Învăţare automată

— Licenţă, anul III, 2018-2019, examenul parţial I —

Nume student: **Grupa:**

1. **(Formula lui Bayes; inferenţe statistice; exemplificarea noţiunii de ipoteză / ipoteze MAP ("Maximum A posteriori Probability"))**

A. Mickey dă cu zarul de mai multe ori, sperând să obţină un 6. Secvenţa celor 10 rezultate obţinute de el în urma acestor aruncări este următoarea: $1, 2, 4, 3, 2, 2, 3, 5, 1, 6$**. Mickey se întreabă dacă nu cumva zarul este măsluit (având tendinţa să producă de mai multe ori faţa 2 decât ar fi normal dacă zarul ar fi perfect).**

Se presupune că în general fiecare set de 100 de zaruri conţine 5 zaruri măsluite (engl., unfair) în aşa fel încât este favorizată apariţia feţei 2, rezultând următoarea distribuţie de probabilitate a celor şase feţe, $(1, 2, 3, 4, 5, 6)$**:** $P = [0.1,\ 0.5,\ 0.1,\ 0.1,\ 0.1,\ 0.1]$**.**

Folosind teorema lui Bayes, furnizaţi-i lui Mickey informaţia care-l interesează: în ce măsură putem spune că zarul este măsluit [sau nu]?

***Observaţie*: Puteţi folosi următoarele aproximări:** $\log_2 3 = 1.585$**;** $\log_2 5 = 2.322$**;** $\log_2 19 = 4.248$**.**

(Algoritmul Bayes Naiv şi algoritmul Bayes Corelat: aplicare)

B. Se dă setul de date alăturat, cu A **şi** B **variabile de intrare, iar** C **variabilă de ieşire.**

A	B	C	nr. apariţii
0	0	1	3
0	1	0	1
0	1	1	4
1	0	0	5
1	1	0	2
1	1	1	1

a. Care este numărul minim de probabilităţi ce trebuie estimate pentru a putea construi după aceea (pe acest set de date) un clasificator de tip Bayes Naiv? Justificaţi.

b. Similar, pentru clasificatorul Bayes Corelat. Justificaţi.

c. Care este *decizia* clasificatorului Bayes Naiv pentru $A = 0, B = 1$**? Precizaţi cu ce *probabilitate* este luată această decizie.**

d. Care este *decizia* clasificatorului Bayes Corelat pentru $A = 0, B = 1$**? Precizaţi cu ce *probabilitate* este luată această decizie.**

e. Dacă rezultatele obţinute la punctele c **şi** d **diferă (fie şi numai în privinţa *probabilităţilor* cu care sunt luate deciziile), care este explicaţia? Justificaţi în mod riguros.**

2. **(Algoritmul ID3: aplicare; calculul erorii la antrenare, respectiv la validare)**

A. Daţi *definiţia* noţiunii de *arbore de decizie*.

B. Folosiţi următorul set de date pentru a învăţa cu ajutorul unui arbore de decizie dacă o ciupercă este sau nu comestibilă, utilizând atributele discrete *Formă*, *Culoare* şi *Miros*.

Formă	*Culoare*	*Miros*	*Comestibilă*
C	B	1	1
D	B	1	1
D	W	1	1
D	W	2	1
C	B	2	1
D	B	2	0
D	G	2	0
C	U	2	0
C	B	3	0
C	W	3	0
D	W	3	0

***Observaţie* (1): Dacă la calculul entropiilor şi / sau al câştigului de informaţie folosiţi *alte formule decât* cele date în *definiţia* acestor noţiuni, *enunţaţi-le* la cazul general şi apoi *demonstraţi-le* în mod succint.**

***Observaţie* (2): Dacă veţi folosi corect *definiţiile*, veţi constata că veţi avea de făcut foarte puţine calcule!**

a. Calculaţi entropia condiţională specifică $H(Comestibilă \mid Miros = 1$ sau $Miros = 3)$.

