Vorlesungsplan

- 17.10. Einleitung
- 24.10. Ein- und Ausgabe
- 31.10. Reformationstag, Einfache Regeln
- **7.11. Naïve Bayes, Entscheidungsbäume**
- 14.11. Entscheidungsregeln, Assoziationsregeln
- 21.11. Lineare Modelle, Instanzbasiertes Lernen

Statistische Modellierung

- “Gegenteil” von 1R: nutzt alle Attribute
- Zwei Annahmen: Attribute sind
  - *gleich wichtig*
  - *statistisch unabhängig* (gegeben die Klasse)
    - wenn man den Wert eines Attributes weiß, kann man nichts über den Wert eines anderen Attributes sagen (die Klasse wird als bekannt vorausgesetzt)
- Unabhängigkeitsannahme ist nie korrekt!
- Aber, die Methode funktioniert gut in der Praxis

Wahrscheinlichkeiten für Wetterdaten

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes

Outlook	Temp	Humidity	Windy	Play
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Bayesche Regel

- Wahrscheinlichkeit eines Ereignis *H* gegeben eine Beobachtung *E* :

$$\Pr[H \mid E] = \frac{\Pr[E \mid H]\Pr[H]}{\Pr[E]}$$

- *A priori* Whr. von *H* : $\Pr[H]$
  - Wahrscheinlichkeit des Ereignis *vor* Beobachtung
- *A posteriori* Whr. von *H* : $\Pr[H \mid E]$
  - Wahrscheinlichkeit *nach* Beobachtung

Thomas Bayes

Born: 1702 in London, England
Died: 1761 in Tunbridge Wells, Kent, England

Naïve Bayes für Klassifikation

- Klassifikationslernen: Whr. einer Klasse gegeben eine Instanz?
  - Beobachtung *E* = Instanz
  - Ereignis *H* = Klassenwert für Instanz
- Naïve Annahme: Beobachtung wird in Teile zergliedert (Attribute) die *unabhängig* sind

$$\Pr[H \mid E] = \frac{\Pr[E_1 \mid H]\Pr[E_2 \mid H]\ldots\Pr[E_n \mid H]\Pr[H]}{\Pr[E]}$$

Problem “Häufigkeit ist Null”

- Was tun wenn ein Attributwert nicht mit jeder Klasse auftritt? (z.B. “Humidity = high” für Klasse “yes”)
  - Whr. ist null!
    $$\Pr[Humidity = High \mid yes] = 0$$
  - *A posteriori* Whr. würde auch null sein!
    (Unabhängig wie wahrscheinlich die anderen Werte sind!)
    $$\Pr[yes \mid E] = 0$$
- Ausweg (*Laplace Schätzer)* : addiere 1 zu jeder Anzahl aller Attributwert-Kombinationen (Zähler) und die Anzahl der möglichen Werte zum Nenner.
- Ergebnis: Whr. sind nie Null!

Fehlende Werte

- **Training:** Instanz wird für die Häufigkeit der Attributwert-Klassen Kombination nicht mitgezählt
- **Klassifikation:** Attribut wird aus der Berechnung weggelassen
- Beispiel:

Outlook	Temp.	Humidity	Windy	Play
?	Cool	High	True	?

Likelihood of "yes" = $3/9 \times 3/9 \times 3/9 \times 9/14 = 0.0238$

Likelihood of "no" = $1/5 \times 4/5 \times 3/5 \times 5/14 = 0.0343$

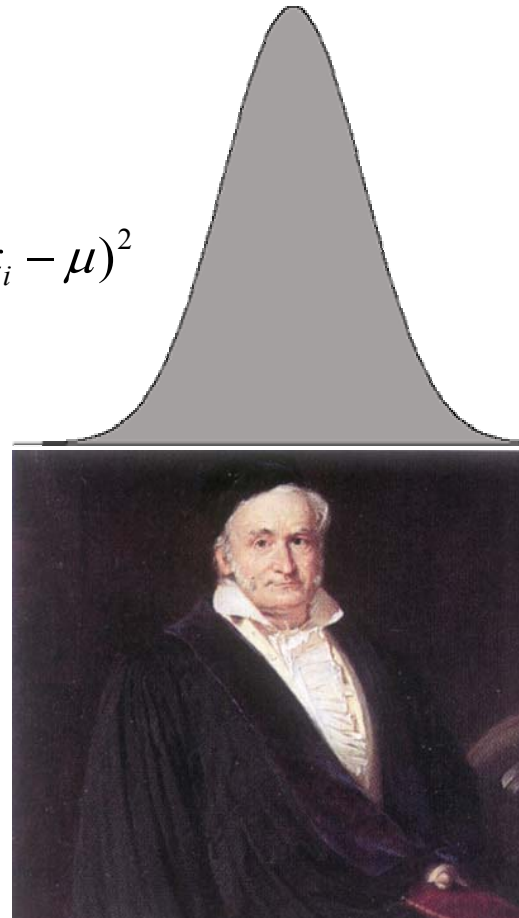
P("yes") = 0.0238 / (0.0238 + 0.0343) = 41%

