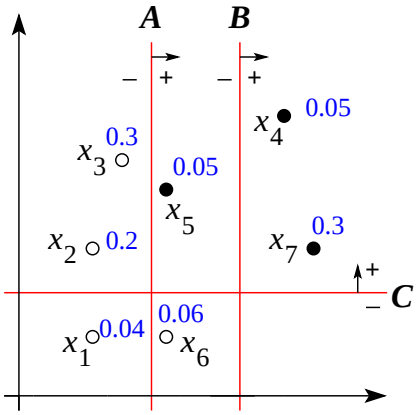
Învăţare automată

— Licenţă, anul III, 2020-2021, examenul parţial II —

Nume student: **Grupa:**

1. **(Algoritmul AdaBoost: exemplu de aplicare pe date din $\mathbb{R}^2$; întrebări calitative)**

Fie setul de date de antrenament din figura alăturată. Am marcat cu semnul $\circ$ exemplele / instanţele negative ($y_i = -1$) şi cu semnul $\bullet$ instanţele pozitive ($y_i = +1$). Figura conţine de asemenea ponderile normalizate (adică, probabilităţile) asociate exemplelor de antrenament, aşa cum au rezultat în urma executării unui anumit număr de iteraţii ale algoritmului AdaBoost. În figură sunt trasaţi şi trei compaşi de decizie, $h(x;\theta_A)$, $h(x;\theta_B)$, and $h(x;\theta_C)$ sau, pe scurt, A, B şi C.



a. Care dintre aceşti trei compaşi de decizie consideraţi că a fost folosit la *precedenta iteraţie* a algoritmului AdaBoost, în aşa fel încât să rezulte ponderile [asociate exemplelor] prezentate în figură? Veţi răspunde indicând A, B sau C şi veţi justifica în mod riguros alegerea pe care aţi făcut-o.

b. Pe care dintre aceşti trei compaşi de decizie consideraţi că-l va selecta algoritmul AdaBoost la *iteraţia următoare*? Veţi răspunde indicând A, B sau C şi veţi justifica riguros, prin calcule, alegerea pe care aţi făcut-o.

c. În figura dată, încercuiţi instanţele de antrenament (este posibil să nu fie niciuna!) pe care ansamblul (i.e., combinaţia liniară de compaşi de decizie)

$$H_2(x) = sign(\alpha_A h(x;\theta_A) + \alpha_C h(x;\theta_C)) \text{ cu } \alpha_A = 0.3 \text{ şi } \alpha_C = 0.5,$$

nu le poate clasifica în mod corect.

2. **(Concepte din $\mathbb{R}$ *reprezentabile* cu ajutorul combinaţiilor liniare de compaşi de decizie)**

Presupunem că avem o problemă de clasificare a unor instanţe pe axa reală: fiecare instanţă x_i este un număr real, iar etichetele pe care urmează să le preziceţi sunt binare, $y_i \in \{-1, +1\}$.

Pentru această problemă de clasificare, veţi folosi *ansambluri*, adică nişte combinaţii liniare de separatori / ipoteze „slabe". (Atenţie! NU trebuie să folosiţi algoritmul AdaBoost!) Vă readucem aminte că un astfel de *clasificator* are forma următoare:

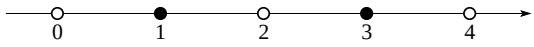
$$\hat{y} = sign\Big(\sum_{t=1}^{T} \alpha_t h_t(x)\Big), \tag{1}$$

unde $\hat{y}$ este eticheta prezisă, $sign(x)$ este $+1$ dacă $x > 0$ şi respectiv -1 în cazul contrar, α_t este o *pondere* (număr real strict pozitiv), iar $h_t(x)$ este predicţia făcută de către ipoteza „slabă" h_t. Fiecare h_t ia una dintre următoarele forme:

$$h_t(x; s, +) = \begin{cases} -1 & \text{dacă } x \le s \\ +1 & \text{dacă } x > s \end{cases} \qquad h_t(x; s, -) = \begin{cases} +1 & \text{dacă } x \le s \\ -1 & \text{dacă } x > s, \end{cases}$$

pentru un anumit *prag de separare* (engl., split threshold) $s \in \mathbb{R}$.[1]

Considerăm setul de date din figura alăturată, format din 5 instanţe situate pe axa reală.



a. Arătaţi că acest set de date (LC: sau, acest *concept*) este *reprezentabil* cu ajutorul unei combinaţii liniare formată din 4 ipoteze „slabe". Aşadar, vă cerem să identificaţi în mod explicit 4 ipoteze „slabe" $h_1, \ldots, h_4$, precum şi ponderile lor $\alpha_1, \ldots, \alpha_4$, astfel încât ipoteza combinată $sign\Big(\sum_{t=1}^{T} \alpha_t h_t(x)\Big)$ să fie *consistentă* cu datele de mai sus.

