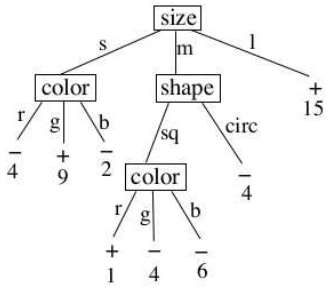
Învăţare automată

— Licenţă, anul III, 2021-2022, examenul parţial I —

Nume student: **Grupa:**

1. **(Algoritmul ID3: 2 exemple simple de aplicare (antrenare + testare))**

a. Considerăm arbore de decizie alăturat.
Numerele situate lângă etichetele +/− reprezintă cât de multe exemple de antrenament au fost asignate la nodul de decizie respectiv în cursul antrenării.



Folosind acest arbore de decizie, clasificaţi următoarele instanţe de test:

Size	Color	Shape	Smell	Classification
small	green	square	pine	
large	blue	square	pine	
medium	green	circle	rotten egg	
medium	red	square	lemon	

***Observaţie*: Pentru convenienţă, pe arborele din imaginea de mai sus, valorile atributelor au fost scrise în mod prescurtat.**

---

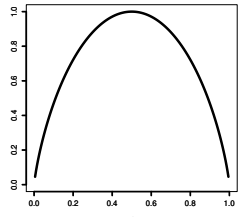
b. Acum vom presupune că aplicaţi algoritmul ID3 pe un set de date de antrenament (neprecizat) şi că folosiţi criteriul câştigului de informaţie maxim (sau, un criteriu *echivalent*(!) cu acesta). Presupunem că

- **setul de date de antrenament este constituit din 128 de exemple, dintre care 64 sunt etichetate pozitiv, iar 64 sunt etichetate negativ;**
- **instanţele de antrenament sunt caracterizate de trei atribute booleene: *Home-Owner*, *In-Debt* şi *Rich*;**
- **pentru 64 de exemple de antrenament, atributul *Home-Owner* este *true*, iar dintre aceste exemple 1/4 sunt negative şi restul sunt pozitive;**
- **pentru 96 de exemple de antrenament, atributul *In-Debt* este *true*, iar dintre aceste exemple 1/2 sunt pozitive şi restul sunt negative;**
- **pentru 32 de exemple de antrenament, atributul *Rich* este *true*, iar dintre aceste exemple 3/4 sunt pozitive şi restul sunt negative.**

***i*. Ce atribut trebuie pus în nodul rădăcină al arborelui ID3?**
(În prealabil, veţi desena compaşii de decizie corespunzători celor trei atribute de intrare care au fost indicate mai sus.)

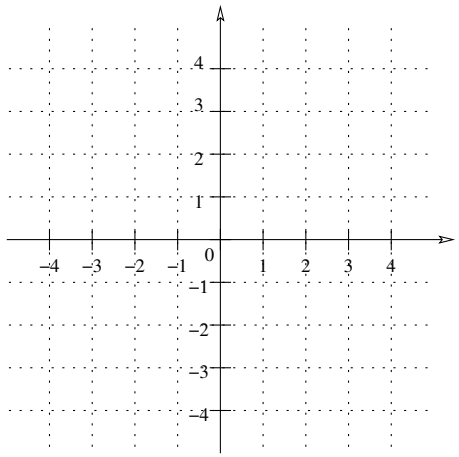
***Observaţie*: Atunci când veţi calcula entropii şi / sau câştiguri de informaţie, le veţi exprima în biţi.**

***ii*. Puteţi determina atributul care trebuie pus în nodul rădăcină al arborelui ID3 fără să faceţi calcule efectiv [pentru entropii], ci doar bazându-vă pe monotonia funcţiei entropie ($H(p)$) pentru distribuţia aleatoare Bernoulli de parametru p (vedeţi graficul alăturat)?**



2. **(Algoritmul k-NN, cu aplicare pe date din $\mathbb{R}^2$, respectiv $\mathbb{R}$: desenarea de diagrame Voronoi / identificarea zonelor de decizie; calcularea erorii la CVLOO)**

