Klasifikacija:
odločitvena drevesa

Vsebina

- Gradnja odločitvenih dreves "odzgoraj navzdol"
- Izbira "najboljšega" atributa (za delitev)
- Informacijski prispevek in
  razmerje informacijskega prispevka

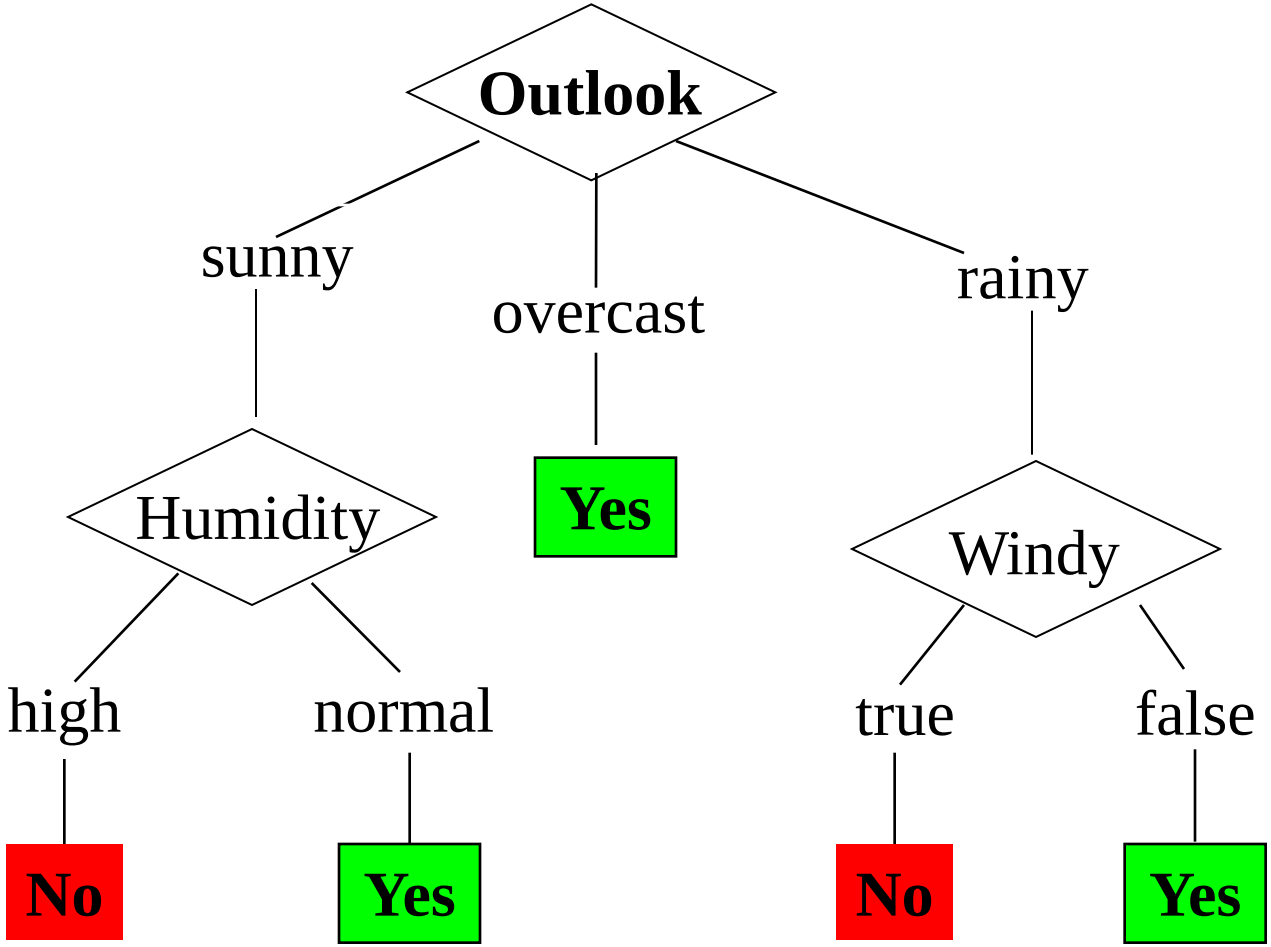
Odločitveno drevo

- Notranje vozlišče predstavlja test po atributu;
- Veja predstavlja rezultat tega testa, npr. "Color=red";
- List predstavlja oznako razreda (ali porazdelitev vrednosti razreda);
- V vsakem vozlišču: en atribut je izbran, po katerem delimo učne primere v kar se da "čiste" podmnožice;
- Nove primere klasificiramo tako, da sledimo ustreznim potem v drevesu od korena do listov.

