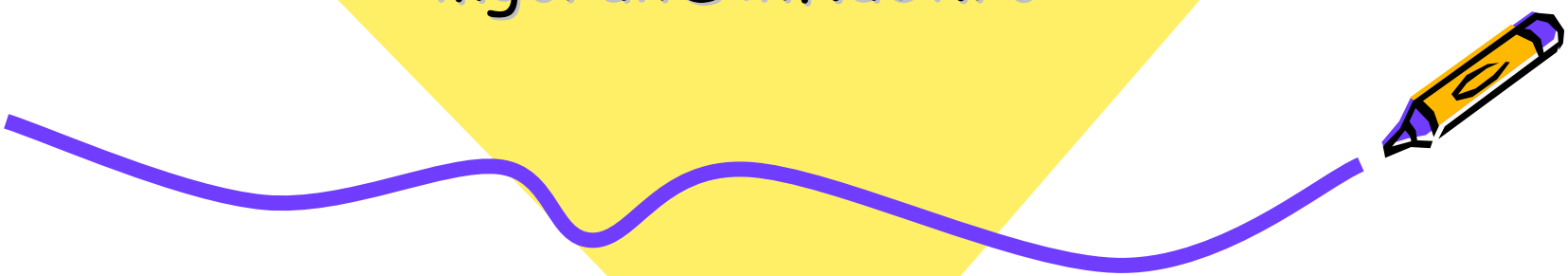
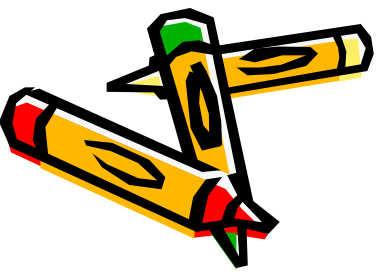
Arbori de clasificare si decizie

mgorun@inf.ucv.ro



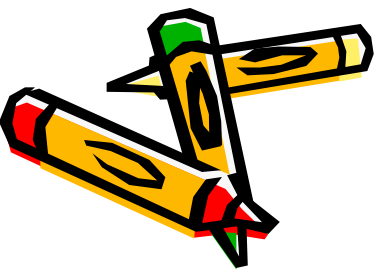
Putem face o clasificare, punând un șir de întrebări, șir în care fiecare întrebare este formulată în funcție de răspunsul primit la precedenta.

Acest procedeu merită a fi folosit în cazul datelor non-metrice, în sensul că de obicei răspunsurile la întrebări vor fi da/nu, adevărat/fals, proprietatea aparține sau nu unei mulțimi de proprietăți etc.



arbore de clasificare

Setul de întrebări referitoare la atributelor obiectelor ce urmează a fi clasificate se reprezintă printr-un ***arbore de clasificare***, care este un arbore în sens informatic.

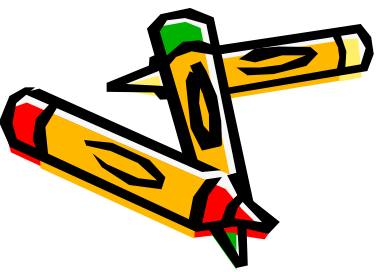


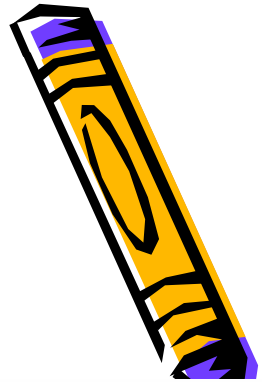
nodul radacina, ramuri, noduri de decizie

Prin convenție, primul nod -***nodul rădăcină***- se află în vârf, legat prin ***ramuri*** (links) de nodurile interne -***noduri de decizie***-.

Clasificarea unui anumit element (pattern) începe din nodul rădăcină, unde este pusă o anumită întrebare relativă la o proprietate specifică.

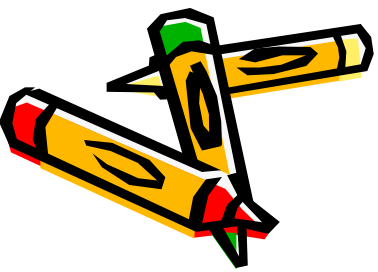
Răspunsurile posibile corespund etichetelor ramurilor.





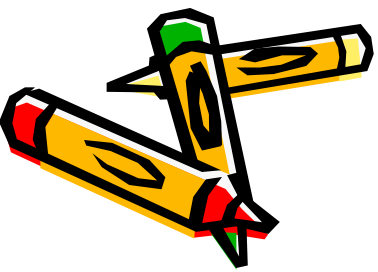
În cazul unor decizii binare, prin convenție, arcul din stânga corespunde unui răspuns afirmativ la test.

În funcție de răspunsuri, care trebuie să fie distincte şi exhausive, urmăm link-ul corespunzător spre un nod descendent, care ar putea fi considerat ca fiind rădăcina unui sub-arbore.



frunze

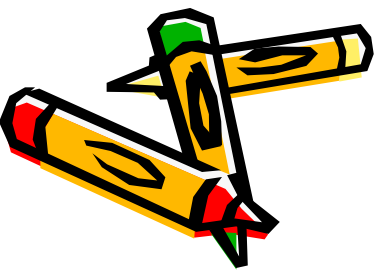
Continuăm astfel, până la nodurile terminale -***frunze***-, cărora nu le mai corespunde nici o întrebare, și care astfel nu mai au ramuri. Unui nod frunză îi corespunde o anumită categorie (clasă).



arbore de clasificare si decizie

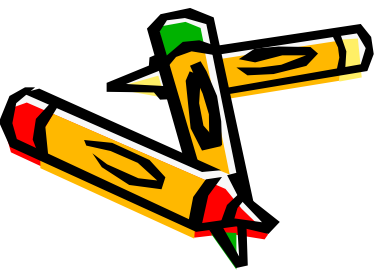
Un arbore de clasificare este utilizat în luarea unei decizii, motiv pentru care este folosită sintagma arbore de *clasificare și decizie*.

Acesta partiționează în mod recursiv mulțimea de antrenament până la obținerea nodurilor finale, care conțin fie numai elemente din aceeași categorie, fie elemente dintr-o categorie dominantă.



Putem interpreta decizia pentru orice clasificare ca fiind suma deciziilor de-a lungul drumului dintre nodul rădăcină și nodul frunză.

Cunoștințele experților umani au deosebită importanță în cazul unei mulțimi de antrenament ce are puține elemente.

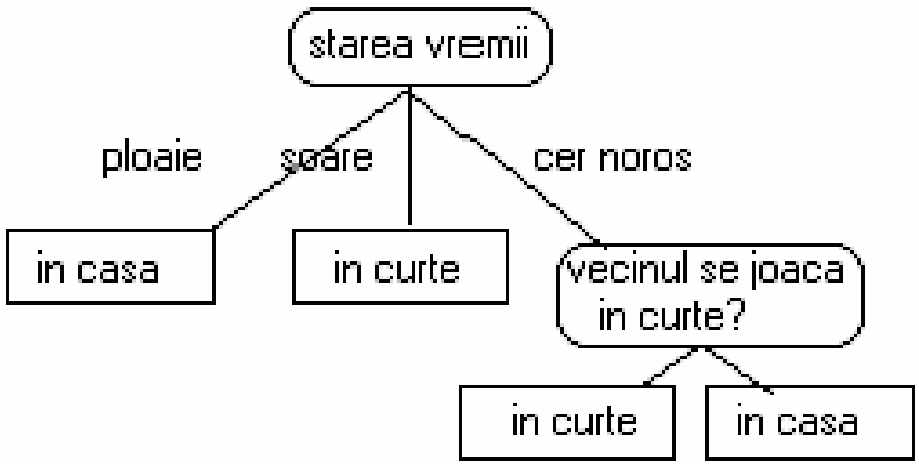


Arborele construit se foloseşte pentru a clasifica exemple necunoscute, în sensul de a decide dacă acestea aparţin sau nu unei anumite clase.

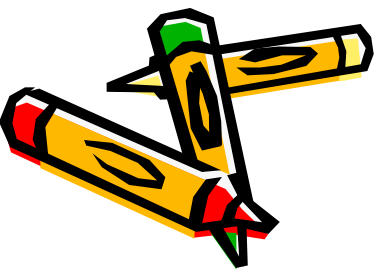
Un arbore de clasificare şi decizie poate fi interpretat ca o reprezentare grafică a unui procedeu de clasificare, nodurile interne fiind testele pentru atribute iar frunzele fiind clasele.

exemplu

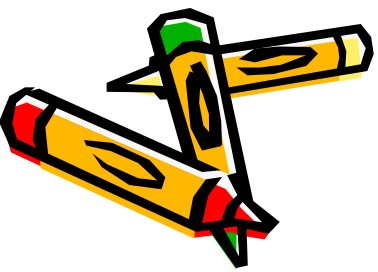
în ce condiții meteo ți se permite, copil fiind, să te joci în curte?



Utilizarea arborelui de clasificare și decizie este indicată nu numai pentru buna clasificare a rezultatelor ci și pentru luarea unor decizii optime prin obținerea unor reguli ușor de înțeles și explicat.



O regulă este creată coborând din vârf -nodul rădăcină- până la fiecare frunză şi este de tipul IF-THEN.
Orice pereche de valori ale unui atribut de-a lungul acestui traseu va forma o conjuncție în ipoteza regulii, iar frunza conținând clasa predictivă va forma consecința regulii .

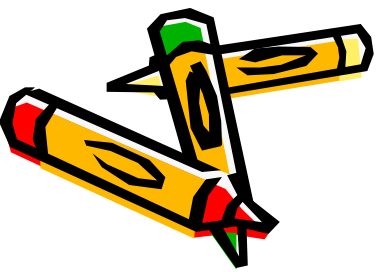


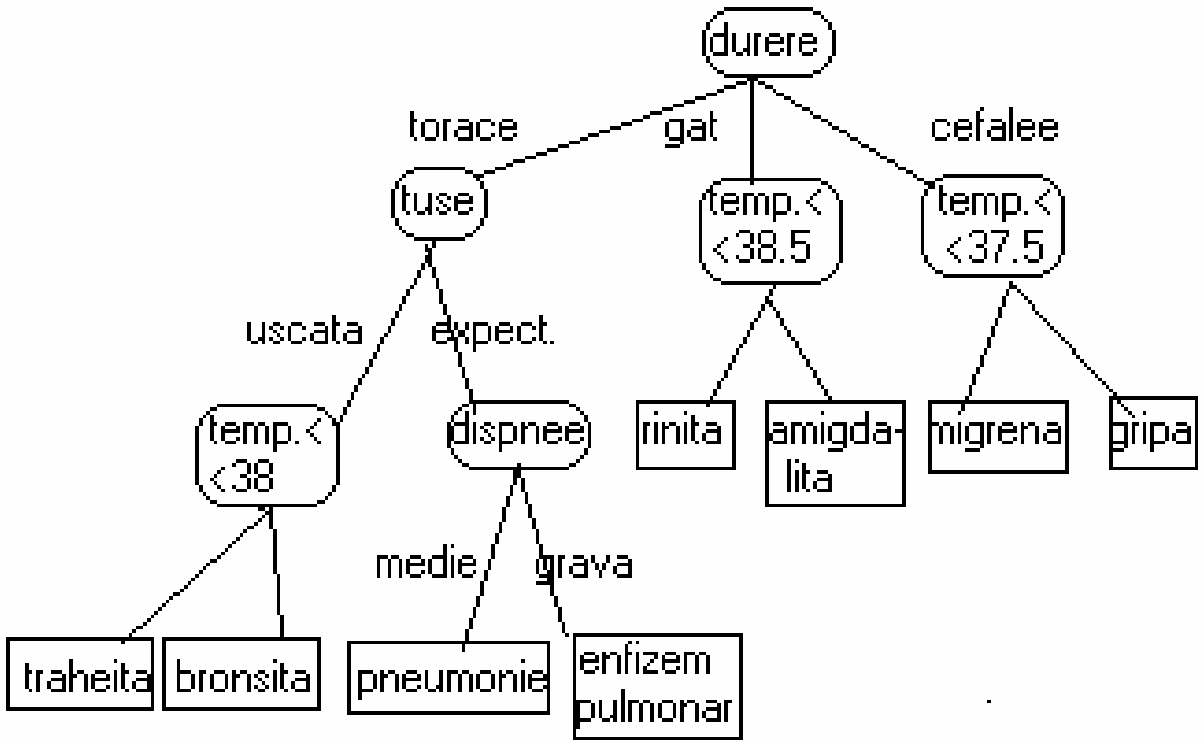
in exemplul dat avem:

- dacă plouă, rămâi să te joci în casă;
- dacă e soare te joci în curte;
- dacă cerul e noros și copilul vecin se joacă în curte, te joci în curte;
- dacă cerul e noros și copilul vecin nu se joacă în curte, te joci în casă.

exemplu

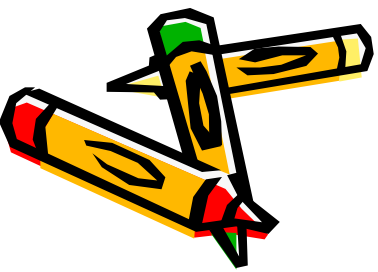
Un medic construieşte următorul arbore de clasificare pentru diagnosticarea unor boli.
Proprietățile, în acest caz simptoamele sunt: durere, tuse, stare febrilă, dispnee (respirație dificilă) .