Puneţi în evidenţă cele 4 ipoteze „slabe" $h_1, \ldots, h_4$ pe desenul de mai sus, folosind praguri de separare $s_1, \ldots, s_4$ precum şi linii verticale, cărora le veţi asocia o etichetare adecvată. Concret, veţi desemna cu semnele + şi − la stânga şi la dreapta fiecărei linii verticale — asociată unui anumit prag s_j — zonele de decizie determinate de această ipoteză „slabă".

***Observaţie*: Ţinând cont de modul în care a fost definită funcţia $sign$ mai sus, *regula de predicţie* (1) va trata cazurile de „paritate" etichetând cu -1 instanţele pentru care suma ponderată (engl., weighted sum) a predicţiilor făcute de ipotezele „slabe" este 0. Acest mod de tratare a cazurilor de „paritate" este [foarte] util pentru rezolvarea punctului a.**

***Indicaţie*: Pentru a justifica uşor consistenţa *ansamblului* ales de dumneavoastră cu datele din figura de mai sus, vă cerem să completaţi un tabel similar cu cel pe care-l folosim la algoritmul AdaBoost atunci când facem calculul erorii la antrenare produse de ipoteza combinată $sign\Big(\sum_{t=1}^{T} \alpha_t h_t(x)\Big)$.**

α_t	h_t	0	1	2	3	4
$\alpha_1 =$	$h_1(x_i)$					
$\alpha_2 =$	$h_2(x_i)$					
$\alpha_3 =$	$h_3(x_i)$					
$\alpha_4 =$	$h_4(x_i)$					
$\sum_{t=1}^{4} \alpha_t h_t(x_i)$						

[1] Remarcaţi faptul că definiţia dată aici pentru noţiunea de *ipoteză „slabă"* corespunde bine-cunoscutului *compas de decizie* (engl., decision stump), pentru cazul particular când datele sunt din $\mathbb{R}$.

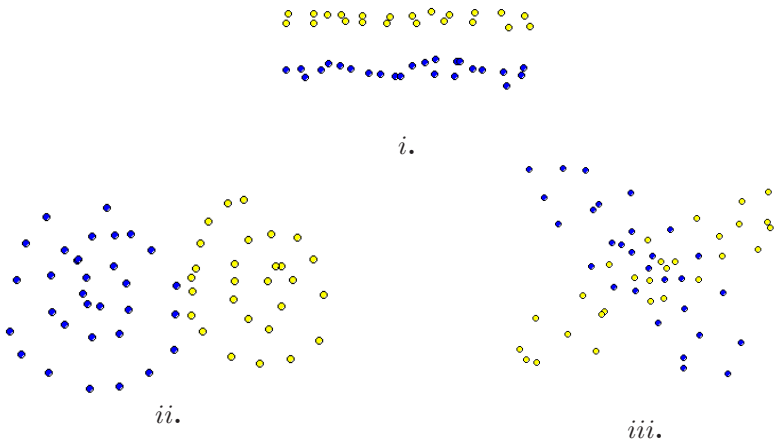
b. Demonstraţi că setul de date de mai sus NU este reprezentabil cu mai puţin de 4 ipoteze „slabe“.

c. *Generalizaţi* rezultatul obţinut la punctul a, referindu-vă la posibilitatea (sau, dimpotrivă, imposibilitatea) de a *reprezenta* — folosind combinaţii liniare de ipoteze „slabe“ aşa cum au fost definite mai sus — dataset-uri arbitrare [formate din instanţe situate] pe axa reală.

3. **(Care dintre cei trei algoritmi de clusterizare — clusterizare ierarhică, K-means şi EM/GMM — este adecvat pentru a clusteriza un anumit set de date?)**