Podatki "weather": Play? – Yes?, No?

Outlook	Temperature	Humidity	Windy	Play?
sunny	hot	high	false	No
sunny	hot	high	true	No
overcast	hot	high	false	Yes
rain	mild	high	false	Yes
rain	cool	normal	false	Yes
rain	cool	normal	true	No
overcast	cool	normal	true	Yes
sunny	mild	high	false	No
sunny	cool	normal	false	Yes
rain	mild	normal	false	Yes
sunny	mild	normal	true	Yes
overcast	mild	high	true	Yes
overcast	hot	normal	false	Yes
rain	mild	high	true	No

Primer drevesa za “Play?”



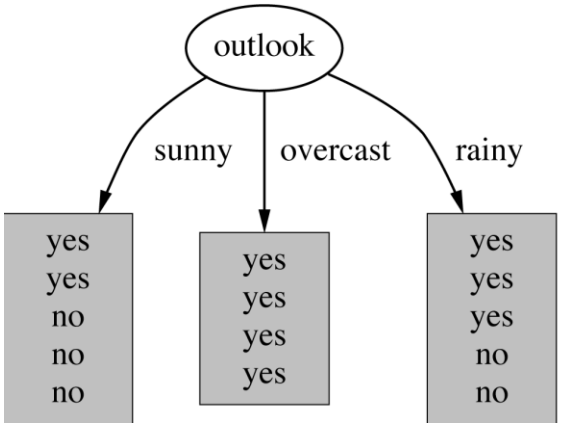
Gradnja odločitvenega drevesa (ID3 algoritem, TDIDT princip)

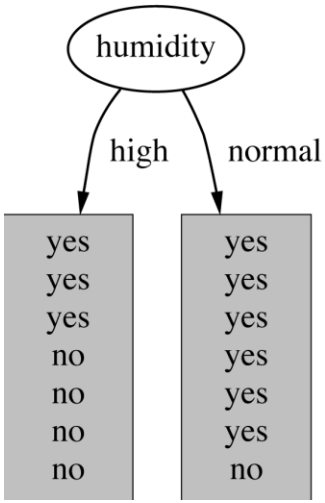
- Gradnja drevesa "odzgoraj navzdol" (top-down)
  - Na začetku so vsi primeri v korenu drevesa;
  - Rekurzivno delimo primere v podmnožice s pomočjo atributov in njihovih vrednosti;
- Rezanje drevesa "odspodaj navzgor" (bottom-up)
  - Odstranimo poddrevesa ali veje drevesa z namenom izboljšati točnost napovedi drevesa na novih primerih;

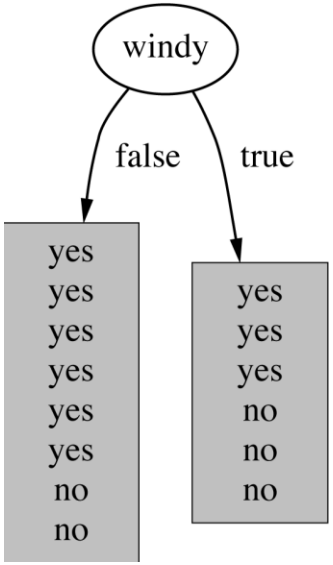
Izbor atributa za delitev primerov

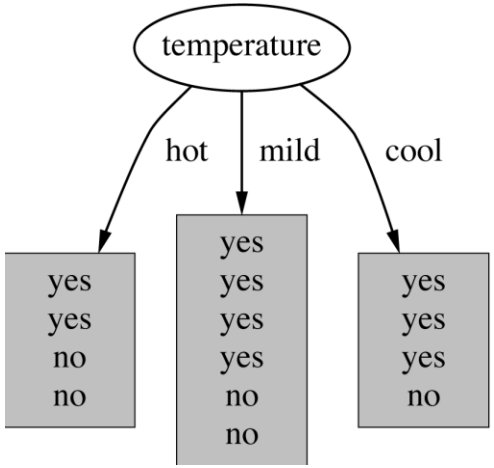
- V vsakem vozlišču drevesa: ocenimo razpoložljive atribute glede na njihovo sposobnost delitve primerov na "čiste" podmnožice – uporabimo **funkcijo primernosti** (goodness function);
- Tipične funkcije primernosti:
  - Informacijski prispevek (ID3, C4.5)
  - Razmerje informacijskega prispevka (C4.5)
  - Gini indeks (CART)