durere
torace
gat
cefalee
tuse
temp.<
<38.5
temp.<
<37.5
uscata
expect.
temp.<
<38
dispnee
rinita
amigda-
lita
migrena
gripa
medie
grava
traheita
bronsita
pneumonie
enfizem
pulmonar

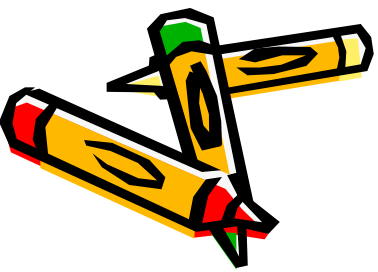
reguli de clasificare

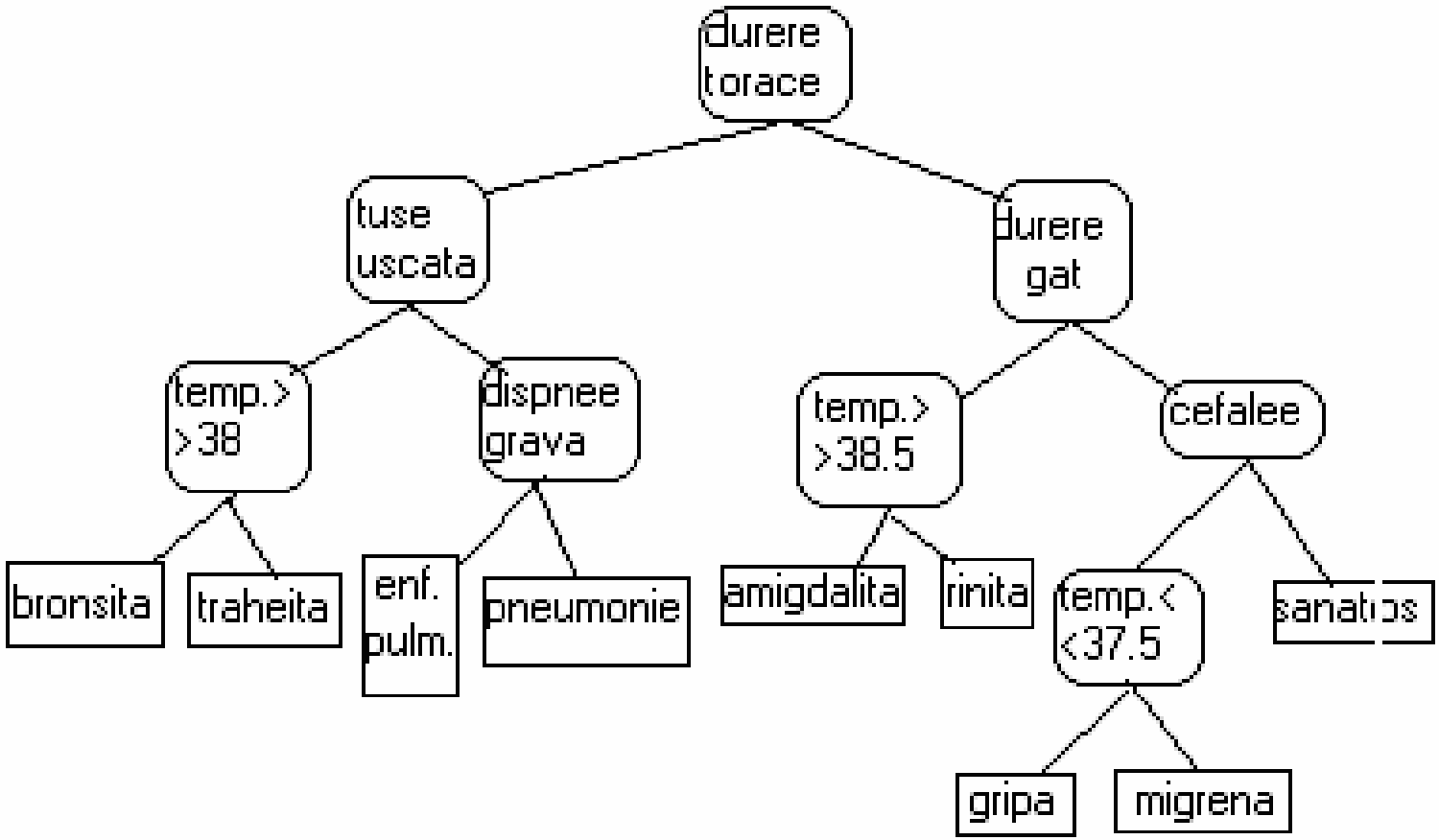
- Pattern-ul {*durere torace, tuse uscată, temperatură* >38} este clasificat ca fiind *bronşită* (traheo - bronşită)
- Pattern-ul {*durere torace, tuse expectorantă, dispnee gravă*) este clasificat ca fiind *enfizem pulmonar*
- Pattern-ul {*cefalee, temperatură* < 37,5} este clasificat ca fiind *migrenă*.



Un arbore oarecare poate fi reprezentat printr-un arbore binar echivalent, prin modificări minime ale întrebărilor test.

Prezentăm arborele binar corespunzător arborelui de diagnosticare construit anterior:

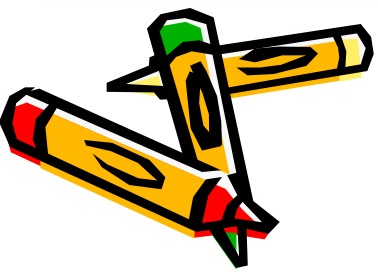



durere
torace
tuse
uscata
durere
gat
temp.>
>38
dispnee
grava
temp.>
>38.5
cefalee
bronsita
traheita
enf.
pulm.
pneumonie
amigdalita
rinita
temp.<
<37.5
gripa
migrena

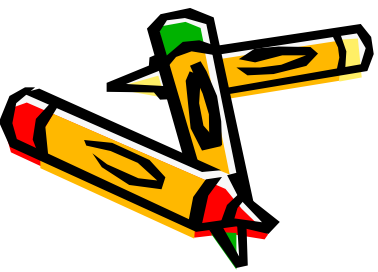
CART

CART (*Classification and Regression Trees, Breiman* 1984) este un algoritm ce constă în construcția arborelui folosind datele din mulțimea de antrenament.

Presupunem că avem o mulțime de date X, în care elementele sunt deja clasificate și cunoaștem proprietățile care fac distincția între clase.

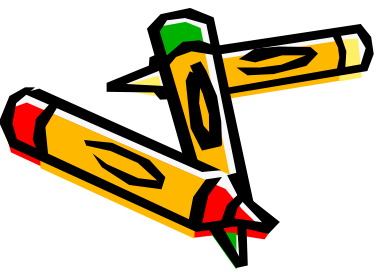


Construind acest arbore, mulțimea de antrenament va fi descompusă în submulțimi din ce în ce mai mici.
Fiecare ieșire (outcome) dintr-un nod se numește ***split*** și corespunde unei descompuneri a mulțimii de antrenament.
Rezultatul ideal ar fi ca să obținem o submulțime ce conține numai elemente ale aceleași clase, o submulțime pură, adică o ***frunză***.



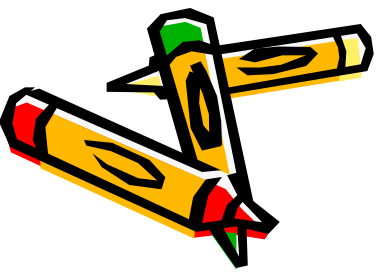
De cele mai multe ori submulțimile conțin și elemente din alte clase.

Ce facem:
acceptăm o decizie imperfectă sau
continuăm construcția arborelui, luând în considerare altă proprietate?

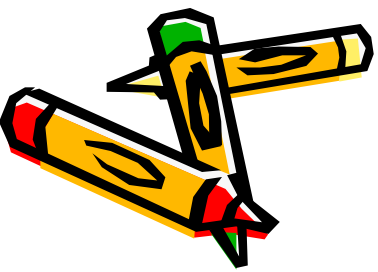


In construirea arborelui urmărim:

- Acuratețea clasificării ;
- Abilitatea explicării motivului luării unei decizii

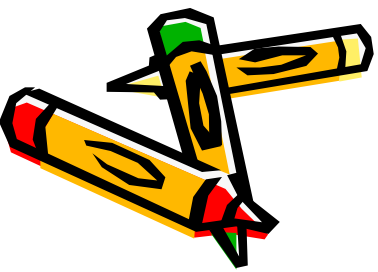


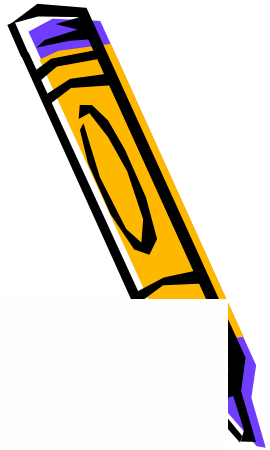
- se construieşte arborele de clasificare şi decizie pe baza mulțimii de antrenament (mulțime de date cunoscute) ;
- se utilizează arborele pentru a clasifica exemple necunoscute, în sensul de a decide cărei clase aparțin.



In construcția arborelui avem de rezolvat o serie de probleme:

- Câte ieşiri să avem la un nod?
- Ce proprietate să testăm la un nod?

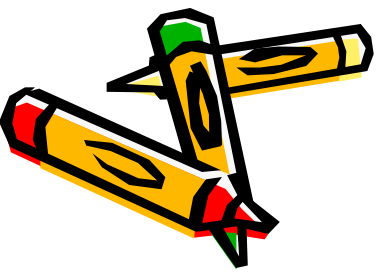




Ne vom ocupa mai mult de cazul arborilor binari.

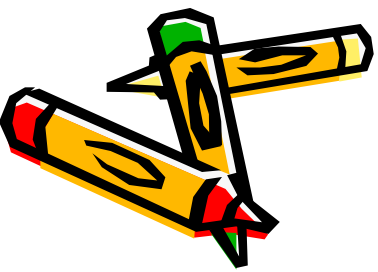
Vom studia doar cazul în care fiecare întrebare se referă la o singură proprietate, cazul arborelui „*monothetic*".

Ideal este să construim un arbore cât mai simplu, cu noduri puține, și în acest scop întrebarea din nodul N trebuie formulată în așa fel încât nodul $N+1$ să fie cât mai „pur".



masura de impuritate

Din punct de vedere matematic este mai simplu să definim o ***măsură de impuritate***, notată $i(N)$, care va fi nulă dacă toate elementele din nod aparțin aceleași clase și va fi maximă dacă în nod avem număr egal de elemente din fiecare clasă.

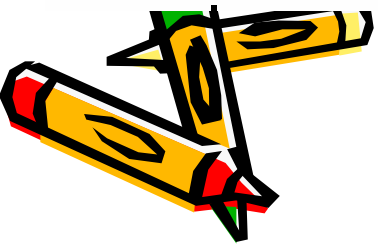


masura entropiei

Considerând că în mulțimea de antrenament există clasele $\Omega_1,\ldots,\Omega_r$, notăm cu $P(\Omega_j)$ raportul dintre numărul de elemente din nodul N aparținând clasei Ω_j și numărul de elemente din nodul N și putem defini:

măsura (*funcția*) ***entropiei*** în nodul N, dată de:

$$i(N) = -\sum_j P(\Omega_j).\log_2 P(\Omega_j);$$

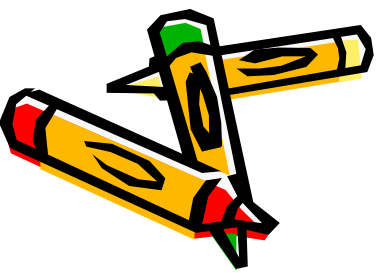


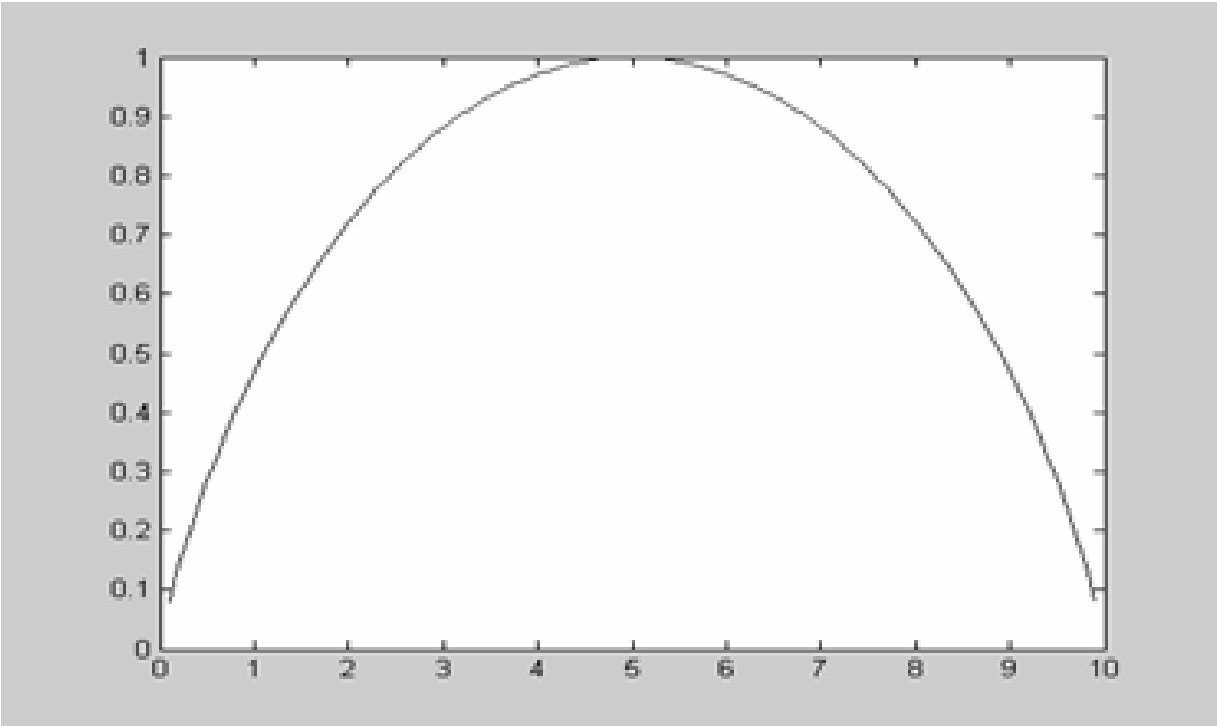
În cazul a două clase Ω_1 și Ω_2 avem:

$$i(N) = -P(\Omega_1)\log_2 P(\Omega_1) - P(\Omega_2)\log_2 P(\Omega_2)$$

dacă în nodul N avem n elemente aparținând claselor Ω_1 și Ω_2, dintre care x în clasa Ω_1, măsura entropiei în nod poate fi considerată a fi funcție de x:

$$f(x) = -\frac{x}{n}\log_2\frac{x}{n} - \left(1-\frac{x}{n}\right)\cdot\log_2\left(1-\frac{x}{n}\right)$$





graficul funcției în cazul $n = 10$.

Valoarea maximă a măsurii entropiei în acest caz este egală cu 1.

informatia castigata

Problema de rezolvat: ce întrebare a testului punem în nodul N?
Considerăm cazul unui arbore binar.
Prin divizarea nodului N, ce are n elemente, folosind atributul A, obținem nodurile N_D și N_S ce au n_D și respectiv n_S elemente.
Informația câștigată prin partiționare este:

$$gain(A) = i(N) - \left(\frac{n_D}{n} \cdot i(N_D) + \frac{n_S}{n} \cdot i(N_S) \right)$$

informatie scontata

Termenul

$$E(A) = \frac{n_D}{n} \cdot i(N_D) + \frac{n_S}{n} \cdot i(N_S)$$

este cunoscut sub numele de *informația scontată*.

Suntem interesați ca informația câştigată să fie cât mai mare, acesta fiind criteriul alegerii atributului pentru partiționare.

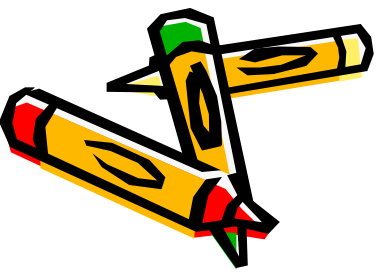
exemplu

- În perioada de vârf a virozelor respiratorii, elevii unei școli pot fi împărțiți în două categorii: bolnavi și sănătoși.

Descrierea va fi făcută pe baza a două atribute:

temperatura, atribut numeric

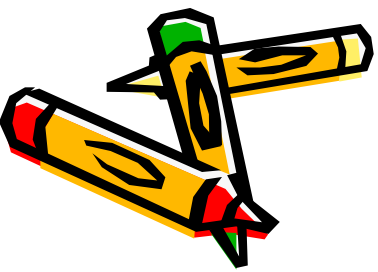
gât iritat, atribut nominal.



Considerăm un eşantion de 300 de pacienţi dintre care 200 sunt sănătoşi,

	gât iritat	gât normal
temperatura < 37.5	6 S , 37 B	191 S, 1 B
temperatura > 37.5	2 S, 21 B	1 S, 41 B

B (bolnavi) şi S (sănătoşi)



- calculăm entropia stării de sănătate:

$$i(stare) = -\frac{200}{300}\log_2\frac{200}{300} - \frac{100}{300}\log_2\frac{100}{300} = 0.9183\,;$$

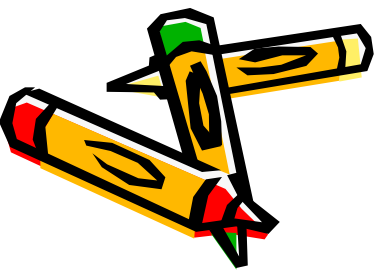
- calculăm entropia pentru temperatura > 37.5, respectiv temperatura < 37.5:

$$i(temp > 37,5) = -\frac{3}{65}\log_2\frac{3}{65} - \frac{62}{65}\log_2\frac{62}{65} = 0.2698\,,$$

$$i(temp < 37,5) = -\frac{197}{235}\log_2\frac{197}{235} - \frac{38}{235}\log_2\frac{38}{235} = 0.6384\,;$$

informația câștigată folosind atributul temperatură:

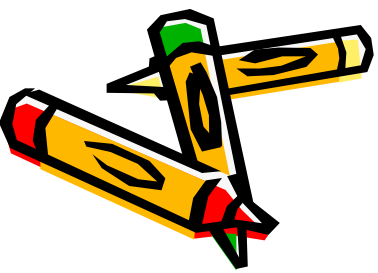
$$gain(temp) = 0.9183 - \frac{235}{300} \cdot 0.6384 - \frac{65}{300} \cdot 0.2698 = 0.3598$$



- calculăm entropia pentru gât iritat, respectiv gât neiritat:

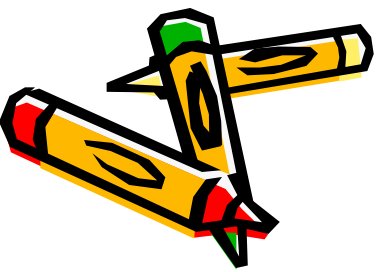
$$i(gat\ iritat) = -\frac{8}{66}\log_2\frac{8}{66} - \frac{58}{66}\log_2\frac{58}{66} = 0.5328\ ,$$

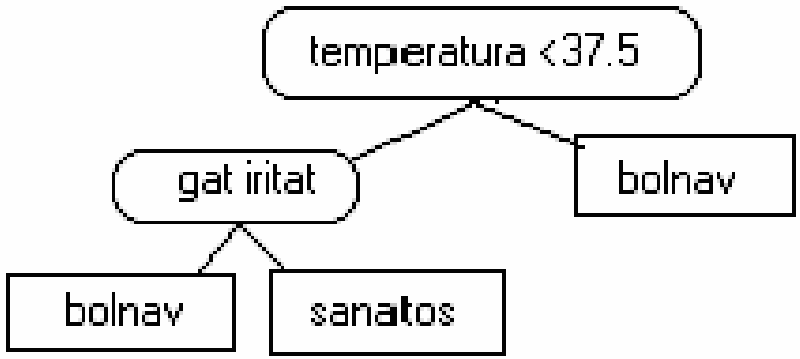
$$i(gat\ neiritat) = -\frac{192}{234}\log_2\frac{192}{234} - \frac{42}{234}\log_2\frac{42}{234} = 0.6790\ ;$$



- informația câștigată folosind atributul starea gâtului:

$$gain(stare\ gat) = 0.9183 - \frac{66}{300} \cdot 0.5328 - \frac{234}{300} \cdot 0.6790 = 0.2715$$

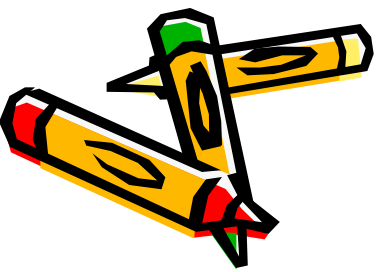




Frunzele nu sunt pure, dar au cea mai mică măsură de impuritate.

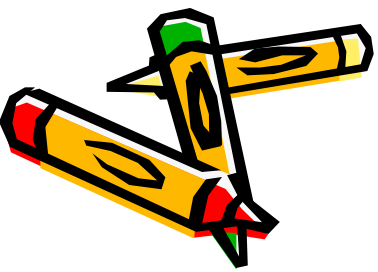
Putem avea impuritate din mai multe motive:

- date incorecte,
- date corecte dar atribute insuficiente,
- clasificarea implică un anumit grad de nedeterminare.



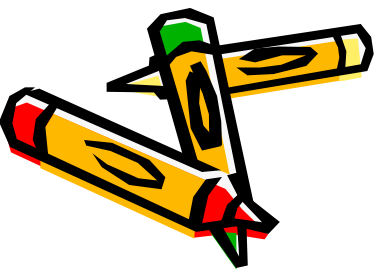
În cazul a trei clase Ω_1, Ω_2 și Ω_3 avem:

$$i(N) = -P(\Omega_1)\log_2 P(\Omega_1) - P(\Omega_2)\log_2 P(\Omega_2) -$$
$$- P(\Omega_3)\log_2 P(\Omega_3)$$

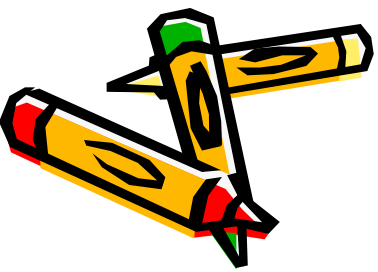


dacă în nodul N avem n elemente aparținând claselor Ω_1, Ω_2 și Ω_3, dintre care x în clasa Ω_1 și y în Ω_2, măsura entropiei în nodul N poate fi considerată a fi funcție de x și y:

$$f(x,y) = -\frac{x}{n}\log_2\frac{x}{n} - \frac{y}{n}\log_2\frac{y}{n} - \left(1-\frac{x+y}{n}\right)\cdot\log_2\left(1-\frac{x+y}{n}\right).$$

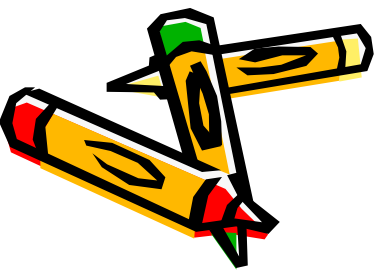


Calculăm extremele acestei funcții și obținem că $\left(\frac{n}{3}, \frac{n}{3}\right)$ este punct de maxim și astfel maximul impurității entropiei este 1.5850



Prin divizarea nodului N, ce are n elemente, folosind atributul A, obținem nodurile $N_1,\ldots,N_p$, ce au $n_1,\ldots,n_p$ elemente. Informația câștigată prin partiționare este:

$$gain(A) = i(N) - \sum_{k=1}^{p} \frac{n_k}{n} \cdot i(N_k)$$



exemplu

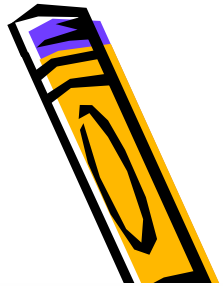
- În domeniul bancar, estimarea riscului acordării unui credit unei anumite persoane:

construim un arbore de clasificare și decizie, având în vedere următoarele proprietăți:

- comportamentul anterior al persoanei când a beneficiat de credite (istoria creditelor),
- datoria curentă,
- venit lunar
- garanții.

multimea de antrenament

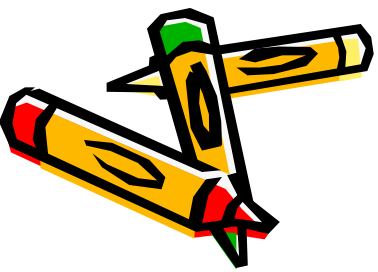
	RISC	Istoria creditelor	Datorii	Garanții	Venit RON
1	înalt	proastă	multe	nu există	400-1000
2	înalt	necunoscută	multe	nu există	1000-2000
3	moderat	necunoscută	puține	nu există	1000-2000
4	înalt	necunoscută	puține	nu există	400-1000
5	scăzut	necunoscută	puține	nu există	peste 2000
6	scăzut	necunoscută	puține	adecvate	peste 2000
7	înalt	proastă	puține	nu există	400-1000
8	moderat	proastă	puține	nu există	peste 2000
9	scăzut	bună	puține	nu există	peste 2000



10	scăzut	bună	multe	adecvate	peste 2000
11	înalt	bună	multe	nu există	400-1000
12	moderat	bună	multe	nu există	1000-2000
13	scăzut	bună	multe	nu există	peste 2000
14	înalt	proastă	multe	nu există	1000-2000
15	înalt	necunoscută	multe	nu există	1000-2000
16	moderat	necunoscută	puține	nu există	1000-2000
17	moderat	proastă	puține	adecvate	1000-2000
18	scăzut	necunoscută	puține	adecvate	peste 2000
19	scăzut	bună	puține	adecvate	400-1000
20	înalt	proastă	multe	nu există	400-1000