P("no") = 0.0343 / (0.0238 + 0.0343) = 59%

Nummerische Attribute

- Mögliche Annahme: Attribut ist *normalverteilt* (gegeben die Klasse)
- Whr. Dichtefkt. für Normalverteilung ist definitiert durch zwei Parameter:
  - *Sample mean* μ $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$
  - *Standard deviation* σ $\sigma = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \mu)^2$
  - Dichtefkt. *f(x)* ist

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



Statistik für die Wetterdaten

Outlook			Temperature		Humidity		Windy			Play	
	Yes	*No*	*Yes*	*No*	*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	64, 68,	65, 71,	65, 70,	70, 85,	False	6	2	9	5
Overcast	4	0	69, 70,	72, 80,	70, 75,	90, 91,	True	3	3		
Rainy	3	2	72, …	85, …	80, …	95, …					
Sunny	2/9	3/5	μ =73	μ =75	μ =79	μ =86	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	σ =6.2	σ =7.9	σ =10.2	σ =9.7	True	3/9	3/5		
Rainy	3/9	2/5									

- Bespiel Dichtewert:

$$f(temperature = 66 \mid yes) = \frac{1}{\sqrt{2\pi}\,6.2} e^{-\frac{(66-73)^2}{2*6.2^2}} = 0.0340$$

Klassifizierung eines neuen Tages

- Neuer Tag:

Outlook	Temp.	Humidity	Windy	Play
Sunny	66	90	true	?

Likelihood of “yes” = $2/9 \times 0.0340 \times 0.0221 \times 3/9 \times 9/14 = 0.000036$

Likelihood of “no” = $3/5 \times 0.0291 \times 0.0380 \times 3/5 \times 5/14 = 0.000136$

P(“yes”) = 0.000036 / (0.000036 + 0. 000136) = 20.9%

P(“no”) = 0.000136 / (0.000036 + 0. 000136) = 79.1%

- Fehlende Werte werden beim Training in der Berechnung von Mittelwert und Standardabweichung nicht berücksichtigt

Wahrscheinlichkeitsdichten

- Beziehung zwischen Wahrscheinlichkeit & Dichte:

$$\Pr[c - \frac{\varepsilon}{2} < x < c + \frac{\varepsilon}{2}] \approx \varepsilon * f(c)$$

- Aber: dies ändert nicht die Berechnung der *posteriori* Whr. beim Naïve Bayes Klassifikator weil ε herausgekürzt wird
- Exakte Beziehung:

$$\Pr[a \leq x \leq b] = \int_a^b f(t)dt$$

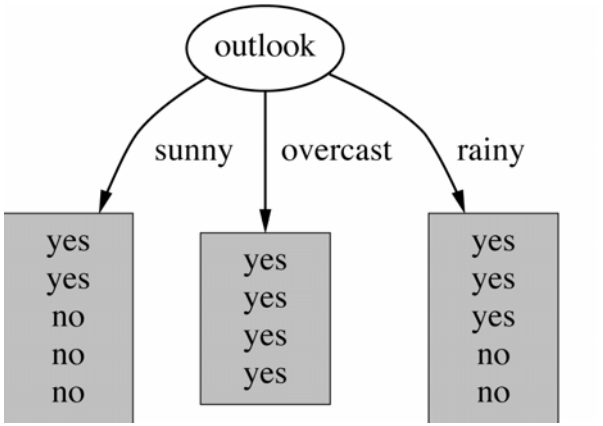
Naïve Bayes: Diskussion

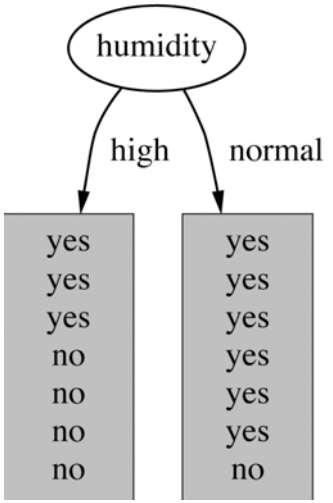
- Naïve Bayes fkt. oft überraschend gut (auch wenn die Unabhängigkeitsannahme verletzt ist)
- Warum? Für Klassifikation werden keine akkuraten Whr. gebraucht solange wie die max. Whr. zur korrekten Klasse zugeordnet wird
- Aber: Hinzufügen vieler redundanter Attribute macht Probleme (z.B. identische Attribute)
- Bemerkung: viele nummerischen Attribute sind nicht normalverteilt
  ($\rightarrow$ Histogramme o. Kernel Density Estimators)

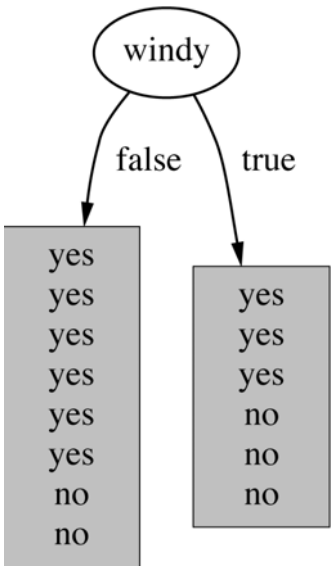
Konstruktion von Entscheidungsbäumen

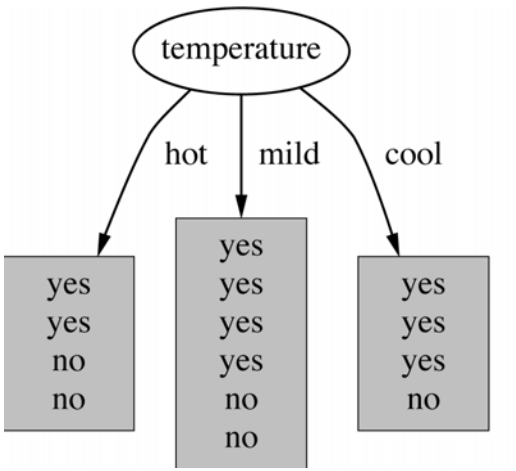
- Strategie: top down
  Rekursives *Teilen und Herrschen*
  - Wähle ein Attribut als Wurzelknoten root
    Erzeuge Kindknoten für jeden mögl. Attributwert
  - Teile Instanzen in Untermengen,
    eine für jeden Kindknoten
  - Setze den Prozeß rekursiv fort für jeden Kindknoten
- Stop, falls alle Instanzen dieselbe Klasse haben

Welches Attribut wählen?









Kriterium zur Attributauswahl

- Welches ist das beste Attribut?
  - kleine Bäume sind bevorzugt
  - Heuristik: wähle Attribut welches die “reinsten” Knoten erzeugt
- Populäres *Reinheitskriterium*: *Informationzuwachs*
  - Informationszuwachs steigt mit der durchschnittlichen Reinheit der Untermengen
- Strategie: wähle Attribut, das den größten Informationszuwachs ergibt

Berechnung der Information

- Maß für Information in *Bits*
  - Gegeben eine Whr.verteilung, die durchschnittlich benötigte Information um ein Ereignis vorherzusagen ist die Entropie der Verteilung
  - Entropie ist die benötigte Information in Bits (können auch Anteile von Bits sein)
- Formel der Entropie:

$$\text{entropy}(p_1, p_2, \ldots, p_n) = -p_1 \log p_1 - p_2 \log p_2 \ldots - p_n \log p_n$$

Claude Shannon

"Father of information theory"

Born: 30 April 1916

Died: 23 February 2001

Claude Shannon, who has died aged 84, perhaps more than anyone laid the groundwork for today's digital revolution. His exposition of information theory, stating that all information could be represented mathematically as a succession of noughts and ones, facilitated the digital manipulation of data without which today's information society would be unthinkable.