A. ***Comentariu*:**[2]

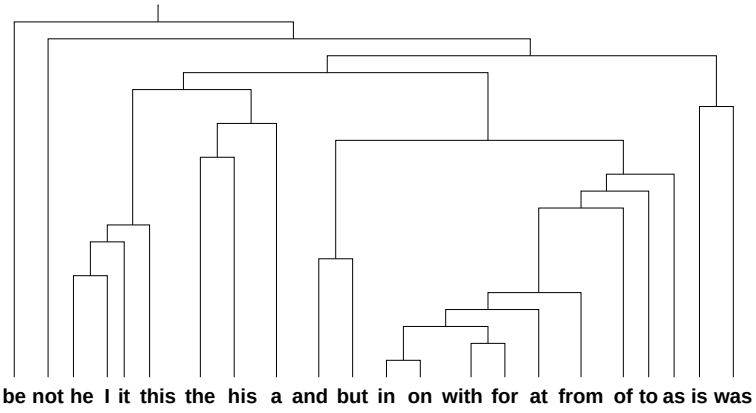
La seminar am analizat problema 61 de la capitolul *Clusterizare*, în care sunt prezentate mai multe seturi de date (vedeţi figura de mai jos) şi se cere să se precizeze care dintre cele trei tipuri de metode de clusterizare predate la curs — şi anume, clusterizare ierarhică (cu similaritate *single-*, *complete-*, şi respectiv *average-linkage*), algoritmul K-means şi algoritmul EM pentru modele de tip mixtură de distribuţii gaussiene[3] (GMM) — este [mai] adecvat pentru a produce clusterele care au fost indicate în această figură, folosind cele două culori, galben şi albastru.



i.

ii. *iii.*

B. La curs am arătat că putem face clusterizarea celor mai frecvente cuvinte care apar într-un corpus lingvistic,[4] folosind clusterizarea ierarhică. În acest scop, am arătat că se poate utiliza ca *măsură de similaritate* dintre două cuvinte oarecare numărul de „contexte“ (identice) în care apar aceste două cuvinte.[5] Iată exemplul de *dendrogramă* care a fost prezentat la curs:



[2]Nu veţi avea nimic de rezolvat la secţiunea aceasta, dar este indicat să o lecturaţi, în particular pentru a putea să răspundeţi la punctul *b* de la secţiunea B.

[3]Aici, ne referim în mod implicit la cazul distribuţiilor gaussiene bivariate / bidimensionale. (Însă acest *specific* nu afectează linia raţionamentului din exerciţiul prezent.)

[4]Un *corpus lingvistic* este o colecţie de documente. Exemplul dat la curs se referă la corpusul [Universităţii] Brown, foarte cunoscut şi studiat în domeniul lingvisticii computaţionale.

[5]Pentru un cuvânt oarecare dintr-un text dat, putem considera drept „context“ o „fereastră“ constituită dintr-un anumit număr de cuvinte la stânga şi la dreapta cuvântului respectiv.

a. *Întrebare*: Putem oare să aplicăm algoritmul K-means cu distanţa euclidiană şi / sau algoritmul EM/GMM pe acelaşi set de date ca mai sus (i.e., corpusul Brown), în aşa fel încât să obţinem rezultate bune?

***Precizări*:**

1. Întrucât algoritmii K-means şi EM/GMM lucrează pe date numerice, pentru a putea răspunde la *întrebarea* de mai sus ne propunem să asociem fiecărui cuvânt câte o valoare numerică, şi anume indicele poziţiei lui într-un dicţionar / lexicon. (Practic, acesta este o listă formată din toate cuvintele din corpus.)
2. Pentru aceşti doi algoritmi, K — numărul de clustere în cazul lui K-means şi, respectiv, numărul de componente ale mixturii în cazul lui EM/GMM — poate fi ales, de exemplu, ca fiind numărul „natural“ de clustere stabilit în urma clusterizării ierarhice, aşa cum am arătat de pildă la problema 1.b de la capitolul de *Clusterizare*.

Explicaţi succint, pentru fiecare dintre cei doi algoritmi de clusterizare menţionaţi, motivele care au dus la acceptarea sau, dimpotrivă, la respingerea lui de către dumneavoastră ca răspuns la *întrebarea* pusă mai sus.

- **K-means (cu distanţa euclidiană):**

- **EM/GMM:**

b. Cele două task-uri de clustering (cel de la secţiunea A şi respectiv cel de la secţiunea B) din această problemă pun în evidenţă o anumită *caracteristică* a clusterizării ierarhice — în comparaţie cu clustering-ul partiţional şi cu clustering-ul bazat pe modelare probabilistă —, alta decât (1) capabilitatea de a produce ierarhii de clustere şi (2) faptul că nu necesită fixarea în avans a numărului de clustere care trebuie formate.
Care credeţi că este această *caracteristică*?

4. **(Algoritmul K-means: aplicare pe date din $\mathbb{R}^2$ monotonia criteriului B)**

Considerăm setul de date din tabelul alăturat. Fiecare linie din tabel reprezintă un punct din $\mathbb{R}^2$; pentru convenienţă, identificăm fiecare punct / instanţă cu câte o literă.