- calculăm măsura entropiei:

$$i(risc) = -\frac{8}{20}\log_2\frac{8}{20} - \frac{5}{20}\log_2\frac{5}{20} - \frac{7}{20}\log_2\frac{7}{20} = 1.5589\ ;$$



- calculăm câștigul de informație, obținut prin utilizarea atributului *istoria creditelor*, pentru divizarea nodului:

$$i(ist.proasta) = -\frac{4}{6}\cdot\log_2\frac{4}{6} - \frac{2}{6}\cdot\log_2\frac{2}{6} = 0.9183\ ,$$

$$i(ist.necunoscuta) = -\frac{3}{8}\cdot\log_2\frac{3}{8} - \frac{2}{8}\cdot\log_2\frac{2}{8} - \frac{3}{8}\cdot\log_2\frac{3}{8} =$$

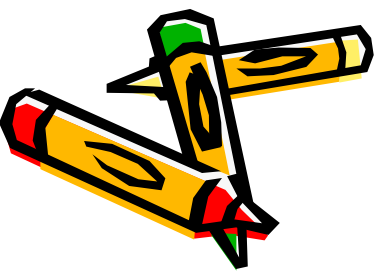
$$= 1.5613,$$

$$i(ist.buna) = -\frac{1}{6}\cdot\log_2\frac{1}{6} - \frac{1}{6}\cdot\log_2\frac{1}{6} - \frac{4}{6}\cdot\log_2\frac{4}{6} = 1.2516\ ,$$

$$gain(ist.credite) = i(risc) - \frac{6}{20} \cdot i(ist.proasta) -$$

$$- \frac{8}{20} \cdot i(ist.necunoscuta) - \frac{6}{20} \cdot i(ist\ buna) =$$

$$= 1.5589 - \left(\frac{6}{20} \cdot 0.9183 + \frac{8}{20} \cdot 1.5613 + \frac{6}{20} \cdot 1.2516 \right) = 0.2834\ ;$$



- calculăm câştigul de informaţie, obţinut prin utilizarea atributului *datorii*:

$$i(\textit{datorii multe}) = -\frac{6}{9}\cdot\log_2\frac{6}{9} - \frac{1}{9}\cdot\log_2\frac{1}{9} - \frac{2}{9}\cdot\log_2\frac{2}{9} = 1.2244,$$

$$i(\textit{datorii putine}) = -\frac{2}{11}\cdot\log_2\frac{2}{11} - \frac{4}{11}\cdot\log_2\frac{4}{11} - \frac{5}{11}\cdot\log_2\frac{5}{11} =$$

$$= 1.4949\ ,$$

$$gain(\textit{datorii}) = 1.5589 - \frac{9}{20}\cdot 1.2244 - \frac{11}{20}\cdot 1.4949 = 0.1857\ ;$$

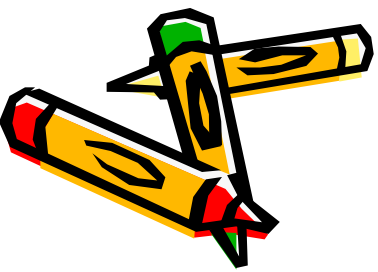
- câștigul de informație pe baza atributului *garanții* :

  $gain(garantii) = 0.1966$;

- câștigul de informație pe baza atributului *venit*:

  $gain(venit) = 0.8120$.

In nodul rădăcină vom avea atributul ***venit***, pentru care am obținut cel mai mare câștig de informație.



Studiem ce atribut utilizăm în subnodul corespunzător celor cu venit între 400-1000 RON, calculând câştigurile de informaţie corespunzătoare:

$$i(venit\ 400-1000)=0.65\ ,$$

$$i(ist\ proasta)=0\ ,\ i(ist.necunoscuta)=0\ ,\ i(ist\ buna)=1\ ,$$

$$gain(ist\ credite)=0.65-\frac{1}{3}=0.3167\ ;$$

Avem $gain(datorii)=0.1909$ şi $gain(garantii)=0.65$ şi astfel în acest subnod atributul ales va fi ***garanţii***.

Pentru cei ce au un venit lunar cuprins între 1000-2000 RON:

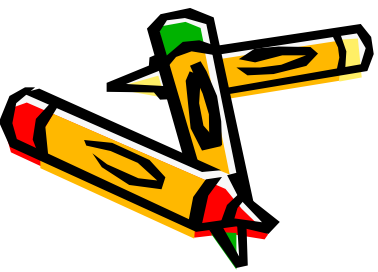
$$i(venit\,1000-2000)=0.9852\ ,$$

$$i(ist\ proasta)=1\ ,\ i(ist.necunoscuta)=0\ ,\ i(ist\ buna)=1\ ,$$

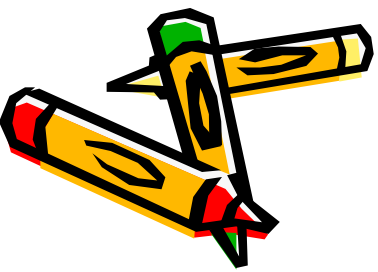
şi astfel $gain(ist\ credite)=0.9852-\frac{4}{7}-\frac{2}{7}=0.1281$.

Având $gain(datorii)=0.5216$ şi $gain(garantii)=0.1281$,

îi vom asocia subnodului atributul *datorii*.



Cei cu datorii puține prezintă un risc moderat pentru bancă, în schimb în cazul celor cu datorii multe, creăm un subnod căruia îi atribuim întrebarea legată de ***istoria creditelor*** avute.

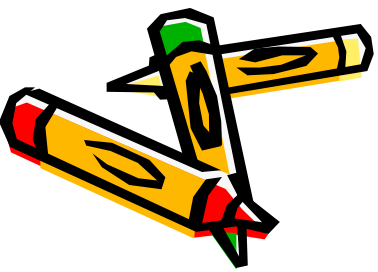


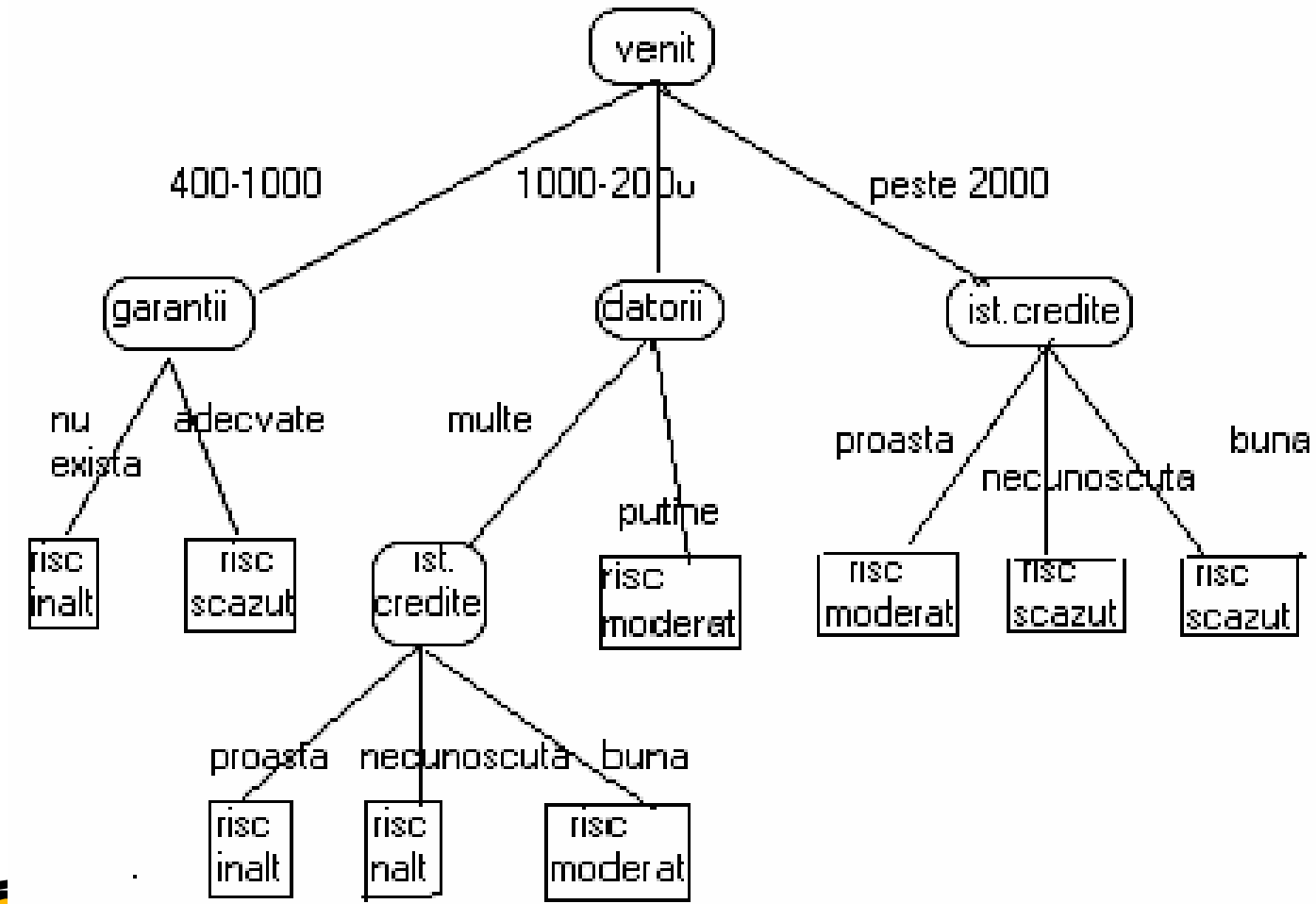
Pentru cei cu venit mai mare de 2000 RON avem următoarele câştiguri de informaţie:

$$gain(ist\ credite) = 0.8120\ , gain(datorii) = 0.4184\ ,$$

$$gain(garantii) = 0.3484\ ,$$

atributul fiind astfel ***istoria creditelor***.




venit
400-1000
1000-2000
peste 2000
garantii
datorii
ist.credite
nu exista
adecvate
multe
putine
proasta
necunoscuta
buna
risc inalt
risc scazut
ist. credite
risc moderat
risc moderat
risc scazut
risc scazut
proasta
necunoscuta
buna
risc inalt
risc nalt
risc moderat

masura de impuritate Gini

Altă metodă de definire a impurității unui nod, "***măsura de impuritate Gini***", dată de:

$$i_G(N) = \sum_{i \neq j} P(\Omega_i) \cdot P(\Omega_j) = \left(\sum_{i=1}^{n} P(\Omega_i) \right)^2 - \sum_{i=1}^{n} P^2(\Omega_i) =$$

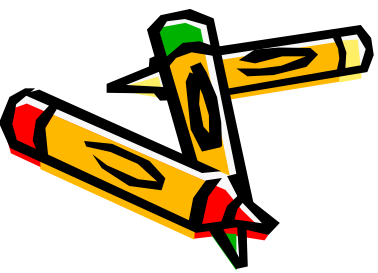
$$= 1 - \sum_{i=1}^{n} P^2(\Omega_i)$$

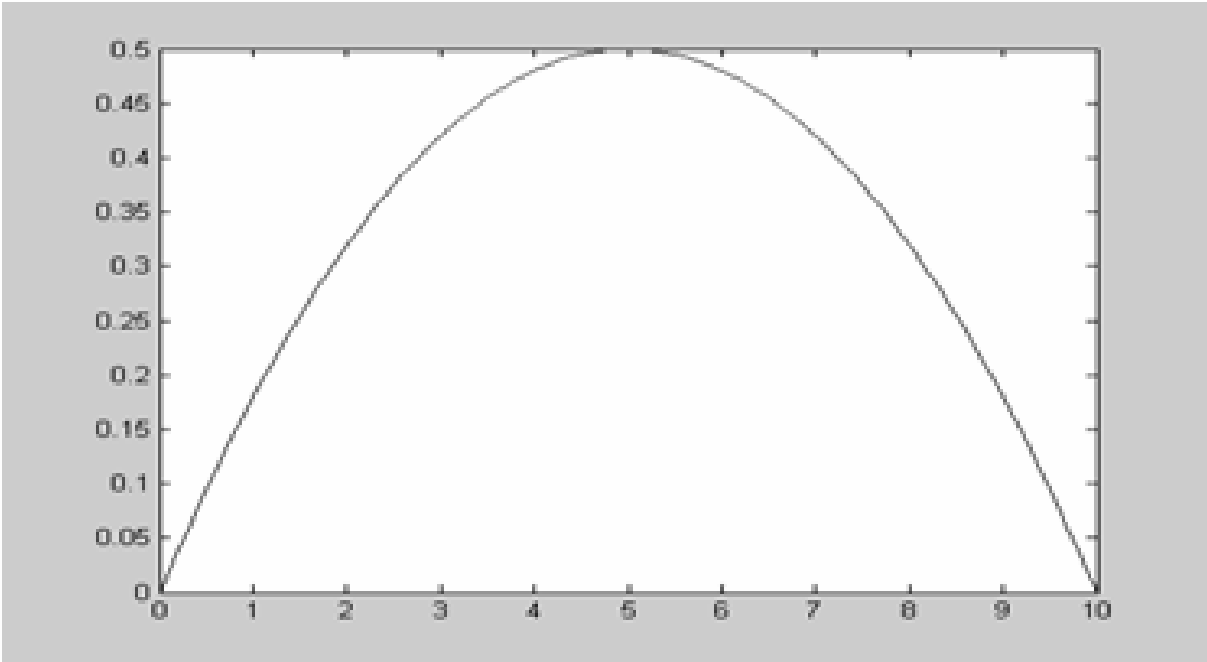
Dacă în mulțimea de antrenament există doar două clase Ω_1 și Ω_2, avem:

$$i_G(N) = 1 - P^2(\Omega_1) - P^2(\Omega_2),$$

și anume dacă în nodul N avem n elemente aparținând claselor Ω_1 și Ω_2, dintre care x din clasa Ω_1, impuritatea Gini în nodul N poate fi considerată a fi funcție de x

$$f(x) = 1 - \left(\frac{x}{n}\right)^2 - \left(1 - \frac{x}{n}\right)^2$$



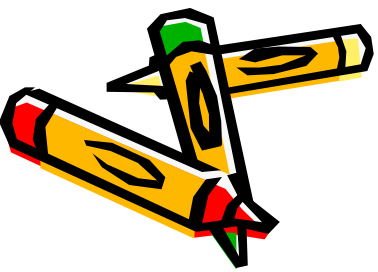


Valoarea maximă a măsurii Gini în acest caz este 1.