***Sugestie*: Aplicaţi (deci, scrieţi mai întî) definiţia noţiunii de entropie condiţională specifică.**

b. Ce atribut va alege algoritmul ID3 ca rădăcină a arborelui de decizie?

c. Elaboraţi întregul arbore de decizie care va fi învăţat din datele de mai sus (fără pruning).

d. Exprimaţi cu ajutorul unui set de reguli din calculul propoziţional clasificarea produsă de arborele de decizie obţinut. (IF ... THEN *Comestibilă*; IF ... THEN $\neg$*Comestibilă*.)

e. Să presupunem că avem un set de date de validare:

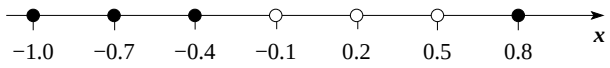
Formă	*Culoare*	*Miros*	*Comestibilă*
C	B	2	0
D	B	2	0
C	W	2	1

Care va fi eroarea produsă de arborele de decizie pe mulţimea de date de antrenare respectiv pe datele de validare? (Exprimaţi răspunsul ca număr de exemple clasificate greşit.)

3. **(Algoritmul AdaBoost: întrebări în legătură cu aplicarea algoritmului pe un set de date din $\mathbb{R}$)**

A. Demonstraţi că minimul funcţiei $Z(\alpha_t) \stackrel{def.}{=} (1-\varepsilon_t)\exp(-\alpha_t) + \varepsilon_t \exp(\alpha_t)$ se atinge pentru valoarea argumentului $\bar{\alpha}_t = \ln\sqrt{\frac{1-\varepsilon_t}{\varepsilon_t}}$. Calculaţi $Z(\bar{\alpha}_t)$.

B. Considerăm setul de date de antrenament din figura următoare. Vom folosi algoritmul AdaBoost cu compaşi de decizie în rolul de ipoteze ,,slabe".



a. Determinaţi separatorul decizional corespunzător primei ipoteze ,,slabe", h_1. Desenaţi-l pe figura de mai sus şi indicaţi [eventual printr-o mică săgeată perpendiculară pe acest separator] care este zona clasificată cu $+$.

b. Calculaţi ε_1 şi α_1. Cât este acurateţea obţinută de AdaBoost la antrenare dacă oprim acum algoritmul?

c. Cât va fi valoarea noilor probabilităţi / ponderi $D_2(i)$ pentru fiecare dintre cele şapte exemple de antrenament? (*Atenţie*! Nu uitaţi să faceţi normalizarea cu ajutorul factorului Z_1.)

d. Desenaţi separatorul decizional corespunzător celei de-a doua ipoteze ,,slabe", h_2. Indicaţi iarăşi zona de decizie corespunzătoare clasei $+$.

e. Care sunt exemplele de antrenament cărora le va fi asignată cea mai mică pondere / probabilitate după ce algoritmul AdaBoost va fi terminat cea de-a doua sa iteraţie?

f. Se îmbunătăţeşte oare acurateţea la antrenare obţinută de AdaBoost la a doua iteraţie în raport cu cea obţinută la prima iteraţie?

g. Va putea oare AdaBoost să obţină eroare la antrenare 0 pe acest set de date? Justificaţi riguros, citând un *rezultat teoretic* prezentat la curs.

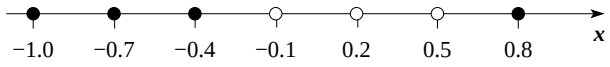
***Observaţie importantă*: Dacă veţi folosi *rezultate teoretice importante* prezentate la curs, *calculele* pe care va trebui să le faceţi la acest exerciţiu vor fi *foarte simple*!**

Răspuns:

[a.-c.] Iteraţia $t = 1$:

i	1	2	3	4	5	6	7
$D_1(i)$							

s	-1.15	-0.25	$+0.65$
$err_{D_1}(X < s)$			
$err_{D_1}(X \geq s)$			



$\varepsilon_1 =$

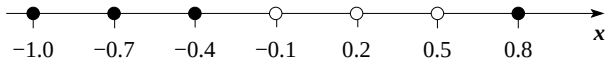
$\alpha_1 =$

i	1	2	3	4	5	6	7
$D_2(i)$							

$Z_1 =$

[d.-f.] Iteraţia $t = 2$:

s	-1.15	-0.25	$+0.65$
$err_{D_2}(X < s)$			
$err_{D_2}(X \geq s)$			



$\varepsilon_2 =$

$\alpha_2 =$

i	1	2	3	4	5	6	7
$D_3(i)$							

$Z_2 =$

t	α_t	x_1	x_2	x_3	x_4	x_5	x_6	x_7
1	$\alpha_1 =$							
2	$\alpha_2 =$							
	H							

***Observaţie*: În acest tabel, veţi trece clasificările făcute de ipotezele „slabe“ h_t (pentru $t = 1$ şi $t = 2$) şi respectiv ipoteza „combinată“ H livrată de AdaBoost după efectuarea primelor două iteraţii. Veţi *încercui* (în tabelul acesta) *clasificările eronate*, sub forma $\oplus$ sau $\ominus$.**