A	1	1
B	3	3
C	6	6
D	6	12
E	9	9
F	11	11
G	0	3
H	3	0
I	9	3

a. Executaţi manual două iteraţii complete ale algoritmului K-means pe acest set de date, folosind $K = 3$ şi distanţa euclidiană. Clusterele vor fi iniţializate după cum urmează:

$$C_1^{(0)} = \{A, B, F\},\ C_2^{(0)} = \{C, H, I\},\ C_3^{(0)} = \{D, E, G\}.$$

Veţi face reprezentarea datelor pe gridurile de mai jos, şi anume câte unul pentru fiecare iteraţie.

***Indicaţie*: În acele cazuri în care *metoda geometrică* [bazată pe reprezentarea mediatoarelor determinate de perechi de centroizi] nu este întru totul concludentă în privinţa asignării unei instanţe oarecare la un anumit cluster / centroid, va trebui să faceţi verificarea riguroasă, folosind *metoda analitică* [bazată pe calculul distanţelor].**

b. La problema 42 am arătat că are loc următoarea *proprietate*:
Dacă definim „criteriul variaţiei inter-clustere“ prin expresia

$$B(\mu^{(t)}) \stackrel{def.}{=} \sum_{j=1}^{K} \frac{n_j^{(t)}}{n} ||\mu_j^{(t)} - \bar{x}||^2,$$

unde

- **n este numărul total de instanţe de clusterizat,**
- **$n_j^{(t)}$ este numărul de instanţe din clusterul C_j la iteraţia t în momentul (re)calculării mediilor,**
- **$\mu_j^{(t)}$ este media instanţelor din clusterul C_j, calculată la iteraţia t (înainte de reasignarea instanţelor la noii centroizi), iar**
- **$\bar{x}$ este media tuturor instanţelor de clusterizat (adică, centrul de greutate al intregii mulţimi de instanţe),**

atunci acest „criteriu“ $B(\mu^{(t)})$ este crescător (dar nu neapărat strict crescător) de la o iteraţie la alta.[6]

Folosind rezultatele obţinute la punctul a, demonstraţi (numeric!) că

$$B(\mu^{(1)}) \leq B(\mu^{(2)}).$$

***Indicaţie*: Pentru a rezolva cu mai multă uşurinţa această cerinţă, vă sugerăm să**
– arătaţi că pe aceste date avem $\sum_{j=1}^{3} \mu_j^{(1)} = \sum_{j=1}^{3} \mu_j^{(2)}$, iar apoi
– dovediţi că această egalitate implică (pe aceste date!) echivalenţa

$$B(\mu^{(1)}) \leq B(\mu^{(2)}) \Leftrightarrow \sum_{j=1}^{3} ||\mu_j^{(1)}||^2 \leq \sum_{j=1}^{3} ||\mu_j^{(2)}||^2.$$

Ultima inegalitate este uşor de verificat numeric (şi chiar trebuie să o verificaţi!).

[6] De fapt, ceea ce am enunţat aici este o versiune simplificată a proprietăţii care a fost enunţată (şi demonstrată) la problema 42.

Răspuns:

a.

Iniţializare:

$C_1^{(0)} = \{A, B, F\}$
$C_2^{(0)} = \{C, H, I\}$
$C_3^{(0)} = \{D, E, G\}$.

Iteraţia 1:

$\mu_1^{(1)} =$

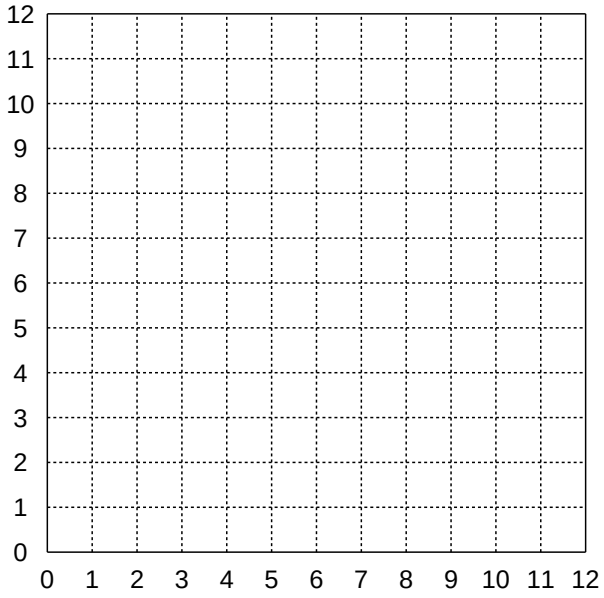
$\mu_2^{(1)} =$

$\mu_3^{(1)} =$

$C_1^{(1)} =$

$C_2^{(1)} =$

$C_3^{(1)} =$



Iteraţia 2:

$\mu_1^{(2)} =$

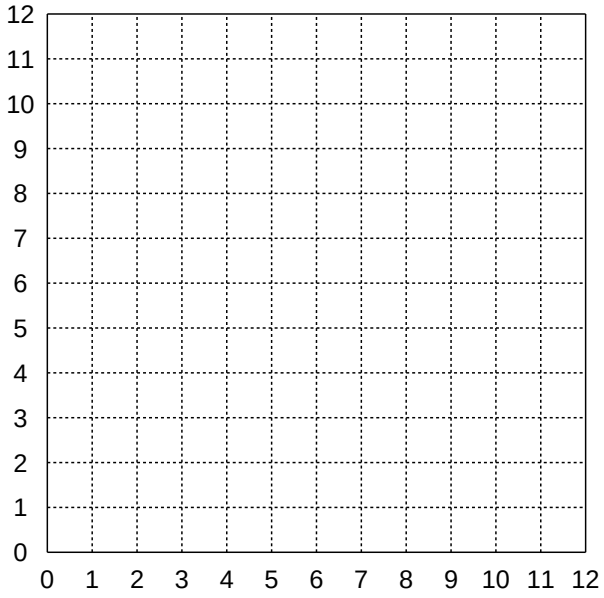
$\mu_2^{(2)} =$

$\mu_3^{(2)} =$

$C_1^{(2)} =$

$C_2^{(2)} =$

$C_3^{(2)} =$



b. ...

5. **(EM/GMM, cazul unidimensional: executarea manuală a pasului M (precum şi a următorului pas E), pentru o mixtură de un tip particular, parametrii liberi fiind π_1, μ_1, μ_2)**

Să zicem că dorim să antrenăm (engl., to fit) un model de tip mixtură de distribuţii gaussiene unidimensionale, folosind $K = 2$ componente. Avem $n = 5$ instanţe, ale căror valori (notate cu x_i, cu $i = 1, \ldots, 5$) sunt:

$$5,\ 15,\ 25,\ 30,\ 40.$$

Folosim algoritmul EM pentru a găsi estimările de verosimilitate maximă (engl., maximum likelihood estimates) pentru parametrii modelului, şi anume: probabilităţile de selecţie (sau, mixare) ale celor două componente (vom nota aceste probabilităţi cu π_1 şi π_2) şi mediile acestor componente (notate cu μ_1 şi respectiv μ_2). Deviaţiile standard ale celor două componente sunt considerate fixate şi au valoarea 10.

Presupunem că în timpul execuţiei algoritmului EM, la o anumită iteraţie t, la pasul E au fost determinate mediile variabilelor neobservabile Z_{ij} — aceste medii se mai numesc *responsabilităţi*; engl., responsibilities — corespunzătoare celor două componente, pentru fiecare dintre cele cinci instanţe, după cum urmează:

$p_{i1}^{(t)}$	0.2	0.2	0.8	0.9	0.9
$p_{i2}^{(t)}$	0.8	0.8	0.2	0.1	0.1

a. Deduceţi expresia funcţiei „auxiliare“ $Q(\pi_1, \mu_1, \mu_2 | \pi_1^{(t-1)}, \mu_1^{(t-1)}, \mu_2^{(t-1)})$ care urmează să fie optimizată la execuţia pasului M al respectivei iteraţii (t) a algoritmului EM.

b. Ce valori vor fi atribuite parametrilor π_1, π_2, μ_1 şi μ_2 la execuţia respectivului pas M?

Veţi optimiza funcţia „auxiliară“ Q pe care aţi determinat-o la punctul a. [În cazul în care nu aţi rezolvat punctul a sau nu ştiţi cum se optimizează respectiva funcţie Q — dar doar în acest caz! —, puteţi folosiţi *formule* de genul celor pe care le-am predat la curs. Rezultatele trebuie să fie aceleaşi, în ambele variante!]

c. *i.* **Deduceţi expresiile necesare pentru a calcula $p_{i1}^{(t+1)}$ şi $p_{i2}^{(t+1)}$, mediile variabilelor neobservabile Z_{ij} corespunzătoare următorului pas E al algoritmului EM.**
ii. **Calculaţi aceste probabilităţi pentru $i = 1$.**

***Indicaţie*: Puteţi folosi următoarele aproximări ale funcţiei de densitate de probabilitate (p.d.f.) pentru distribuţia gaussiană standard: $\mathcal{N}(0.9 \,|\, 0,\, 1) = 0.2660$ şi $\mathcal{N}(2.4 \,|\, 0,\, 1) = 0.0223$.**