Prin divizarea nodului N, ce are n elemente, folosind atributul A, obținem nodurile N_D și N_S ce au n_D și respectiv n_S elemente. Indexul Gini de partiționare este:

$$Gini_{split}(N) = \left(\frac{n_D}{n} \cdot i_G(N_D) + \frac{n_S}{n} \cdot i_G(N_S) \right)$$

Cea mai mică valoare a indexului Gini de partiționare ne dă acel atribut care minimizează impuritatea divizării,



exemplu

- În exemplul anterior, cu viroze respiratorii, avem:

$$i_G(temp < 37{,}5) = 1 - \left(\frac{197}{235}\right)^2 - \left(\frac{38}{235}\right)^2 = 0.2711\ ,$$

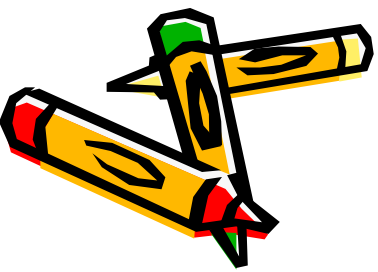
$$i_G(temp > 37{,}5) = 1 - \left(\frac{3}{65}\right)^2 - \left(\frac{62}{65}\right)^2 = 0.0880\ ,$$

$$Gini_{split}(temperatura) = \frac{235}{300} \cdot 0.2711 + \frac{65}{300} \cdot 0.0889 = 0.2314$$

indicele de partiționare Gini pentru *starea de iritare* a gâtului:

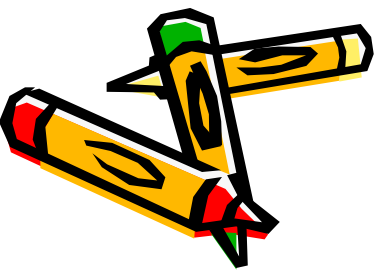
$$Gini_{split}\,(stare\ gat) = \frac{66}{300}\cdot 0.2130 + \frac{234}{300}\cdot 0..2945 = 0.3649\,.$$

utilizând această măsură de impuritate, decidem să facem prima partiționare cu atributul *temperatura*, la fel ca în cazul precedent.



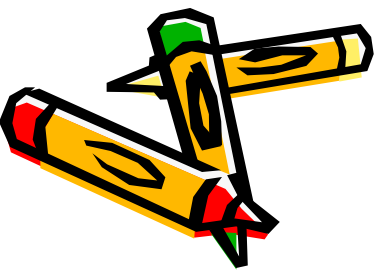
Dacă prin divizarea nodului N, ce are n elemente, folosind atributul A, obținem nodurile $N_1,\ldots,N_p$, ce au $n_1,\ldots,n_p$ elemente, indicele de partiționare Gini pentru atributul A din nodul N este:

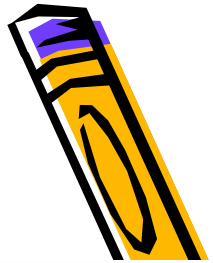
$$Gini_{split}(A) = \sum_{k=1}^{p} \frac{n_k}{n} \cdot i_G(N_k).$$



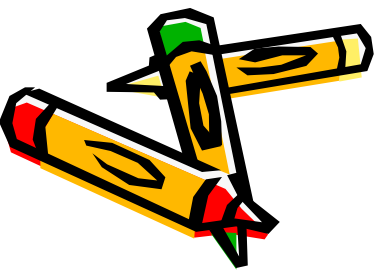
exemplu

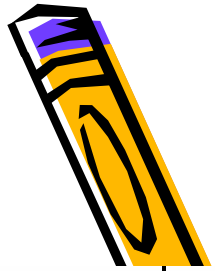
- O agenție de turism dorește să realizeze profilul clientului ce alege să petreacă prin intermediul ei concediul în țară sau în străinătate, folosind ca atribute: *vârsta*, *starea civilă* (căsătorit/necăsătorit), *venitul lunar* (sub 1500 RON/peste 1500 RON), *studiile* (superioare/medii).



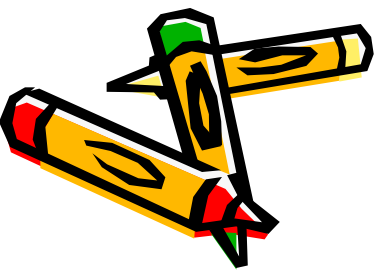


	Destinația	Vârsta	Stare civilă	Venit	Studii
1	țară	27	căsătorit	<1500	medii
2	străinătate	29	necăsătorit	>1500	superioare
3	țară	52	căsătorit	<1500	medii
4	străinătate	58	necăsătorit	>1500	superioare
5	țară	30	necăsătorit	<1500	medii
6	țară	39	căsătorit	<1500	medii
7	țară	60	căsătorit	<1500	medii
8	țară	51	căsătorit	>1500	superioare
9	străinătate	24	necăsătorit	<1500	superioare
10	țară	22	necăsătorit	< 1500	medii





11	străinătate	64	căsătorit	>1500	superioare
12	străinătate	61	căsătorit	> 1500	superioare
13	înstrăinătate	29	căsătorit	> 1500	medii
14	țară	65	căsătorit	<1500	medii
15	țară	45	necăsătorit	< 1500	medii
16	străinătate	32	necăsătorit	>1500	medii
17	străinătate	34	căsătorit	< 1500	superioare
18	străinătate	38	necăsătorit	<1500	medii
19	țară	49	căsătorit	<1500	medii
20	țară	32	necăsătorit	< 1500	medii
21	țară	48	căsătorit	> 1500	superioare



Vom împărți mulțimea pe categorii de vârstă: mai mică, respectiv mai mare decât 25, 35, 45, 55, căutând valoarea optimă de partiționare prin utilizarea indexului Gini.

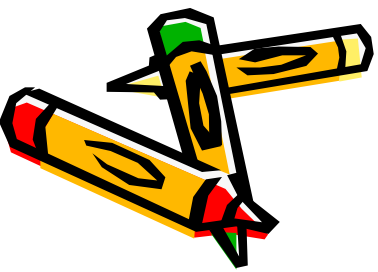
	concediu în țară	concediu în străinătate
vârsta < 25	1	1
vârsta > 25	11	8

$$i_G(virsta < 25) = 0.5\ ;\ i_G(virsta > 25) = 0.4875\ ;$$

$$Gini_{split}(virsta = 25) = \frac{2}{21} \cdot 0.5 + \frac{19}{21} \cdot 0.4875 = 0.4887$$

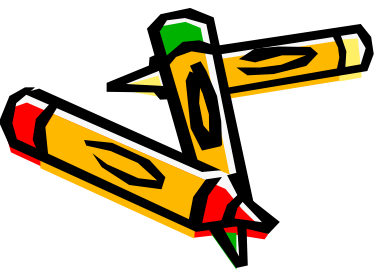
	concediu în țară	concediu în străinătate
vârsta < 35	4	5
vârsta > 35	8	4

$$Gini_{split}\,(virsta = 35) = \frac{9}{21} \cdot 0.4936 + \frac{12}{21} \cdot 0.4444 = 0.4656$$



	concediu în țară	concediu în străinătate
vârsta < 45	6	6
vârsta > 45	6	3

$$Gini_{split}\,(virsta = 45) = \frac{12}{21} \cdot 0.5 + \frac{9}{21} \cdot 0.4444 = 0.4762$$

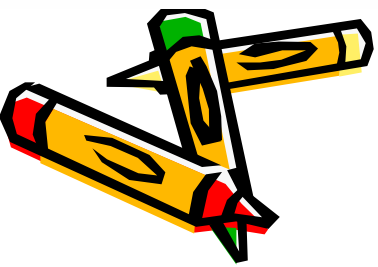


	concediu în țară	concediu în străinătate
vârsta < 55	7	6
vârsta > 55	3	3

$$Gini_{split}\,(virsta = 55) = \frac{13}{21} \cdot 0.4970 + \frac{9}{21} \cdot 0.4688 = 0.4863$$

Cea mai mică valoare a indicelui Gini de partiționare este pentru 35 de ani.
Se ia media aritmetică dintre această valoare și altă valoare de vârstă apropiată (adică 25 ani sau 45 ani), și anume pe cea care are indicele de partiționare mai mic, în cazul nostru 45.

În nodul rădăcină atributul va fi " *vârstă* < 40 ani".



În cazul clienților în vârstă mai mică de 40 ani avem:

	concediu în țară	concediu în străinătate
căsătorit	2	2
necăsătorit	3	4

$$Gini_{split}(stare\ \ civila) = \frac{4}{11} \cdot 0.5 + \frac{7}{11} \cdot 0.4898 = 0.4935\ ;$$

	concediu în țară	concediu în străinătate
peste 1500 RON	0	3
sub 1500 RON	5	3

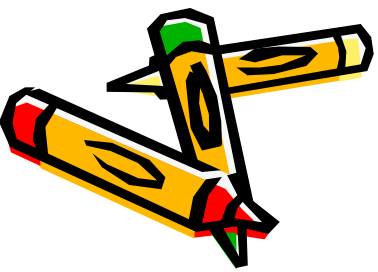
$$Gini_{split}(venit) = 0.3409;$$

	concediu în țară	concediu în străinătate
superioare	0	3
medii	5	3

$$Gini_{split}(studii) = 0.3409$$

Având aceeaşi valoare a indicelui de partiționare Gini pentru atributele *venit*, respectiv *studii*, alegem aleator, să zicem *studii superioare*.

Calculăm indicii de partiționare pentru a decide ce subnod alegem în cazul celor ce au vârsta sub 40 ani şi studii medii:



	concediu în țară	concediu în străinătate
peste 1500 RON	0	2
sub 1500 RON	5	1

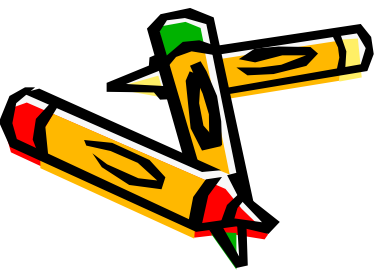
$Gini_{split}\ (venit) = 0.2084\ ;$

	concediu în țară	concediu în străinătate
căsătorit	2	1
necăsătorit	3	2

$Gini_{split}\ (stare\ civila) = 0.4666\ .$

Evident următorul atribut este *venit* < 1500 RON.

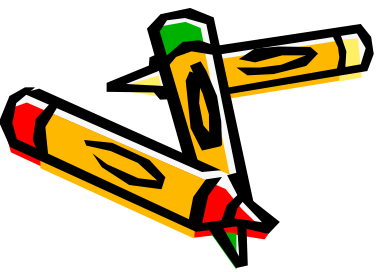
Să vedem dacă atributul *stare civilă* influențează puritatea frunzei "concediu în țară" în cazul clienților cu vârsta sub 40 ani., studii medii și venitul lunar sub 1500 RON.



	concediu în țară	concediu în străinătate
căsătorit	2	1
necăsătorit	3	0

Concluzie (pe baza mulțimii de antrenament):

Persoanele necăsătorite cu vârsta sub 40 ani, studii medii și venitul lunar sub1500 RON aleg să-și petreacă concediul în țară, în schimb doar 66% dintre cei căsătoriți din aceiași categorie aleg aceeași destinație.



- în cazul clienților în vârstă mai mare de 40 ani avem:

	concediu în țară	concediu în străinătate
căsătorit	7	2
necăsătorit	0	1

$$Gini_{split}\ (stare\ civila) = 0.3111\ ;$$

	concediu în țară	concediu în străinătate
peste 1500 RON	2	3
sub 1500 RON	5	0

$$Gini_{split}\ (venit) = 0.24\ ;$$

	concediu în țară	concediu în străinătate
superioare	2	3
medii	5	0

$$Gini_{split}\ (studii) = 0.2400\ .$$

Din nou avem aceeaşi valoare a indicelui de partiționare pentru atributele venit, respectiv studii, să alegem acum atributul venit < 1500 RON.