Shannon's master's thesis, obtained in 1940 at MIT, demonstrated that problem solving could be achieved by manipulating the symbols 0 and 1 in a process that could be carried out automatically with electrical circuitry. That dissertation has been hailed as one of the most significant master's theses of the 20th century. Eight years later, Shannon published another landmark paper, *A Mathematical Theory of Communication*, generally taken as his most important scientific contribution.

Shannon applied the same radical approach to cryptography research, in which he later became a consultant to the US government.

Many of Shannon's pioneering insights were developed before they could be applied in practical form. He was truly a remarkable man, yet unknown to most of the world.

Beispiel: Attribut *Outlook*

- *Outlook* = *Sunny* :

$$\text{info}([2,3]) = \text{entropy}(2/5,3/5) = -2/5\log(2/5) - 3/5\log(3/5) = 0.971\,\text{bits}$$

- *Outlook* = *Overcast* :

$$\text{info}([4,0]) = \text{entropy}(1,0) = -1\log(1) - 0\log(0) = 0\,\text{bits}$$

Dies ist normaler-weise undefiniert.

- *Outlook* = *Rainy* :

$$\text{info}([3,2]) = \text{entropy}(3/5,2/5) = -3/5\log(3/5) - 2/5\log(2/5) = 0.971\,\text{bits}$$

- Erwartete Information für das Attribut:

$$\text{info}([3,2],[4,0],[3,2]) = (5/14)\times 0.971 + (4/14)\times 0 + (5/14)\times 0.971$$
$$= 0.693\,\text{bits}$$

Berechnung des Informationszuwaches (information gain)

- Informationszuwachs:

  Information vor der Teilung

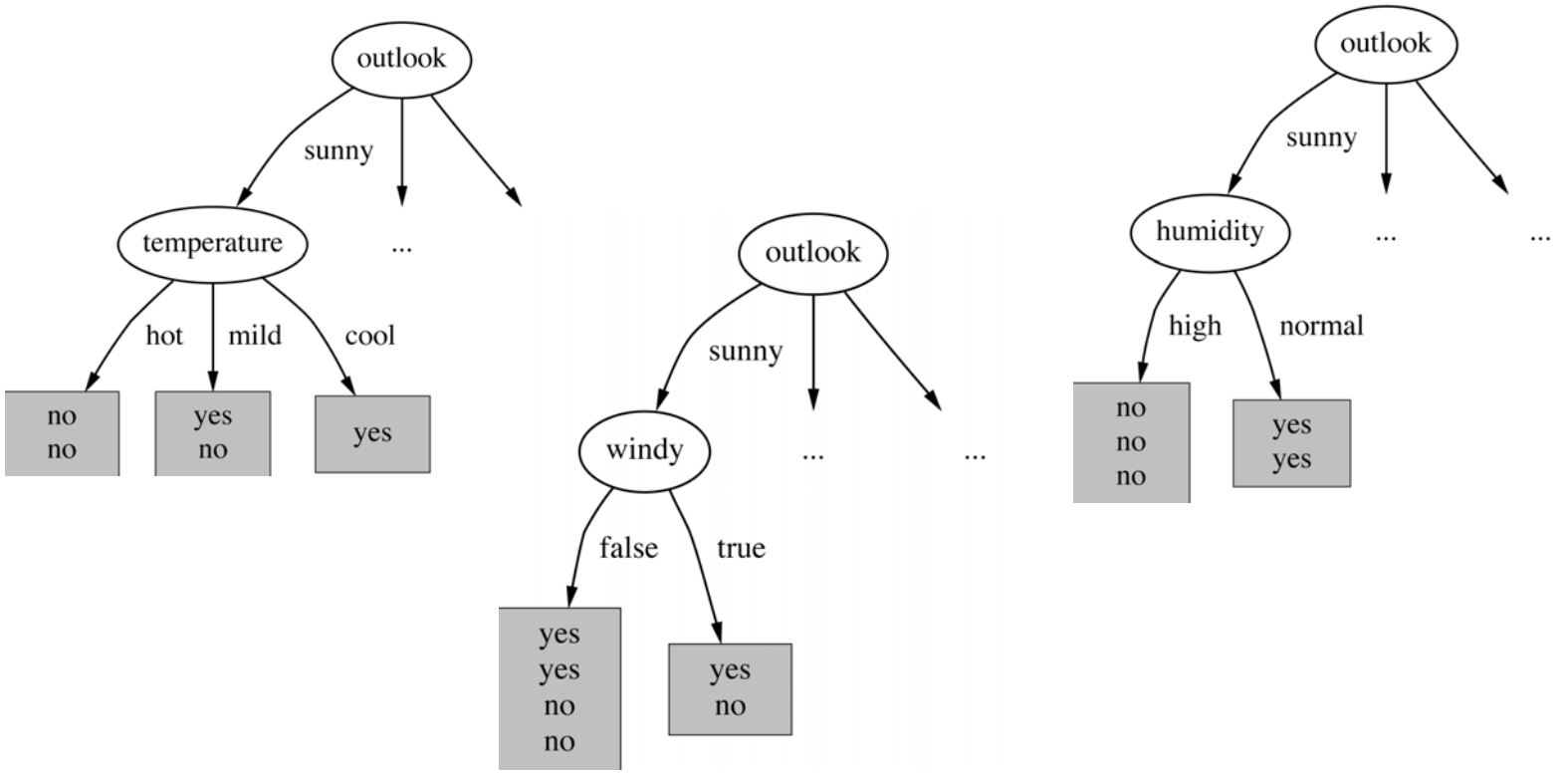
  – Information nach der Teilung

gain(*Outlook*) = info([9,5]) – info([2,3],[4,0],[3,2])
= 0.940 – 0.693
= 0.247 bits

- Informationszuwachs für Wetterdaten:

gain(*Outlook*) = 0.247 bits
gain(*Temperature*) = 0.029 bits
gain(*Humidity*) = 0.152 bits
gain(*Windy*) = 0.048 bits

Fortsetzung der Teilung

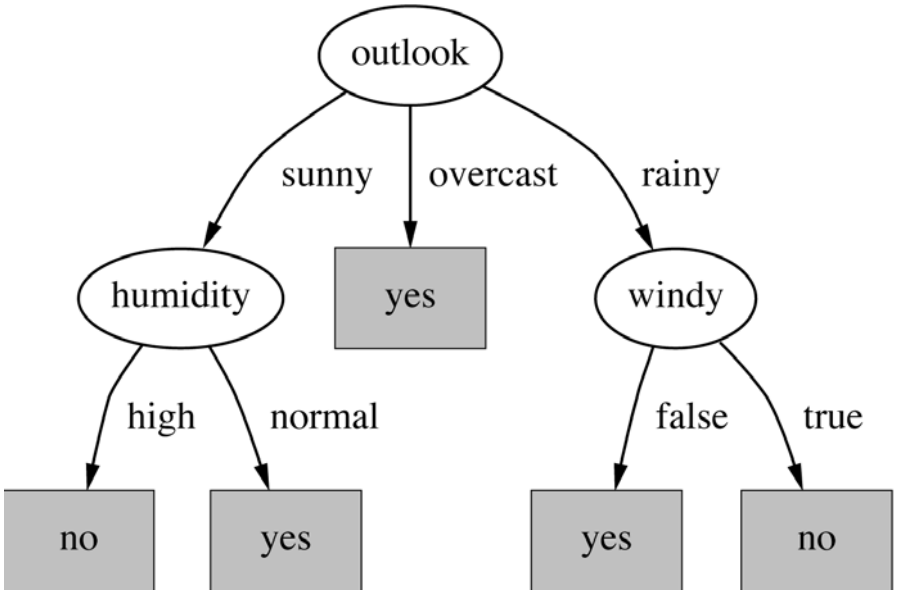


gain(*Temperature*) = 0.571 bits

gain(*Humidity*) = 0.971 bits

gain(*Windy*) = 0.020 bits

Fertiger Entscheidungsbaum



- Bemerkung: nicht alle Blätter müssen rein sein; manchmal haben identische Instanzen verschiedene Klassen

  $\Rightarrow$ Teilen hört auf wenn Daten nicht mehr geteilt werden können

Wunschliste für ein Reinheitsmaß

- Geforderte Eigenschaften für ein Reinheitsmaß:
  - Wenn Knoten rein ist, soll das Maß null sein
  - Wenn Unreinheit maximal ist (alle Klassen gleich whr.), Maß soll maximal sein
  - Maß soll Mehrschritt-Entscheidungen unterstützen (Entscheidungen können in mehreren Schritten gemacht werden):

$$\text{measure}([2,3,4]) = \text{measure}([2,7]) + (7/9) \times \text{measure}([3,4])$$

- Entropie ist ist die einzige Funktion, die alle drei Eigenschaften hat!

