale mit höherem Skalenniveau anwenden (vgl. Entscheidungsbäume).
- Speicherbedarf des Waldes kann u.U. groß sein.
- Spezialfall für $d' = d$: sog. „Bagging“ von Bäumen.

Details siehe z.B. T. Hastie, R. Tibshirani, J. Friedman: The Elements of Statistical Learning, p. 282ff, 587ff

- Bei der Klassifikation kann alternativ zum Mehrheitsentscheid auch über den Wahrscheinlichkeitsverteilungen der Klassen gemittelt werden.
- Die folgende Bezeichnungen werden häufig synonym verwendet: Randomized Decision Forests, Randomized Forests, Random Forests.

8.3. String Verfahren

Muster: Sequenz von diskreten Symbolen.

Begriffe erläutert am Beispiel von DNA-Sequenzen

Sequenz: AGCTTCGAATC

Alphabet: $\mathcal{A}$ = {A, G, C, T} A: Adenin, G: Guanin, C: Cytosin, T: Thymin

Symbole: A, G, C, T

Text: lange Sequenzen

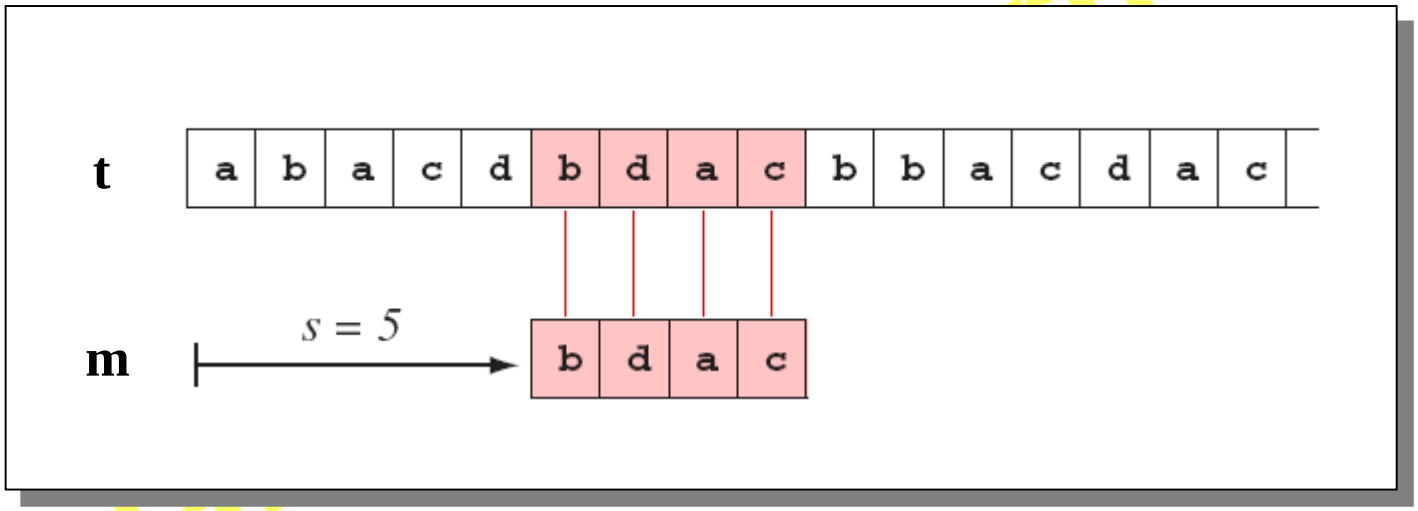
Faktor: zusammenhängender Teil einer Sequenz

Die Symbole entsprechen nominalen oder ordinalen Merkmalen.

Aufgabenstellung String Matching

Gegeben ist ein Text **t** und eine Sequenz **m**, wobei der Text i.d.R. viel länger ist als die Sequenz.

Frage: Ist die Sequenz **m** ein Faktor des Textes **t** und wo liegt sie?