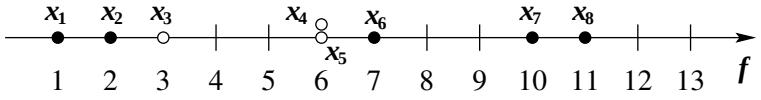
A. Se dă un set de date de antrenament din planul euclidian ($\mathbb{R}^2$), format din instanţele negative $(-1, 0)$, $(2, 1)$ şi $(2, -2)$ şi instanţele pozitive $(0, 0)$ şi $(1, 0)$.



a. Pe reperul de axe de coordonate ortogonale alăturat marcaţi aceste exemple de antrenament. Veţi folosi *convenţia* noastră de notare: simbolul • desemnează instanţe pozitive (+), iar simbolul ∘ instanţe negative (−). Desenaţi apoi graniţele de decizie determinate de algoritmul 1-NN. Veţi haşura zona / zonele de decizie corespunzătoare instanţelor pozitive.

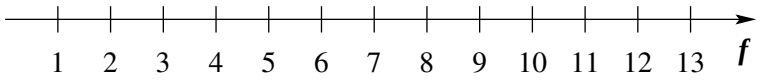
b. Cum va clasifica instanţa de test $(1, -1.01)$ de către algoritmul 1-NN? Dar de către algoritmul 3-NN? Justificaţi, în ambele cazuri.

B. În desenul de mai jos este reprezentat un set de antrenament care conţine 8 instanţe, fiecare dintre ele având doar o trăsătură (engl., feature), notată cu f.



Remarcaţi faptul că sunt două instanţe pentru care valoarea trăsăturii f este aceeaşi, şi anume 6. Aceste două instanţe sunt reprezentate prin două simboluri ∘, situate unul deasupra celuilalt, dar de fapt ele ar fi trebuit să fie reprezentate ca două simboluri ∘ suprapuse (unul peste celălalt), întrucât aceste instanţe au exact aceeaşi valoare pentru trăsătura f.

c. La acest punct al problemei veţi folosi algoritmul 1-NN. Vă cerem ca pe linia de mai jos să haşuraţi — şi, bineînţeles, să delimitaţi prin separatori decizionali — zonele în care algoritmul 1-NN va prezice semnul +, dat fiind setul de date de antrenament din figura de mai sus.

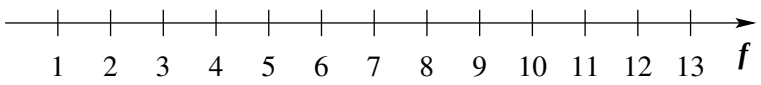


Cu ce etichetă vor fi clasificate instanţele de test $f = 2.5$ şi $f = 6.5$? Justificaţi.

***Atenţie*!**
1. În cazul în care există două sau mai multe instanţe situate exact pe „marginea" [adică, pe conturul circular al] k-NN-vecinătăţii asociate instanţei de clasificat, se va considera că toate aceste instanţe aparţin respectivei vecinătăţi, iar fiecare dintre ele dispune de un vot întreg.
2. În caz de *paritate de voturi*, veţi alege eticheta instanţei de antrenament care este situată la *distanţă minimă* către stânga faţă de instanţa de test respectivă. (Distanţă minimă este 0 atunci când instanţa de test coincide cu o instanţă de antrenament!)
3. Aceste reguli vor fi aplicate şi la punctele următoare.

d. Dacă faceţi cross-validare folosind metoda "leave-one-out" pe acest set de date, în conjuncţie cu algoritmul 1-NN, cât va fi eroarea produsă? Veţi arăta în mod clar — alcătuind un *tabel* care să conţină 1-NN-vecinătăţile respective — cum anume aţi ajuns la rezultatul pe care l-aţi indicat.

e. Similar cu punctul c, însă aici veţi folosi algoritmul 5-NN.[1]



***Indicaţie*: Justificaţi în mod riguros rezultatul, referindu-vă la diverse intervale de valori (sau valori particulare) ale lui f:**

Cazul 1: $f \in (-\infty, 4)$**:** ...