Kateri atribut izbrati?
("čistost" podmnožic)









Kriterij za izbiro "najboljšega" atributa

- Kateri atribut je najboljši?
  - Tisti, ki na koncu pripelje do najmanjšega drevesa;
  - Hevristika: izberemo atribut, ki razdeli podatke na "najčistejše" podmnožice (primer na prejšnji prosojnici);
- Kako merimo "čistost"? ***Informacijski prispevek***
  - Informacijski prispevek narašča s "čistostjo" podmnožic, na katere nek atribut razdeli podatke;
- Strategija: izberemo atribut z največjim informacijskim prispevkom

Računanje informacije

- Informacijski prispevek merimo v **bitih**
  - Informacijo, ki jo potrebujemo, da lahko napovemo nek dogodek, če poznamo verjetnostno porazdelitev "vseh" dogodkov, imenujemo **entropija** porazdelitve;
  - Entropija poda zahtevano informacijo v bitih (kar ni nujno celo število – lahko so delčki bitov!)
- Formula za izračun entropije:

$$\text{entropija}(p_1, p_2, \ldots, p_n) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 \ldots - p_n \log_2 p_n$$

*Claude Shannon

"Oče teorije informacij"

Rojen: 30. aprila 1916

Umrl: 23. februarja 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.

Primer: atribut "Outlook", 1

Outlook	Temperature	Humidity	Windy	Play?
sunny	hot	high	false	No
sunny	hot	high	true	No
overcast	hot	high	false	Yes
rain	mild	high	false	Yes
rain	cool	normal	false	Yes
rain	cool	normal	true	No
overcast	cool	normal	true	Yes
sunny	mild	high	false	No
sunny	cool	normal	false	Yes
rain	mild	normal	false	Yes
sunny	mild	normal	true	Yes
overcast	mild	high	true	Yes
overcast	hot	normal	false	Yes
rain	mild	high	true	No

Primer: atribut "Outlook", 2

- "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":

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

***Pozor**:*
***$\log_2(0)$** ni definirana vrednost, vseeno pa lahko vzamemo, da je **$0*\log_2(0)$** enako nič.*

- "Outlook" = "Rainy":

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

- Pričakovana informacija atributa "Outlook":

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

$$= 0.693\,\text{bits}$$

Računanje informacijskega prispevka

- Informacijski prispevek =

(informacija pred razbitjem) – (informacija po razbitju)

$$\text{gain}(\text{"Outlook"}) = \text{info}([9,5]) - \text{info}([2,3],[4,0],[3,2]) = 0.940 - 0.693$$

$$= 0.247 \text{ bits}$$

- Poskusimo izračunati za atribut “Humidity”!

Primer: atribut "Humidity"

- "Humidity" = "High":

$$\text{info}([3,4]) = \text{entropy}(3/7,4/7) = -3/7\log(3/7) - 4/7\log(4/7) = 0.985 \text{ bits}$$

- "Humidity" = "Normal":

$$\text{info}([6,1]) = \text{entropy}(6/7,1/7) = -6/7\log(6/7) - 1/7\log(1/7) = 0.592 \text{ bits}$$

- Pričakovana informacija atributa "Humidity":

$$\text{info}([3,4],[6,1]) = (7/14)\times 0.985 + (7/14)\times 0.592 = 0.79 \text{ bits}$$

- Informacijski prispevek:

$$\text{info}([9,5]) - \text{info}([3,4],[6,1]) = 0.940 - 0.788 = 0.152$$

Informacijski prispevek – "weather"

- Informacijski prispevek =

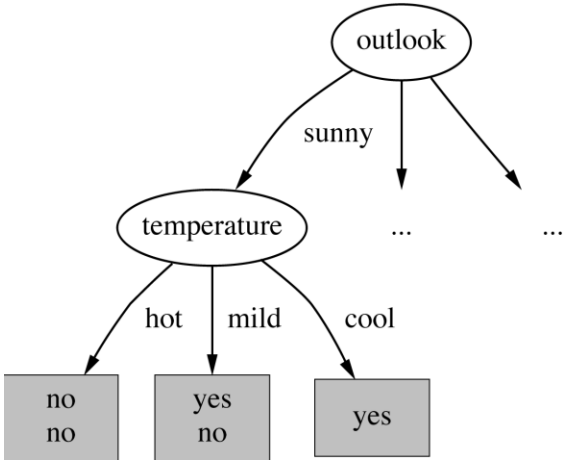
(informacija pred razbitjem) – (informacija po razbitju)