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

Nächster-Nachbar-Klassifikation mit einem Distanzmaß

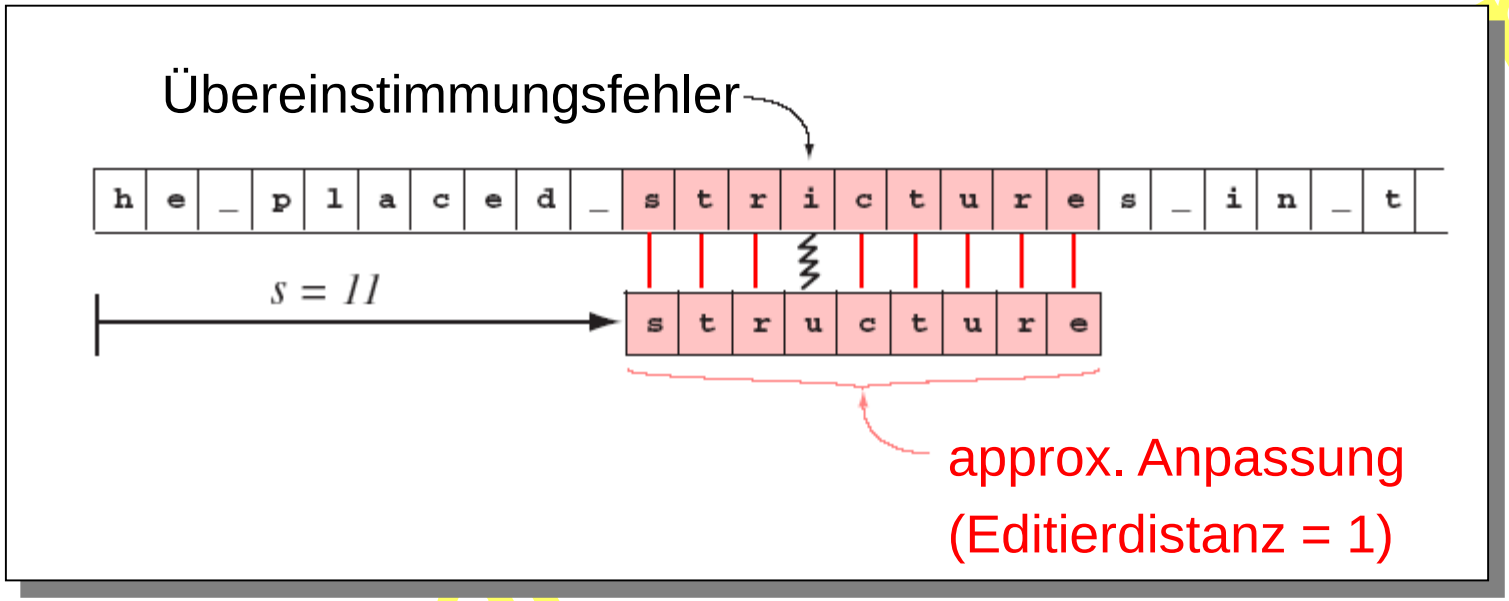
$\mathbf{m}_1, \mathbf{m}_2$ Sequenzen

Bsp.: Editier-Distanz: minimale Anzahl der Operationen (Einfügen, Entfernen, Austauschen) um $\mathbf{m}_1$ in $\mathbf{m}_2$ zu verwandeln.

Idee:

- Beim Training werden alle gegebenen Strings (Faktoren) mit ihren Klassenzugehörigkeiten gespeichert.
- Beim Klassifizieren werden für den unbekannten String alle „Distanzen“ zu den Strings der Lernstichprobe berechnet.
- Die Klasse des Strings mit der minimalen Distanz wird zugewiesen.

Näherungsweises String Matching



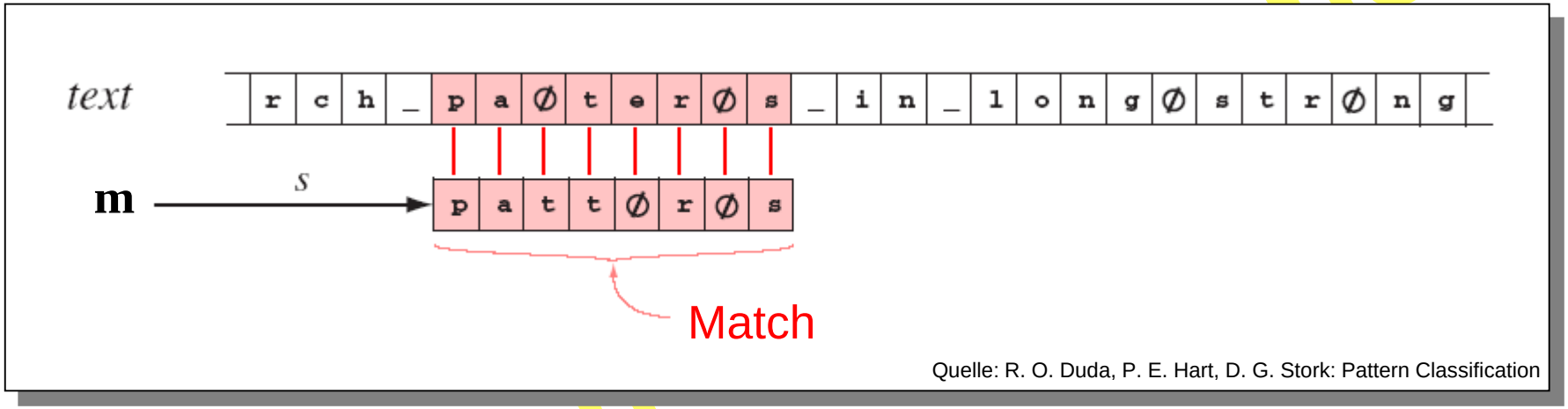
Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

m Sequenz, **t** Text

Gesucht: die Stelle im Text, für die die Editierdistanz zwischen **m** und einen Faktor von **t** minimal ist.