Din datele prezentate rezultă că aceia cu un asemenea venit lunar aleg să-şi petreacă concediul în țară.

vom calcula indicii de partiționare pentru clienții peste 40 ani, cu venit mai mare de 1500 RON:

	concediu în țară	concediu în străinătate
căsătorit	2	2
necăsătorit	0	1

$$Gini_{split}\ (stare\ civila) = 0.4\ ;$$

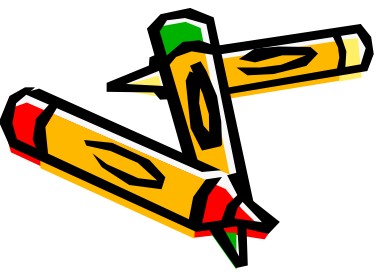
	concediu în țară	concediu în străinătate
superioare	2	3
medii	0	0

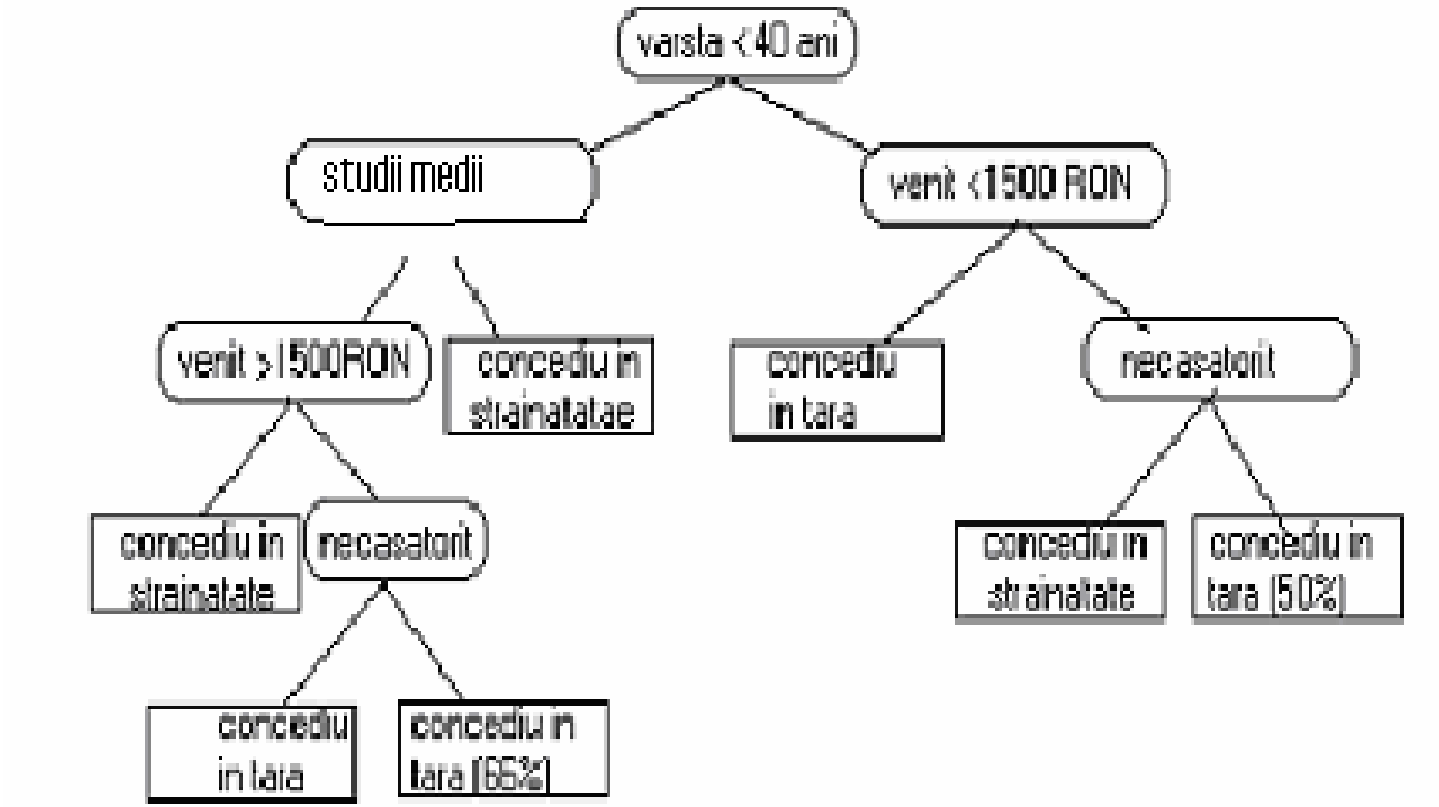
$$Gini_{split}\ (studii) = 0.4444\ .$$

Următorului subnod îi atribuim atributul *stare civilă* și anume necăsătorit.

Nu putem afirma nimic despre destinația de concediu a clienților agenției, în vârstă de peste 40 ani, cu venit mai mare de 1500 RON, căsătoriți, în raport cu nivelul studiilor acestora deoarece jumătate dintre ei aleg să-și petreacă concediul în țară.

În urmă acestui studiu, obținem următorul arbore de clasificare:

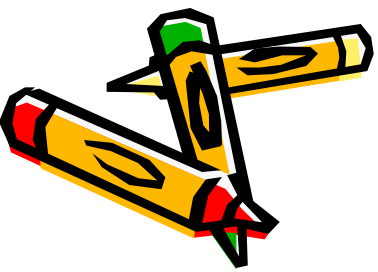



varsta <40 ani
studii medii
venit <1500 RON
venit >1500RON
concediu in strainatatae
concediu in tara
necasatorit
concediu in strainatate
necasatorit
concediu in strainatatie
concediu in tara (50%)
concediu in tara
concediu in tara (66%)

Cu ajutorul arborilor de clasificare şi decizie se pot formula reguli.

În exemplul prezentat, pe baza mulțimii de antrenament, construind arborele de clasificare, putem deduce din regulile obținute profilul clientului agenției, ce își petrece vacanța în țara sau străinătate.

Regulile deduse se bazează pe mulțimea de antrenament, în cazul nostru datele oferite de agenție.

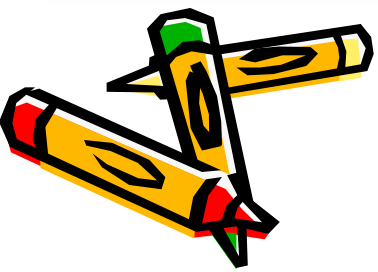


masura clasificarii gresite

măsura clasificării greşite (misclassification) se defineşte prin:

$$i_M(N) = 1 - \max_j P(\Omega_j)$$

Aceasta măsoară probabilitatea minimă ca un element din mulțimea de antrenament să fie greşit clasificat prin folosirea atributului A în nodul N.

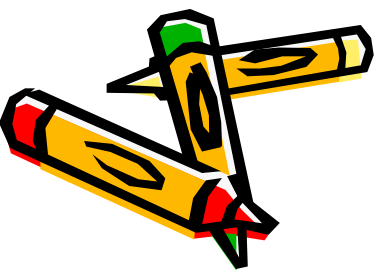


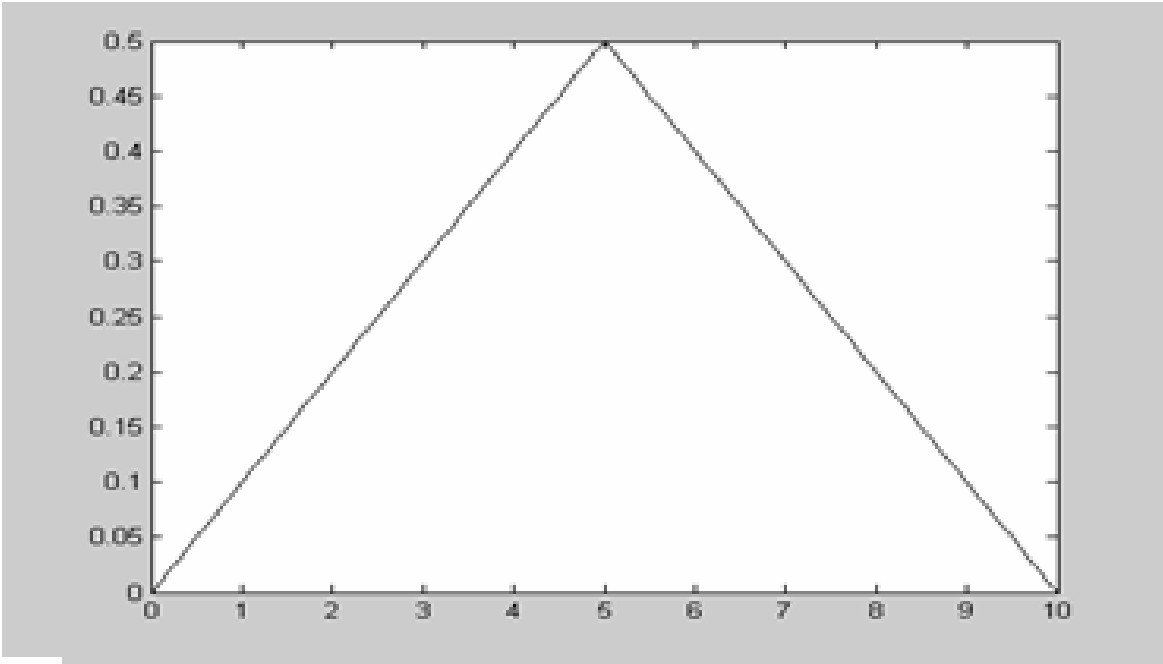
În cazul a două clase avem:

$$i_M(N) = 1 - \max\{P(\Omega_1), P(\Omega_2)\}$$

și anume dacă în nodul N avem n elemente aparținând claselor Ω_1 și Ω_2, dintre care x în clasa Ω_1, impuritatea entropiei în nod poate fi considerată a fi funcție de x:

$$f(x) = 1 - \max\{\frac{x}{n}, 1 - \frac{x}{n}\} = \begin{cases} \frac{x}{n}, & x > \frac{n}{2} \\ 1 - \frac{x}{n}, & x < \frac{n}{2} \end{cases}$$





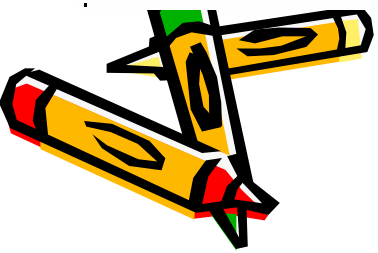
De reținut: cele trei funcții ale impurității, definite în cazul a două clase, au aceeași valoare a abscisei punctului de maxim.

exemplu

- Folosind impuritatea clasificării greşite, să conturăm profilul clientului supermarketului vs. clientul micului magazin din colțul străzii sau de la parterul blocului, utilizând următoarele atribute:

- *posesor sau nu automobil*,
- *venit* lunar (mai mic, respectiv mai mare de 1000 RON),
- *salariat/pensionar*

	client	venit	posesor auto	salariat/ pensionar
1	supermarket	peste 1000	da	salariat
2	supermarket	sub 1000	nu	salariat
3	mic magazin	sub 1000	nu	pensionar
4	mic magazin	sub 1000	nu	salariat
5	supermarket	sub 1000	da	salariat
6	supermarket	sub 1000	da	pensionar
7	mic magazin	peste 1000	da	pensionar
8	mic magazin	sub 1000	nu	pensionar
9	supermarket	peste 1000	nu	salariat
10	supermarket	sub 1000	nu	salariat
11	supermarket	sub 1000	nu	salariat



12	supermarket	peste 1000	da	salariat
13	supermarket	peste 1000	da	pensionar
14	supermarket	sub 1000	da	pensionar
15	mic magazin	sub 1000	nu	pensionar
16	mic magazin	peste 1000	nu	pensionar
17	supermarket	sub 1000	nu	salariat
18	supermarket	sub 1000	da	salariat
19	supermarket	peste 1000	da	salariat
20	supermarket	sub 1000	da	salariat
21	mic magazin	sub 1000	nu	pensionar
22	mic magazin	sub 1000	nu	salariat
23	mic magazin	sub 1000	da	pensionar
24	supermarket	peste 1000	da	salariat
25	supermarket	sub 1000	nu	salariat

Să decidem care va fi nodul rădăcină, folosind măsura de clasificare greşită:

	supermarket	mic magazin
venit lunar peste 1000 RON	6	2
venit lunar sub 1000 RON	10	7

$$i_M(peste\,1000) = 1 - \max\{\frac{6}{8}, \frac{2}{8}\} = 0.25\ ;$$

$$i_M(sub\,1000) = 1 - \max\{\frac{10}{17}, \frac{7}{17}\} = 0.4118\ ;$$

$$i_{split}(venit) = \frac{8}{25} \cdot 0.25 + \frac{17}{25} \cdot 0.4118 = 0.36$$

	supermarket	mic magazin
posesor auto	10	6
nu posedă automobil	7	2

$$i_{split}(auto) == 0.3198\ ,$$

	supermarket	mic magazin
salariat	13	2
pensionar	7	3

$$i_{split}(salariat\ /\ pensionar) == 0.200\ .$$

Conform celui mai mic indice de partiționare, în nodul rădăcină va fi atributul salariat.