Eigenschaften der Entropie

- Mehrschritt Eigenschaft:

$$\text{entropy}(p,q,r) = \text{entropy}(p,q+r) + (q+r) \times \text{entropy}(\frac{q}{q+r}, \frac{r}{q+r})$$
$$p+q+r=1$$

- Vereinfachte Berechnung:

$$\text{info}([2,3,4]) = -2/9 \times \log(2/9) - 3/9 \times \log(3/9) - 4/9 \times \log(4/9)$$
$$= [-2\log 2 - 3\log 3 - 4\log 4 + 9\log 9]/9$$

- Bemerkung: anstelle Informationszuwachs zu maximieren, kann man die Information minimieren

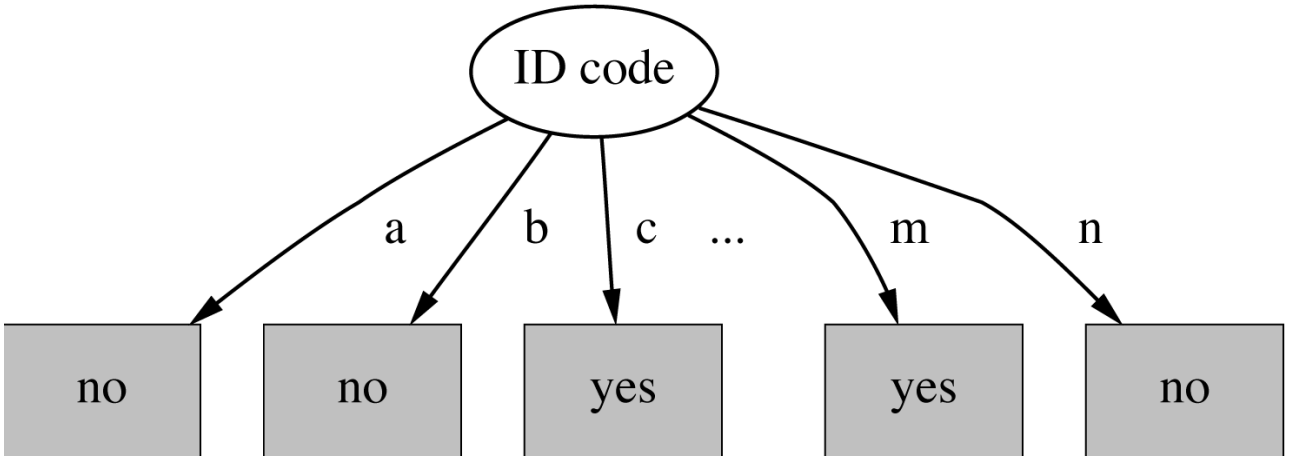
Hochverzweigende Attribute

- Problem: Attribute mit großer Anzahl von Werten (extremer Fall: ID code)
- Teilmengen sind wahrscheinlicher rein wenn in viele kleine Teilmengen aufgeteilt wird
  - $\Rightarrow$ Informationszuwachs tendiert dazu Attribute mit vielen Werten zu bevorzugen
  - $\Rightarrow$ Dies kann zu *overfitting* führen (Auswahl eines Attributes, das nicht optimal für die Vorhersage ist)

Wetterdaten mit *ID code*

ID code	Outlook	Temp.	Humidity	Wind	Play
A	Sunny	Hot	High	False	No
B	Sunny	Hot	High	True	No
C	Overcast	Hot	High	False	Yes
D	Rainy	Mild	High	False	Yes
E	Rainy	Cool	Normal	False	Yes
F	Rainy	Cool	Normal	True	No
G	Overcast	Cool	Normal	True	Yes
H	Sunny	Mild	High	False	No
I	Sunny	Cool	Normal	False	Yes
J	Rainy	Mild	Normal	False	Yes
K	Sunny	Mild	Normal	True	Yes
L	Overcast	Mild	High	True	Yes
M	Overcast	Hot	Normal	False	Yes
N	Rainy	Mild	High	True	No