String Matching mit *don‘t care*-Symbolen

Ansatz: das don‘t care-Symbol ∅ stimmt definitionsgemäß mit jedem anderen Symbol überein (Funktion eines Jokers).



Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification

m Sequenz, **t** Text

Gesucht: die Stelle im Text, für die der Editierdistanz zwischen **m** und einen Faktor von **t** minimal ist.

Mustererkennung mit Hilfe von Grammatiken:

Lernen: Jede Klasse wird durch eine Grammatik $G_i \quad i = 1,\ldots, c$ repräsentiert.

→ Alle Sequenzen der von einer Grammatik G_i erzeugten Sprache sind äquivalent. Grammatiken $\hat{=}$ Modellen der Muster einer Klasse.

Klassifizieren: Ein zu klassifizierendes Muster wird der Klasse zugeordnet, deren Grammatik es erzeugt. → Klassifikation durch Parsen

$G = (A, V, S, P)$ **Grammatik**

$A = \{a, b, c\}$ Alphabet, Terminalsymbole

$V = \{A, B, C, S\}$ Variablen

S Startvariable

$$P = \begin{cases} p_1: & S \to AB & \text{oder} & BC \\ p_2: & A \to BA & \text{oder} & a \\ p_3: & B \to CC & \text{oder} & b \\ p_4: & C \to AB & \text{oder} & a \end{cases}$$ Regeln, Produktionen

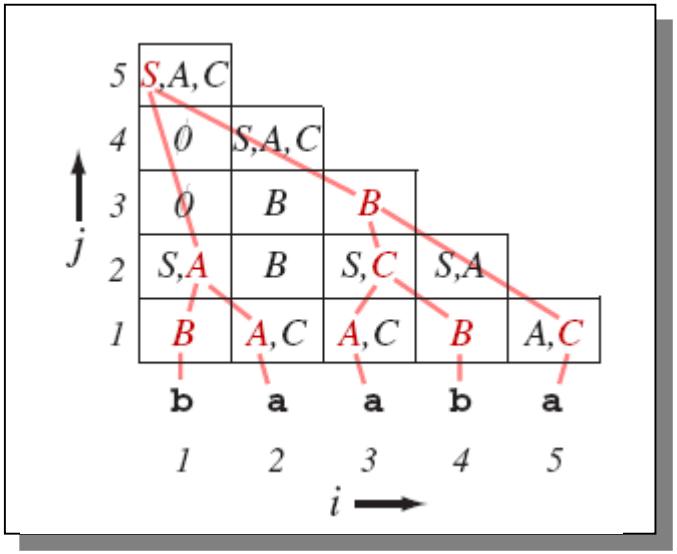
Beispiel

$L(G)$: Sprache := die Menge aller erzeugbaren Sequenzen

8.4. Grammatiken

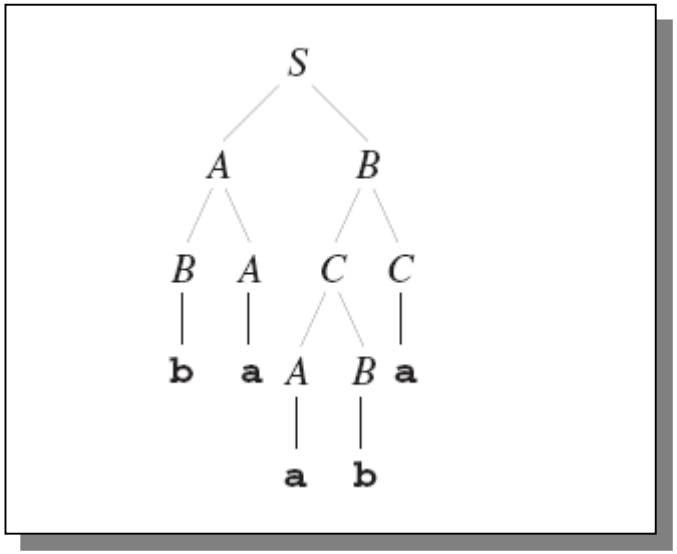
Bsp.: Bottom-Up Parsen

Sequenz ⟶ Startzustand



Bsp.: Top-Down Parsen

Startzustand ⟶ Sequenz



Resultat: „baaba“ gehört zur Sprache $L(\mathrm{G})$.

Lernen: Zu jeder Klasse muss aus den gegebenen Daten D eine Grammatik konstruiert werden.

Problem: Zu einer endlichen Zahl von Beispielen gibt es i.d.R. unendlich viele Grammatiken, die mit diesen konsistent sind.

Ansatz: Ockhams Rasiermesser: man wähle die einfachste, mit den Daten konsistente Grammatik aus.

Quelle: R. O. Duda, P. E. Hart, D. G. Stork: Pattern Classification