Cazul 2: $f = 4$**:** ...

...

Cu ce etichetă vor fi clasificate instanţele de test $f = 4$, $f = 6$ şi $f = 7$? Justificaţi.

[1] Fără să aplicaţi 5-NN pentru instanţele de test $f = 2.5$ şi $f = 6.5$.

3. **(Un exemplu de clasificator de tip bayesian care combină avantajele algoritmilor Bayes Naiv şi Bayes Optimal)**

Fie A, B şi C variabile aleatoare binare independente, fiecare dintre ele având posibilitatea să ia valoarea 0 cu probabilitate de 50%. Considerăm funcţia

$$Y = ((\neg B) \wedge (\neg C)) \vee (A \wedge B).$$

a. Scrieţi *tabela de adevăr* a funcţiei Y.

b. i**. Folosind tabela de adevăr a funcţiei Y — văzută acum ca un set de date de antrenament pentru un clasificator de tip bayesian — *estimaţi* valorile următoarelor probabilităţi condiţionate:**[2]

$P(A=0\|B=0,Y=0)=\ldots$	$P(A=0\|C=0,B=0,Y=0)=\ldots$ $P(A=0\|C=1,B=0,Y=0)=\ldots$
$P(A=0\|B=0,Y=1)=\ldots$	$P(A=0\|C=0,B=0,Y=1)=\ldots$ $P(A=0\|C=1,B=0,Y=1)=\ldots$
$P(A=0\|B=1,Y=0)=\ldots$	$P(A=0\|C=0,B=1,Y=0)=\ldots$ $P(A=0\|C=1,B=1,Y=0)=\ldots$
$P(A=0\|B=1,Y=1)=\ldots$	$P(A=0\|C=0,B=1,Y=1)=\ldots$ $P(A=0\|C=1,B=1,Y=1)=\ldots$

***Atenţie!* Unele dintre aceste probabilităţi condiţionate s-ar putea să nu fie definite. De *exemplu*, $P(A=0|C=0,B=0,Y=0)$ este nedfinită, fiindcă $P(C=0,B=0,Y=0)$, adică (veţi vedea) nu există în tabelul de date nicio combinaţie de forma $C=0,B=0,Y=0$. În dreptul unor astfel de probabilităţi condiţionate nedfinite veţi pune semnul $*$ *(nedefinit)*.**

ii**. Analizând rezultatele estimărilor obţinute mai sus, se poate deduce o relaţie de tip *independenţă condiţională* între variabilele A, B, C şi Y. Care este această relaţie?**

c. i**. Pe baza rezultatului de la punctul $b.ii$, scrieţi *regula de decizie* pe care o poate folosi un clasificator de tip bayesian — diferit de Bayes Naiv şi Bayes Optimal / Comun; să-i spunem *New-Bayes* — pentru învăţarea conceptului / funcţiei Y, astfel încât**
– să producă eroare 0 pe setul de date [de antrenament] de la punctul a, dar
– să necesite un număr de parametri „liberi“ (engl., free) care trebuie estimaţi mai mic decât cel al algoritmului Bayes Optimal.

ii**. *Justificaţi* în mod riguros regula de decizie pe care aţi scris-o şi că eroarea la antrenare produsă de la punctul a este într-adevăr 0.**

iii**. Care sunt aceşti parametri „liberi“?**

iv**. *Comparaţi* numărul acestor parametri „liberi“ cu numărul minimal de parametri necesari de estimat de către clasificatorii Bayes Naiv şi respectiv Bayes Optimal pe acelaşi set de date.**

[2] Estimarea va fi făcută în mod clasic, adică în sensul verosimilităţii maxime (engl., Maximum Likelihood Estimation, MLE). Nu veţi aplica nicio regulă de netezire a acestor probabilităţi, cum ar fi regula “add-one” a lui Laplace.