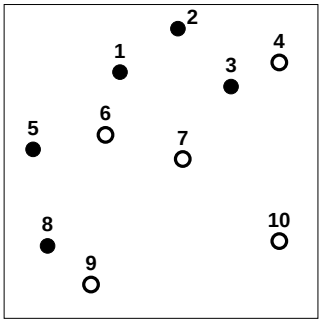
Învăţare automată

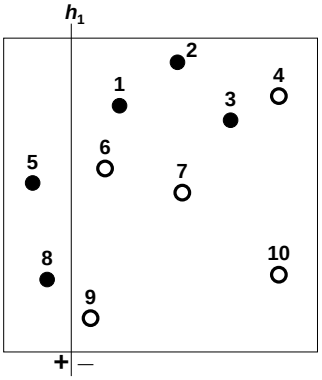
— Licenţă, anul III, 2021-2022, examenul parţial II —

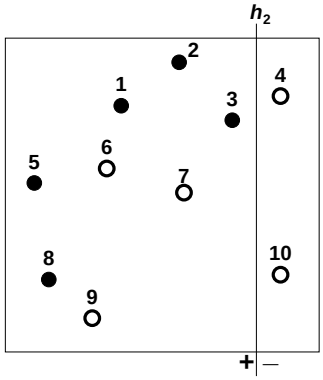
Nume student: **Grupa:**

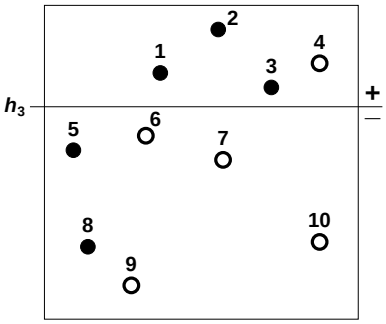
1. [3.5p] **(Algoritmul AdaBoost: aplicare pe un set de date din $\mathbb{R}^2$)**

Se consideră că aplicăm algoritmul AdaBoost pe dataset-ul din figura alăturată. (Pentru uşurinţa exprimării la calcule, am notat pe figură indicii instanţelor de antrenament, în imediata apropiere a acestora.)
Folosim convenţia noastră obişnuită de notare: simbolul $\bullet$ desemnează instanţe pozitive, iar simbolul $\circ$ instanţe negative.
La primele trei iteraţii ale algoritmului au fost selectaţi compaşii de decizie h_1, h_2 şi h_3 (în această ordine), aşa cum se indică în figurile de mai jos.









***Obiectivul* acestei probleme este să determinăm dacă la sfârşitul celor trei iteraţii algoritmul AdaBoost reuşeşte să clasifice perfect toate instanţele de antrenament.**

a. Calculaţi

– distribuţiile probabiliste corespunzătoare celor 3 iteraţii, adică $D_1(x_i)$ $D_2(x_i)$ şi $D_3(x_i)$, pentru $i = 1, \ldots, 10$;
– pentru fiecare dintre cele 3 iteraţii ($t = 1, 2, 3$): eroarea ponderată la antrenare (ε_t) produsă de compasul de decizie h_t, precum şi ponderea (α_t) asociată ipotezei / compasului de decizie h_t.

Veţi completa tabelele următoare şi veţi indica succint(!) modul în care aţi procedat pentru a ajunge la rezultatele respective.

i	1	2	3	4	5	6	7	8	9	10
$D_1(x_i)$										
$D_2(x_i)$	1/6			1/14						
$D_3(x_i)$	7/66			1/22		1/6				

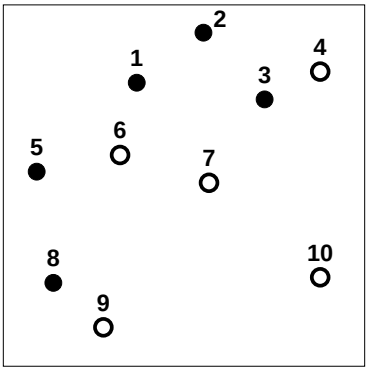
t	1	2	3
ε_t			
α_t			

***Atenţie*! Pentru a vă uşura munca, am completat noi câteva dintre elementele primului tabel. Vă puteţi baza pe valorile indicate de noi, ca să vă simplificaţi raţionamentele / calculele!**

b. Folosind ipoteza combinată obţinută de algoritmul AdaBoost la finalul celei de-a treia iteraţii, stabiliţi

i. **eroarea la antrenare produsă (pentru calcularea ei, puteţi folosi tabelul de mai jos),**

ii. **zonele de decizie corespunzătoare acestui clasificator (veţi justifica modul în care aţi procedat!). Veţi indica aceste zone de decizie, precum şi graniţele de decizie, pe desenul de alăturat.**