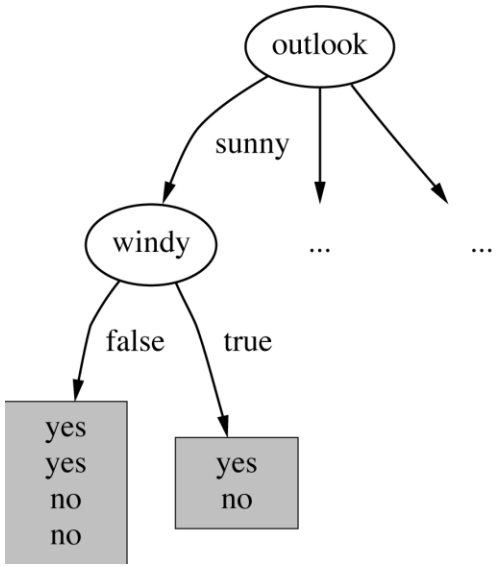
$$\text{gain}(\text{"Outlook"}) = \text{info}([9,5]) - \text{info}([2,3],[4,0],[3,2]) = 0.940 - 0.693$$
$$= 0.247 \text{ bits}$$

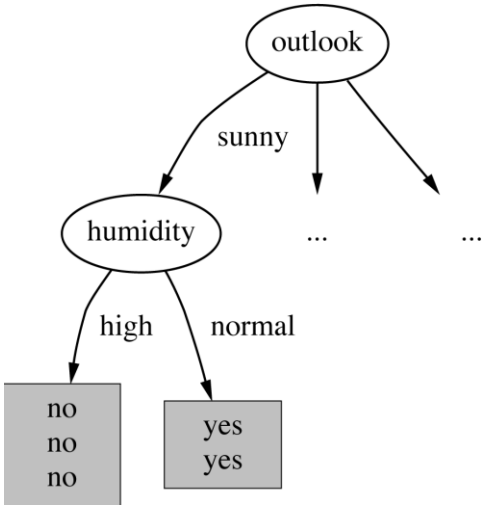
- Informacijski prispevek atributov iz podatkov "weather":

$$\text{gain}(\text{"Outlook"}) = 0.247 \text{ bits}$$
$$\text{gain}(\text{"Temperature"}) = 0.029 \text{ bits}$$
$$\text{gain}(\text{"Humidity"}) = 0.152 \text{ bits}$$
$$\text{gain}(\text{"Windy"}) = 0.048 \text{ bits}$$

Nadaljevanje gradnje drevesa ...





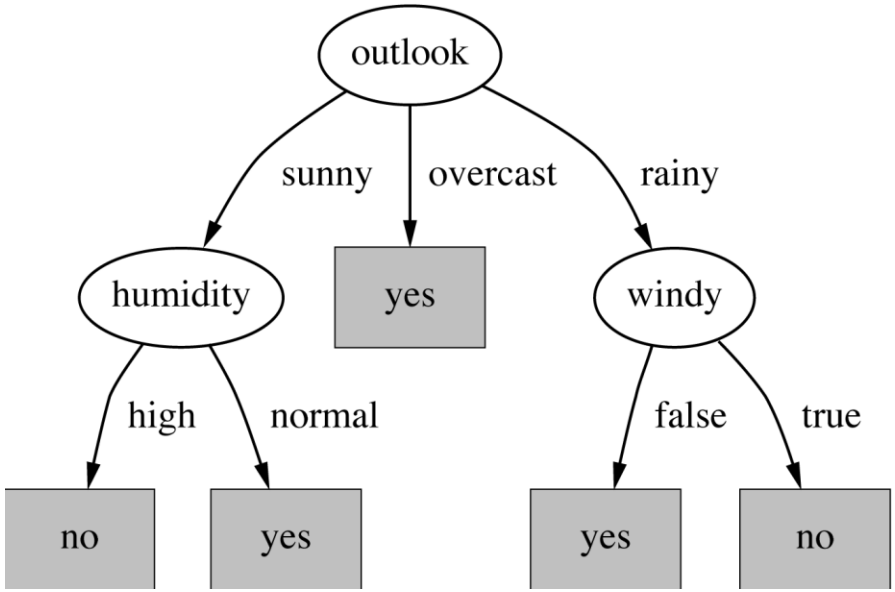


$$gain("Humidity") = 0.971\,bits$$

$$gain("Temperature") = 0.571\,bits$$

$$gain("Windy") = 0.020\,bits$$

Končno odločitveno drevo



- Pozor: niso vedno vsi listi "čisti"; včasih imata lahko primera enake vrednosti vseh atributov, a različna razreda

  $\Rightarrow$ ko ni možno več "vejiti" naprej, se ustavimo!