4. **(BONUS!**
Cât de multe date de antrenament necesită algoritmul Bayes Naiv?
[LC: complexitatea la eşantionare])

Fie variabilele aleatoare $X_1, X_2, \ldots, X_d \overset{not.}{=} Y$**, fiecare dintre ele urmând o distribuţie Bernoulli, de parametru** $1/2$**.**

***Comentariu*:** Variabilele $X_1, X_2, \ldots, X_d$ fiind de tip $Bernoulli(1/2)$, rezultă că a „observa" (sau, a genera) o valoare (x_i) pentru o variabilă oarecare, X_i, fixată este echivalent cu aruncarea unei monede perfecte. Similar, a „observa" (sau, a genera) o instanţă etichetată $\bar{x} \overset{not.}{=} (X_1 = x_1, \ldots, X_d = x_d)$ este echivalent cu aruncarea a d monede perfecte în mod independent (sau, echivalent, cu d aruncări independente ale unei monede perfecte).

Scopul acestui exerciţiu este să calculăm N_{NB}**, o margine inferioară pentru numărul de exemple de antrenament de care este nevoie [în prezentul context] astfel încât, dat fiind un prag** $\varepsilon \in (0, 1)$**, fixat, să putem spune că în setul de date reprezentat de cele** N_{NB} **exemple de antrenament probabilitatea să nu fi întâlnit toate valorile [tuturor] variabilelor** $X_1, X_2, \ldots, X_d$ **este mai mică decât** ε**. (Aşadar, în acest context, cu probabilitate de** $1 - \varepsilon$ **putem estima toţi parametrii algoritmului Bayes Naiv.)**

***Sugestie de lucru*:**

***i*. Considerând (din nou)** X_i **fixat, care este *probabilitatea* ca în** N **exemple de antrenament (unde** $N \in \mathbb{N}$**) să nu fi întâlnit ambele valori ale lui** X_i**?**[1]

***ii*. Folosind inegalitatea** $P(E_1 \cup \ldots \cup E_n) \leq \sum_{i=1}^{n} P(E_i)$ **care este valabilă pentru orice evenimente aleatoare** $E_1, \ldots, E_n$**, şi considerând din nou** N **fixat, calculaţi o *margine superioară* pentru *probabilitatea* ca în** N **exemple de antrenament să *nu* fi întâlnit *toate* valorile [*tuturor*] celor** d **variabile** $X_1, X_2, \ldots, X_d$**.**

***Observaţie importantă*: Pentru fixarea ideilor, la punctele *i*. şi *ii*. puteţi considera** $d = 5$ **şi** $N = 10$**. Veţi da răspunsurile atât în această variantă (adică, folosind aceste date concrete) cât şi în cazul general.**

***iii*. Dacă impuneţi condiţia ca marginea superioară a probabilităţii] de la punctul *ii*. să fie egală cu un** $\varepsilon > 0$ **fixat, puteţi obţine — raţionând analitic, pas cu pas, prin relaţii de echivalenţă — cât trebuie să fie** N_{NB}**, în funcţie de** d **şi** ε **(deci vom scrie** $N_{NB}(d, \varepsilon)$**) astfel încât, dat fiind un set de** N_{NB} **exemple de antrenament (generate cu ajutorul variabilelor aleatoare** $X_1, X_2, \ldots, X_d$**), *probabilitatea* de a *nu* întâlni în acest set *toate* valorile [*tuturor*] variabilelor** $X_1, X_2, \ldots, X_d$ **să fie mai mică decât** ε**.**

***iv*. Calculaţi efectiv marginile superioare** $N_{NB}(d, \varepsilon)$ **pentru** $\varepsilon = 0.01$ **şi** $d \in \{2, 5, 10\}$**.**
Puteţi folosi aproximarea $\log_2 5 = 2.3219$**.**

[1] Echivalent spus, care este *probabilitatea* ca pentru X_i, în *toate* cele N exemple de antrenament, să apară în mod exclusiv(!) *fie* una *fie* cealaltă dintre valorile lui X_i?