Baumstumpf für *ID code* Attribut



- Entropie der Teilung:

$$\text{info}(\text{"ID code"}) = \text{info}([0,1]) + \text{info}([0,1]) + \ldots + \text{info}([0,1]) = 0 \text{ bits}$$

$\Rightarrow$ Informationszuwachs ist maximal für ID Code (0.940 bits)

Zuwachsverhältnis (Gain ratio)

- *Gain ratio*:
  - Modifikation des Informations-zuwachses, die die Tendenz reduziert
- Gain ratio beachtet auch Anzahl der Verzweigungen und deren Größe, wenn ein Attribut gewählt wird.
  - Informationszuwachs wird korrigiert durch Beachtung der inhärenten Information einer Teilung
- Inhärente Information:
  - Entropie der Aufteilung der Instanzen in Teilmengen (Wieviel Bits werden gebraucht, um zu entscheiden zu welcher Teilmenge eine Instanz gehört)

Berechnung des Zuwachsverhältnisses

- Beispiel: inhärente Information für ID code

$$\text{info}([1,1,\ldots,1]) = 14 \times (-1/14 \times \log 1/14) = 3.807 \text{ bits}$$

- Wichtigkeit eines Attributes nimmt zu, wenn inhärente Information zunimmt
- Definition des Zuwachsverhältnisses:

$$\text{gain\_ratio}(\text{"Attribute"}) = \frac{\text{gain}(\text{"Attribute"})}{\text{intrinsic\_info}(\text{"Attribute"})}$$

- Beispiel:

$$\text{gain\_ratio}(\text{"ID\_code"}) = \frac{0.940 \text{ bits}}{3.807 \text{ bits}} = 0.246$$

Zuwachsverhältnisse für Wetterdaten

Outlook		Temperature	
Info:	0.693	Info:	0.911
Gain: 0.940-0.693	0.247	Gain: 0.940-0.911	0.029
Split info: info([5,4,5])	1.577	Split info: info([4,6,4])	1.362
Gain ratio: 0.247/1.577	0.156	Gain ratio: 0.029/1.362	0.021
Humidity		**Windy**	
Info:	0.788	Info:	0.892
Gain: 0.940-0.788	0.152	Gain: 0.940-0.892	0.048
Split info: info([7,7])	1.000	Split info: info([8,6])	0.985
Gain ratio: 0.152/1	0.152	Gain ratio: 0.048/0.985	0.049

Mehr über Zuwachsverhältnis

- “Outlook” wird immer noch gewählt
- Aber: “ID code” hat größeres Zuwachsverhältnis
  - Standardlösung: *ad hoc* Test um Attribute diesen Types auszuschließen
- Problem mit Zuwachsverhältnis: es kann überkompensieren
  - Ein Attribut könnte nur wegen niedriger inhärenter Information gewählt werden
  - Standard Lösung: es werden nur Attribute gewählt, deren Informationszuwachs > als der Durchschnitt ist

Diskussion

- Top-down Induktion von Entscheidungsbäumen: ID3, Algorithmus entwickelt von Ross Quinlan
  - Zuwachsverhältnis ist nur eine Verbesserung des Basis-Algorithmus
  - $\Rightarrow$ C4.5: kann nummerische Attribute, fehlende Daten und Rauschen handhaben
- Ähnlicher Ansatz: CART
- Es gibt noch mehr Auswahlkriterien! (Aber nur wenig Unterschiede in der Qualität der Ergebnisse)

*CART Auswahlkriterium: Gini Index

- Daten T enthalten Beispiele aus n Klassen, Gini Index, gini(T) ist definiert als

$$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$

  p_j ist relative Häufigkeit von Klasse j in T.
- gini(T) ist klein, wenn Klassen in T stark ungleich verteilt sind.

*Gini Index

- Nach Teilung von T in zwei Untermengen T1 und T2 mit Größen N1 und N2,
  der Gini Index ist definiert als:

$$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$

- Das Attribut mit dem kleinsten $gini_{split}(T)$ wird ausgewählt.

Zusammenfassung

- Top-Down Entscheidungsbaum Konstruktion
- Auswahl der Attribute zum Teilen
- Informationszuwachs (Information Gain) bevorzugt Attribute mit vielen Werten
- Zuwachsverhältnis (Gain Ratio) berücksichtigt Anzahl und Größe der Teilmengen bei der Attributauswahl