*Zaželjene lastnosti mer "čistosti"

- Katere so lastnosti, ki jih zahtevamo od mer "čistosti"?
  - Ko je vozlišče "čisto", naj bo mera enaka nič,
  - Ko je "nečistoča" največja (t.j. vse vrednosti razreda so enako zastopane), naj ima mera največjo vrednost,
  - Mera naj ima ***več-nivojsko lastnost***
    (t.j. mero lahko računamo po nivojih – v korakih):

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

- Entropija je funkcija,
  ki zadošča vsem trem podanim zahtevam!

*Lastnosti entropije

- Več-nivojska lastnost:

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

- Poenostavitev računanja entropije:

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

- <u>Pozor</u>:
  namesto maksimizacije informacijskega prispevka, bi lahko preprosto minimizirali entropijo posamezne vejitve.

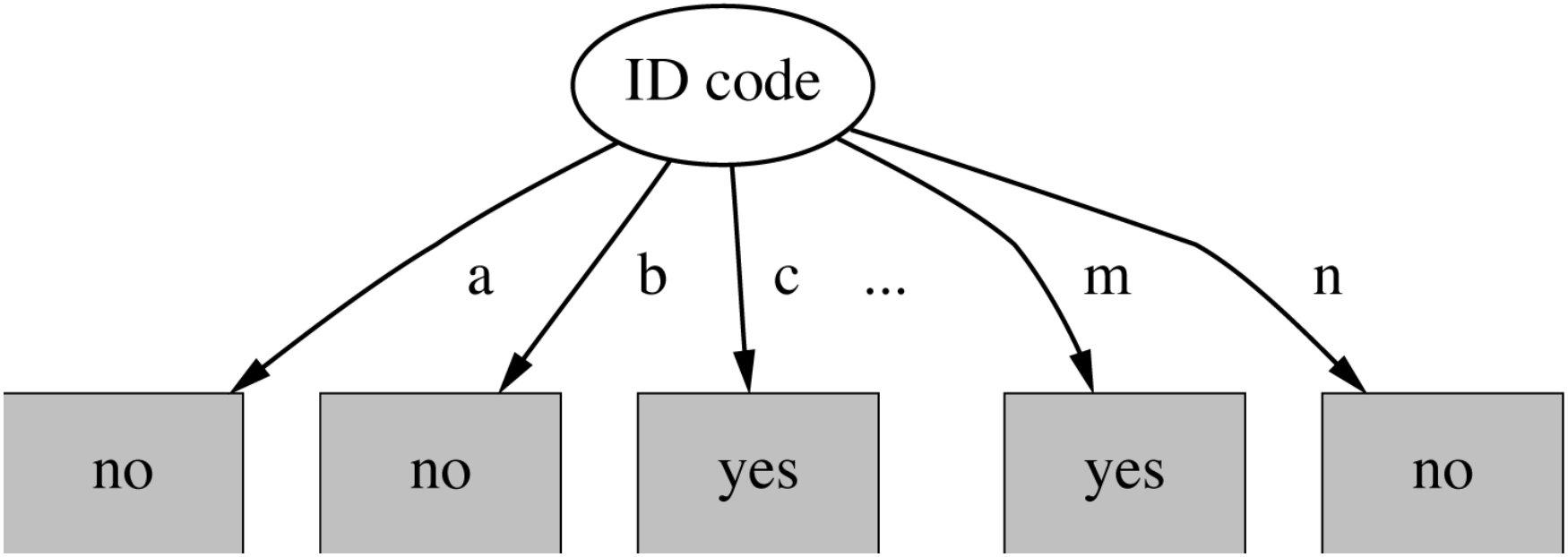
Atributi z veliko vrednostmi/vejami

- Težava: atributi z zelo veliko vrednostmi (ekstremni primer: "ID koda")
- Če ima atribut več vrednosti, je večja verjetnost, da bodo podmnožice "čiste"
  - ⇒ Informacijski prispevek je pristranski in "raje izbira" atribute z več vrednostmi;
  - ⇒ To lahko vodi do prekomernega prileganja (*overfitting*) = izbira atributa, ki ni optimalen za klasifikacijo;