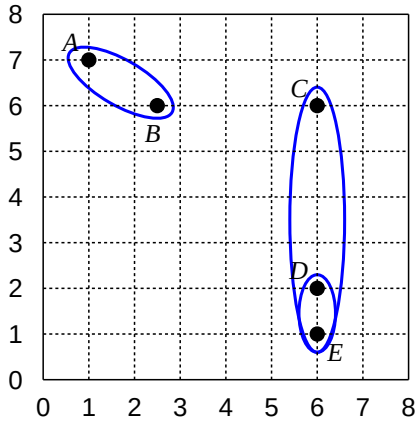
t	α_t	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
1	$\alpha_1 = \ldots$										
2	$\alpha_2 = \ldots$										
3	$\alpha_3 = \ldots$										
	$H_3(x_i)$										

***Atenţie*! Pentru a vă uşura munca, vă furnizăm noi următoarele valori numerice:** $\ln\sqrt{7/3} \approx 0.423$, $\ln\sqrt{11/3} \approx 0.649$, $\ln\sqrt{15/7} \approx 0.923$.

2. [1.75p] **(Clusterizare ierarhică aglomerativă: un exemplu simplu de aplicare, cu single-, complete- şi average-linkage, pe date din $\mathbb{R}^2$)**

Considerăm punctele $A(1,7)$, $B(2.5,6)$, $C(6,6)$, $D(6,2)$ şi $E(6,1)$ din planul euclidian. Pe acest dataset veţi aplica algoritmul de clusterizare ierarhică aglomerativă (i.e., bottom-up), folosind pe rând (separat) funcţiile de similaritate single-linkage, complete-linkage şi average-linkage.

Care dintre aceste funcţii de similaritate va conduce după executarea a trei iteraţii consecutive la ierarhia aplatizată prezentată în figura alăturată?



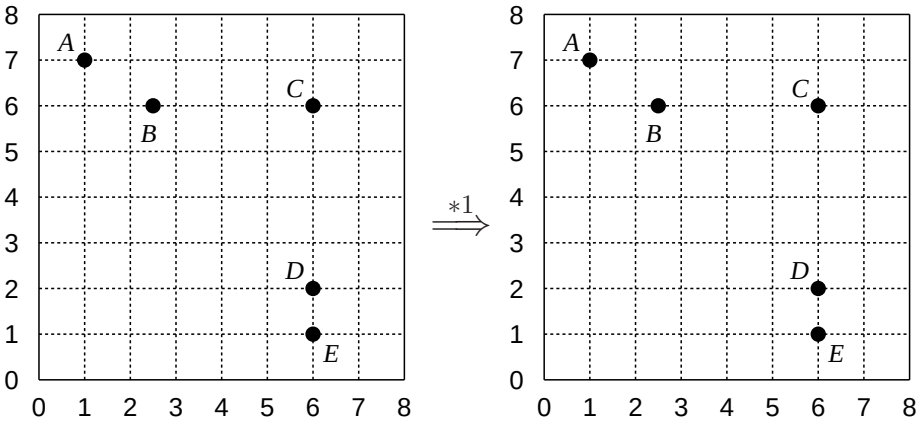
***Indicaţie*:**
Veţi folosi grid-urile de mai jos.
Pentru fiecare dintre cele 3 tipuri de măsuri de similaritate, *veţi justifica riguros* trecerea de la iteraţia 2 la iteraţia 3 determinând (*numeric*!) minimul *distanţelor dintre perechile de clustere* care au fost formate la iteraţia 2.
De *exemplu*, dacă în figura de mai sus notăm $C_1 = \{D, E\}$, $C_2 = \{A, B\}$, atunci *justificarea* acestui rezultat ar putea fi scrisă sub forma

$$\underbrace{d(C_1, C)}_{=\ldots} < \underbrace{d(C_2, C)}_{=\ldots} \text{ şi } d(C_1, C) < \underbrace{d(C_1, C_2)}_{=\ldots}, \qquad (*)$$

unde d va fi una dintre „măsurile de distanţă“ d_{SL}, d_{CL} sau d_{AL}, corespunzătoare măsurilor de similaritate single-linkage, complete-linkage şi respectiv average-linkage.