Pentru salariați avem:

	supermarket	mic magazin
venit lunar peste 1000 RON	5	0
venit lunar sub 1000 RON	9	1

$$i_{split}\,(venit) = 0.0667\ ;$$

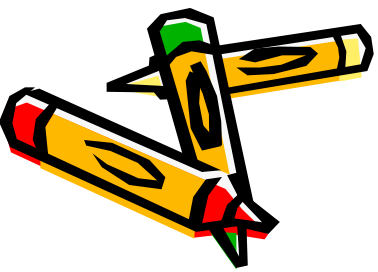
	supermarket	mic magazin
posesor auto	7	0
nu posedă automobil	6	2

$$i_{split}\,(auto) == 0.1778\ .$$

Următorului subnod îi atribuim atributul *venit* lunar sub 1000 RON.

Pentru salariați cu venit sub 1000 RON avem următoarea situație:

	supermarket	mic magazin
posesor auto	3	0
nu posedă automobil	3	2



Să studiem alegerea pe care o fac pensionarii:

	supermarket	mic magazin
venit lunar peste 1000 RON	1	2
venit lunar sub 1000 RON	2	5

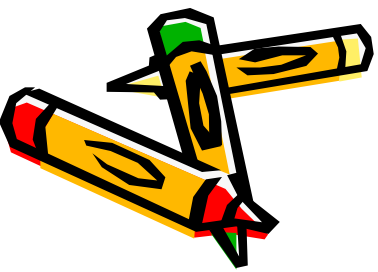
$$i_{split}\,(venit) = 0.300\,;$$

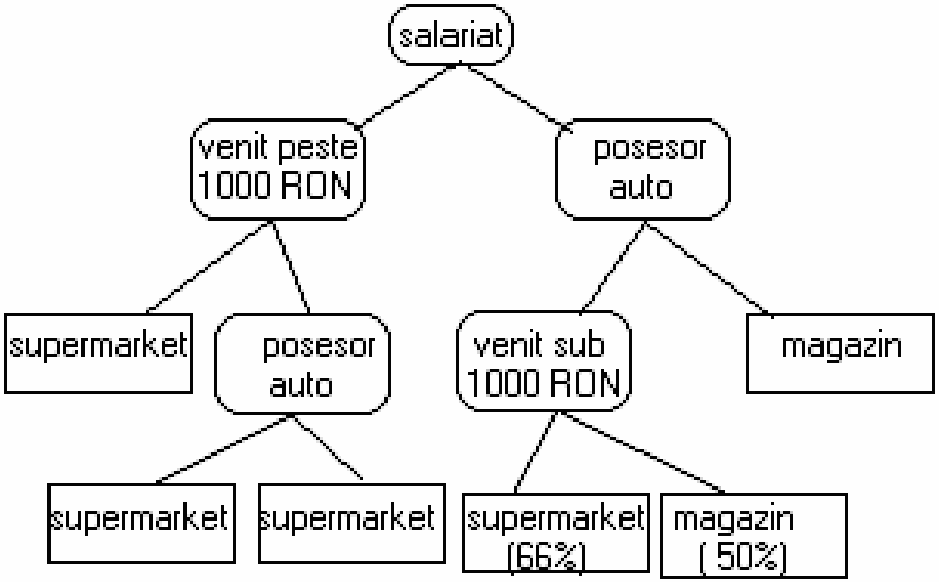
	supermarket	mic magazin
posesor auto	3	2
nu posedă automobil	0	5

$$i_{split}\,(auto) == 0.200\,;$$

Următorul subnod va avea atributul *posesor auto*.

	supermarket	mic magazin
venit lunar peste 1000 RON	1	1
venit lunar sub 1000 RON	2	1

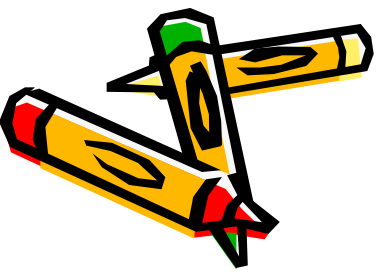




Regulile deduse din acesta, alcătuiesc profilul cumpărătorului din supermarket, respectiv din micul magazin.

Regulile deduse din acesta, alcătuiesc profilul cumpărătorului din supermarket, respectiv din micul magazin.
Teoretic, se afirmă că uneori indexul Gini descreşte, în timp ce măsura clasificării greşite nu.
Pe de altă parte, măsura Gini anticipează noile ramificaţii.

Intervine vreo modificare în exemplul nostru, folosind indicele Gini de partiţionare?



$$Gini_{split}\ (venit) = 0.3350;$$

$$Gini_{split}\ (posesor\ auto) = 0.2841;$$

$$Gini_{split}\ (salariat\ /\ pensionar) = 0.3067;$$

Nodul rădăcină va fi în acest caz posesor auto;

pentru cumpărătorii posesori auto avem:

	supermarket	mic magazin
salariat	7	0
pensionar	3	2

$$Gini_{split}(salariat\ /\ pensionar) = 0.200;$$

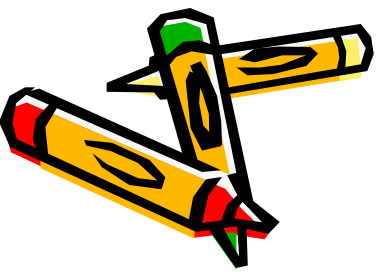
	supermarket	mic magazin
venit lunar peste 1000 RON	5	0
venit lunar sub 1000 RON	5	2

$$Gini_{split}(venit) = 0.2381\,.$$

- Subnodului ce urmează îi atribuim proprietatea salariat.

în cazul pensionarilor, posesori auto:

	supermarket	mic magazin
venit lunar peste 1000 RON	1	1
venit lunar sub 1000 RON	2	1



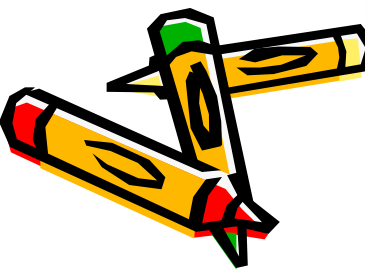
Ce se întâmplă în cazul celor ce nu au automobil:

	supermarket	mic magazin
salariat	6	2
pensionar	0	5

$$Gini_{split}(salariat / pensionar) = 0.200$$

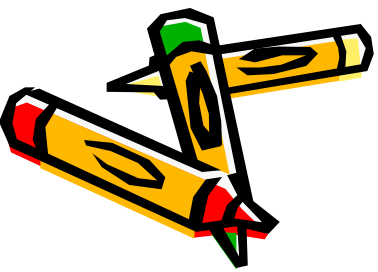
	supermarket	mic magazin
venit lunar peste 1000 RON	1	1
venit lunar sub 1000 RON	5	6

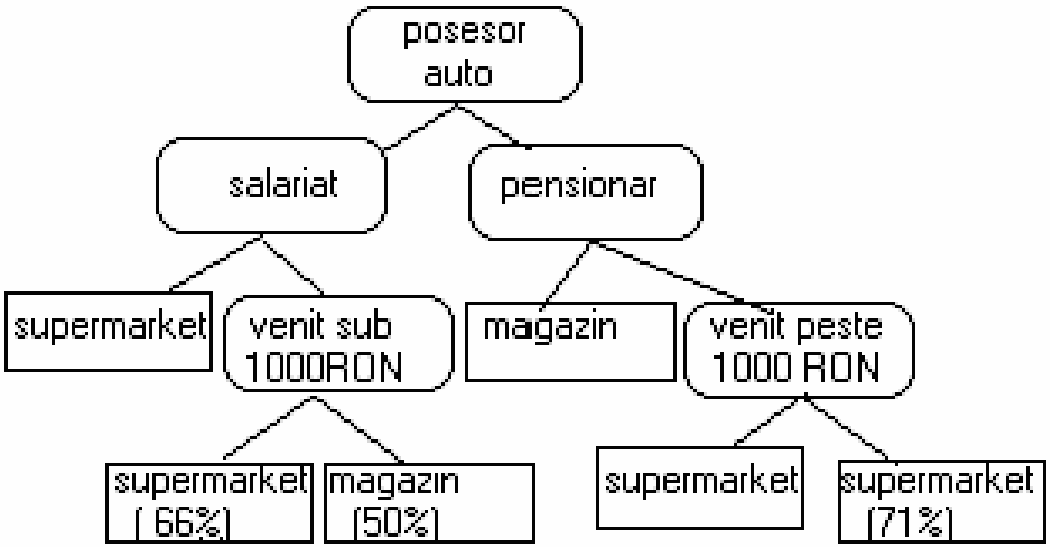
$$Gini_{split}(venit) = 0.4965$$



Situația cumpărătorilor salariați, ce nu au mașină este:

	supermarket	mic magazin
venit lunar peste 1000 RON	1	0
venit lunar sub 1000 RON	5	2



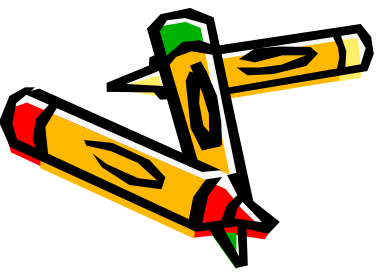


Puritatea frunzei „supermarket" este mai bună, cum arată un calcul simplu, dar regulile deduse, ce descriu profilul cumpărătorului, sunt aceleaşi.

Să folosim câştigul de informaţie pentru aceeaşi mulţime de antrenament.

Calculăm măsura entropiei tipului de magazin: supermarket/mic magazin:

$$i(\textit{magazinul ales}) = -\frac{16}{25}\cdot\log_2\frac{16}{25} - \frac{9}{25}\cdot\log_2\frac{9}{25} = 0.9427\ .$$



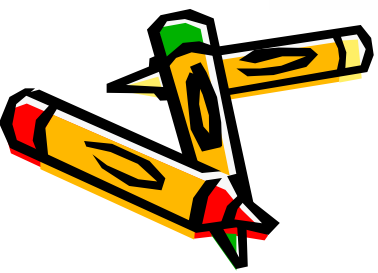
calculăm câştigul de informaţie folosind atributul ***venit***:

$$i(venit<1000) = -\frac{10}{17}\cdot\log_2\frac{10}{17} - \frac{7}{17}\cdot\log_2\frac{7}{17} = 0.9774;$$

$$i(venit>1000) = -\frac{6}{8}\cdot\log_2\frac{6}{8} - \frac{2}{8}\cdot\log_2\frac{2}{8} = 0.8113.$$

$$gain(venit) = i(magazinales) - \frac{17}{25}\cdot i(venit<1000) -$$

$$-\frac{8}{25}\cdot i(venit>1000) = 0.0184.$$



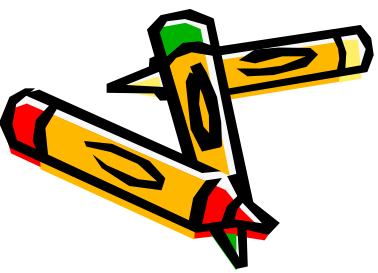
calculăm câștigul de informație folosind atributul *posesor auto*:

$$i(posesor\ \ auto - DA) = -\frac{10}{12}\cdot \log_2 \frac{10}{12} - \frac{2}{12}\cdot \log_2 \frac{2}{12} = 0.6500\ ;$$

$$i(posesor\ \ auto - NU) = -\frac{6}{13}\cdot \log_2 \frac{6}{13} - \frac{7}{13}\cdot \log_2 \frac{7}{13} = 0.9957\ \ ;$$

$$gain(posesor\ auto) = i(magazin\ ales) - \frac{12}{25}\cdot i(posesor\ auto - DA) -$$

$$- \frac{13}{25}\cdot i(posesor\ auto - NU) = 0.1129\ .$$



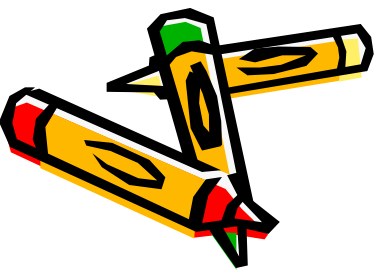
calculăm câştigul de informaţie folosind atributul salariat /pensionar:

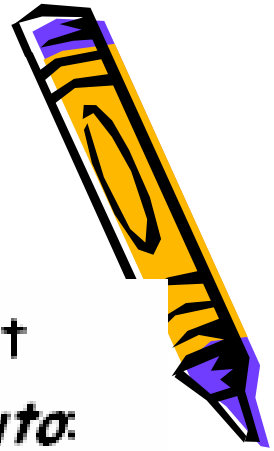
$$i(salariat) = 0.5665\ ;\ i(pensionar) = 0.8813\ ;$$

$$gain(salariat - pensionar) = 0.9427 - \frac{15}{25} \cdot 0.5665 - \frac{10}{25} \cdot 0.8813 =$$

$$= 0.2503\ .$$

Stabilim cel mai mare câştig de informaţie şi astfel în nodul rădăcină atributul va fi ***salariat***.





calculăm pentru salariat câştigul de informaţie obţinut prin folosirea atributelor ***venit***, respectiv ***posesor auto***:

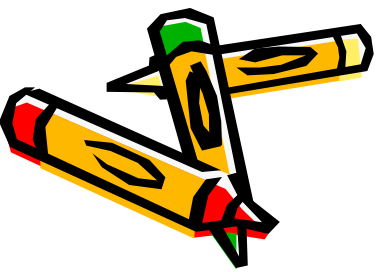
$$i(venit > 1000) = 0;\ i(venit < 1000) = 0.7219;$$

$$gain(venit) = 0.5665 - \frac{10}{15} \cdot 0.7219 = 0.0852;$$

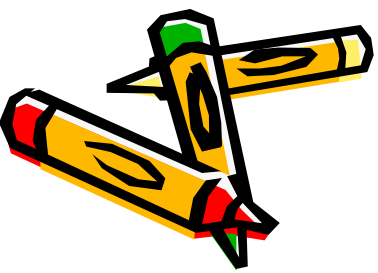
$$i(posesor\ \ auto - da) = 0;\ i(posesor\ \ auto - nu) = 0.8113;$$

$$gain(posesor\ \ auto) = 0.5665 - \frac{8}{15} \cdot 0.8113 = 0.1338.$$

În concluzie, în acest subnod atributul va fi ***posesor auto***;



dacă nu este posesor auto vom considera un subnod cu atributul venit >1000RON.



dacă este pensionar, calculăm câștigul de informație pentru ***venit***, respectiv pentru ***posesor auto***:

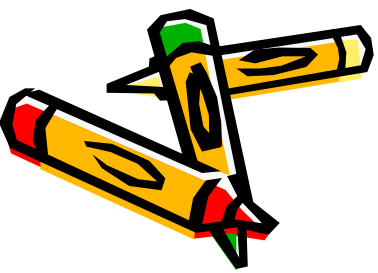
$$i(venit<1000)=0.8631;\ i(venit>1000)=0.9183;$$

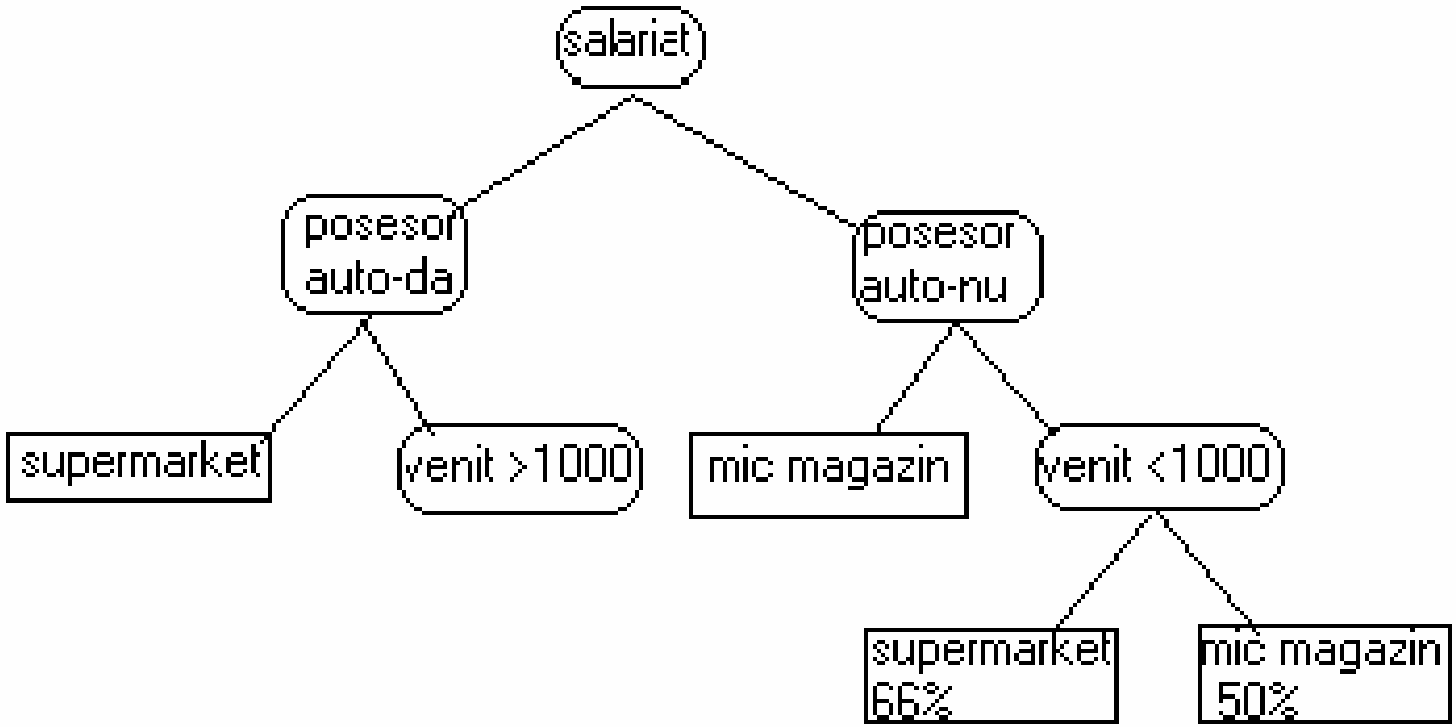
$$gain(venit)=0.8813-\frac{7}{10}\cdot 0.8631-\frac{3}{10}\cdot 0.9183=0.0016;$$

$$i(posesor\quad auto-da)=0.9710;\ i(posesor\quad auto-da)=0.9710;$$

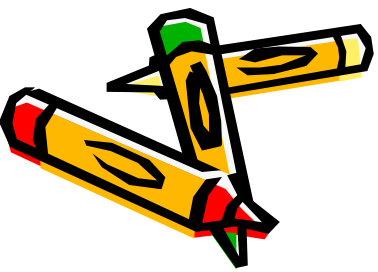
$$gain(posesor\quad auto)=0.8813-\frac{5}{10}\cdot 0.9710=0.3958$$

și astfel atributul va fi ***posesor auto***.




salariat
posesor auto-da
posesor auto-nu
supermarket
venit >1000
mic magazin
venit <1000
supermarket 66%
mic magazin 50%

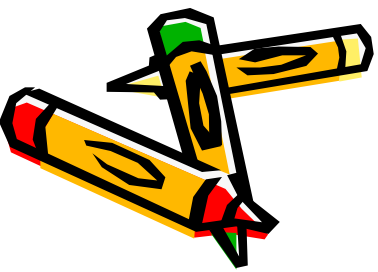
Scopul construcției arborilor de clasificare și decizie:
a obține o predicție cât mai precisă.
Costurile predicției sunt indicatori ai acurateței acesteia.
Consecințele unei clasificări eronate sunt deosebit de importante.



exemplu

în cazul în care un medic greşeşte diagnosticul benign/malign al unei tumori, ce este mai grav :

- tumoarea malignă să fie catalogată drept benignă?
- tumoarea benignă să fie considerată a fi malignă?



prior probabilities

Principalele costuri legate de procesul de clasificare sunt:

- *Probabilitățile prealabile* (*prior probabilities*) sunt acei parametri care specifică probabilitatea ca un obiect să aparțină unei anumite clase.

De obicei, se aleg acei parametri proporționali cu numărul de obiecte din fiecare clasă.

missclassification costs

- *Costuri de clasificare greşită* (*misclassification costs*) se referă la faptul că, în procesul de clasificare, unele categorii au nevoie de o clasificare mai precisă.

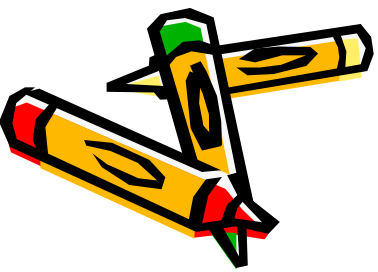
Reluând exemplul cu diagnosticul unei tumori, este mult mai importantă acuratețea clasificării uneia ca fiind malignă, decât ca fiind benignă.

Costurile de clasificare greşită sunt alese astfel încât să reflecte importanța fiecărei clase.

Problema alegerii criteriului de oprire a procesului de divizare a nodurilor.

Procesul de partiționare se derulează până când toate nodurile terminale -frunzele- sunt *pure*, adică vor conține numai elemente din aceeași categorie.

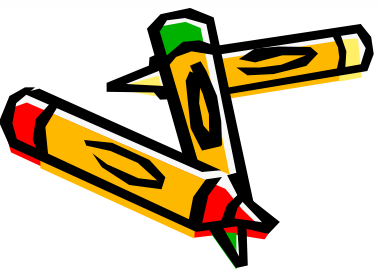
Este important ca pe mulțimea de testare/validare clasificatorul să aibă performanță maximă.



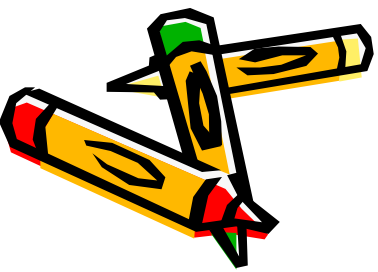
stop

Uzual, sunt utilizate două reguli de *Stop*:

- *Minimul n*: condiția de *Stop* specifică un număr minim de obiecte care să fie conținute în nodurile terminale. Divizarea unui nod ia sfârșit atunci când fie nodul este pur, fie conține numărul minim de obiecte.

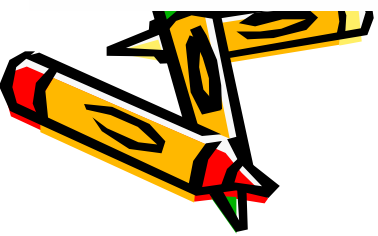


- *Proporția de obiecte*: condiția de *Stop* impune ca divizarea unui nod să ia sfârșit atunci când nodul este pur sau conține un procentaj minim de obiecte dintr-o anumită clasă.



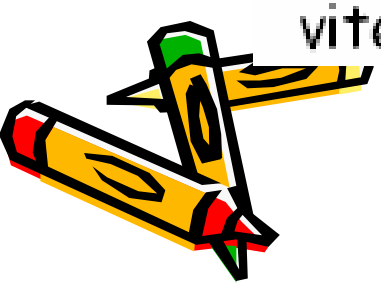
overfitting/underfitting

Un arbore fiind construit pentru a putea fi aplicat la diverse seturi de date, este necesară evitarea unei potriviri prea accentuate (*overfitting*) cu mulțimea pe care s-a făcut antrenamentul.
Când arborele este prea simplu față de datele utilizate la antrenament și, în consecință, atât eroarea de antrenament cât și cea de testare sunt mari, avem de-a face cu situația sub-potrivire (*underfitting*) a arborelui cu datele.



pruning

Overfitting-ul este cel mai adesea întâlnit.
În acest caz se utilizează metodă de fasonare (*pruning*) a arborelui.
Se utilizează metode statistice pentru îndepărtarea ramurilor nesemnificative, redundante, sau care nu urmează pattern-ul general al datelor, obținând astfel un arbore mai puțin „stufos", cu o mai mare viteză de clasificare.



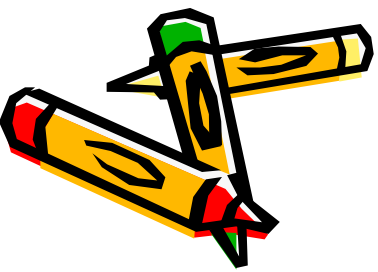
pre-pruning

Există două tipuri de fasonare a unui arbore de clasificare și decizie:

- *Fasonarea prealabilă* (*pre-pruning*): se oprește practic „creșterea" arborelui în timpul procesului de inducție, prin decizia de a se sista divizarea nodului, astfel încât acesta va deveni o „frunză", etichetată cu numele clasei cu cele mai multe elemente.

Principalele condiții de stopare sunt:

- nodul e pur;
- toate valorile atributelor sunt egale.



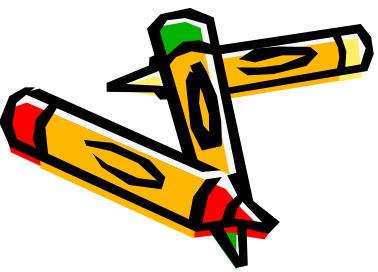
post-pruning

- *Fasonarea ulterioară* (*post-pruning*): are loc după terminarea „creşterii" arborelui, fiind un proces bazat pe măsurarea erorii de clasificare a arborelui. Un nod va fi fasonat prin renunțarea la ramurile sale, el devenind o „frunză", dacă astfel se poate diminua eroarea de clasificare.

Este necesară cuantificarea erorii de clasificare la fiecare pas al tăierii unor ramuri, pentru a se stabili dacă astfel se obține sau nu mărirea performanței clasificatorului.

avantaje

- Sunt uşor de înţeles şi interpretat, forma lor grafică reprezentând un atu puternic în acest sens;
- Necesită un volum mic de pregătire a datelor în raport cu alte tehnici;
- Permit utilizarea atât a datelor nominale cât şi a celor categoriale, fără nicio restricţie;
- Logica deciziei poate fi urmărită uşor, regulile de clasificare fiind la vedere;
- Permit utilizarea unor tehnici statistice clasice pentru validarea modelului;
- Lucrează bine cu mulţimi mari de date.



- în cazul unui număr prea mare de clase se poate deteriora rezultatul;
- algoritmul nu este incremental, în sensul că dacă apar date noi este necesară reluarea fazei de antrenament cu eşantionul complet format din vechile date şi cele noi.