Podatki "weather" + ID koda

ID	Outlook	Temperature	Humidity	Windy	Play?
A	sunny	hot	high	false	No
B	sunny	hot	high	true	No
C	overcast	hot	high	false	Yes
D	rain	mild	high	false	Yes
E	rain	cool	normal	false	Yes
F	rain	cool	normal	true	No
G	overcast	cool	normal	true	Yes
H	sunny	mild	high	false	No
I	sunny	cool	normal	false	Yes
J	rain	mild	normal	false	Yes
K	sunny	mild	normal	true	Yes
L	overcast	mild	high	true	Yes
M	overcast	hot	normal	false	Yes
N	rain	mild	high	true	No

Razbitje po "ID koda" atributu



Entropija razbitja = 0
(vsi listi so "čisti", saj vsebujejo le po en primer)

Informacijski prispevek za atribut "ID koda" je maksimalen

Razmerje informacijskega prispevka

- Razmerje informacijskega prispevka (Gain Ratio): "prilagoditev" informacijskega prispevka z namenom zmanjšanja njegove pristranskosti proti atributom z več vrednostmi;
- Razmerje informacijskega prispevka naj bi:
  - Bilo višje pri "enakomerno porazdeljenih" atributih;
  - Bilo nižje, ko primeri pripadajo vsi eni veji;
- Razmerje informacijskega prispevka upošteva število in velikost vej (pri izbiri atributa):
  - "popravi" informacijski prispevek tako, da upošteva ***intrinzično informacijo*** razbitja (t.j. koliko inforamcije potrebujemo, za določitev veje primeru)

Intrinzična informacija

- Intrinzična informacija = entropija porazdelitve primerov v posamezne veje atributa

$$IntrinsicInfo(S,A) \equiv -\sum \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}.$$

- ***Razmerje informacijskega prispevka*** (Quinlan'86) normalizira informacijski prispevek z intrinzično informacijo atributa:

$$GainRatio(S,A) = \frac{Gain(S,A)}{IntrinsicInfo(S,A)}.$$

Izračun razmerja info. prispevka

- Primer: intrinzična informacija za "ID kodo"

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

- **Pomembnost atributa se zmanjšuje z naraščanjem intrinzične informacije**
- Primer razmerja informacijskega prispevka:

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

- Primer:

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

Razmerje informasijskega prispveka za podatke "weather"

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

Več o razmerju info. prispevka

- "Outlook" je še vedno "najboljši"
- Toda: "ID koda" ima višje razmerje info. prispevka
  - Popravek: *ad hoc* test, da bi preprečili razbitje po take vrste atributih
- Težava razmerja informacijskega prispevka: lahko nad-kompenzira
  - Lahko izbere atribut zgolj zaradi nizke intrinzične info.
  - Popravek:
    - 1. korak: izberemo le atribute z "dovolj visokim" info. prispevkom
    - 2. korak: izbrane atribute primerjamo po razmerju info. prispevka

*Kriterij algoritma CART: **Gini indeks**

- Če podatki (**T**) vsebujejo primere iz **n** razredov, potem je Gini indeks definiran kot:

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

kjer $\mathbf{p_j}$ predstavlja verjetnost primerov z razredom **j** v podatkih **T**;

- **gini(T)** je najmanjši, če so razredi karseda neenakomerno zastopani v T;

*Gini indeks

- Če množico **T** razbijemo na dve podmnožici $\mathbf{T_1}$ in $\mathbf{T_2}$ velikosti $\mathbf{N_1}$ in $\mathbf{N_2}$, je Gini indeks razbitja definiran kot:

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

- Atribut z najmanjšim $\mathbf{gini_{split}(T)}$ bo izbran kot "najboljši".

Diskusija

- Algorithm za izgradnjo odločitvenih dreves ID3 (Iterative Dichotomizer 3) je razvil Ross J. Quinlan:
  - Razmerje informacijskega prispevka je le ena možnih modifikacij tega algoritma;
  - To je pripeljalo k razvoju algoritma C4.5, ki zna obravnavati numerične atribute, manjkajoče vrednosti in šumne podatke;
- Podoben pristop: CART (ne bomo obravnavali);
- Obstaja še mnogo drugih kriterijev za izbor atributov! (a skoraj ni razlik v klasifikacijski točnosti rezultatov)

Povzetek

- Gradnja drevesa "odzgoraj navzdol"
- Izbira atributa za delitev
- Informacijski prispevek "preferira" atribute z večjim številom vrednosti
- Razmerje informacijskega prispevka upošteva število in "velikost vej" drevesa pri izbiri atributa