Răspuns:

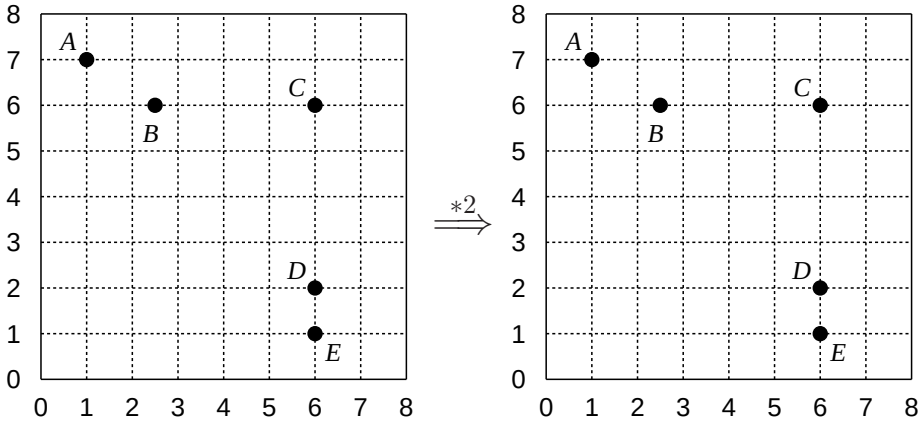
Single-linkage:



Iteraţiile 1 şi 2. **Iteraţia 3.**

$*1$**:**

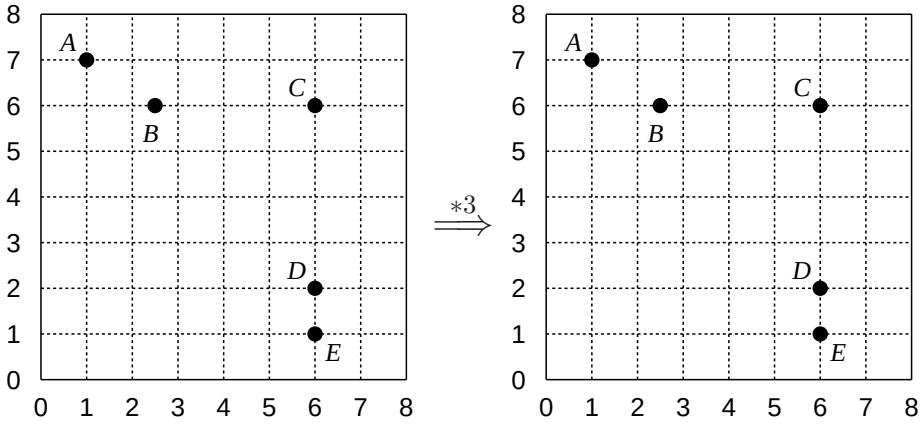
Complete-linkage:



Iteraţiile 1 şi 2. **Iteraţia 3.**

$*2$**:**

Average-linkage:



Iteraţiile 1 şi 2. **Iteraţia 3.**

$*3$**:**

---

3. [1.75p] (Algoritmul K-means: chestiuni legate de valoarea criteriului J)

Vă readucem aminte că la clusterizare cu algoritmul K-means obiectivul este să găsim poziţiile celor K centroizi ai clusterelor, notaţi cu $c_j \in \mathbb{R}^d, j \in \{1, \ldots, K\}$, astfel încât suma pătratelor distanţelor dintre fiecare instanţă x_i şi cel mai apropiat centroid să fie minimizată. Aşadar, funcţia obiectiv este

$$\sum_{i=1}^{n} \min_{j \in \{1,\ldots,K\}} \|x_i - c_j\|^2, \tag{1}$$

unde n este numărul de instanţe de clusterizat. Altfel spus, încercăm să găsim $c_1, \ldots, c_k$ astfel încât să fie minimizată expresia (1). Pentru aceasta, efectuăm mai multe iteraţii în cadrul cărora asignăm fiecare instanţă x_i la cel mai apropiat centroid şi apoi actualizăm poziţia fiecărui centroid c_j la media instanţelor asignate la clusterul j.

Însă prietenul tău Ionuţ, în loc să menţină numărul de clustere K fixat, încearcă să minimizeze valoarea expresiei (1) variindu-l pe K. Tu eşti de părere că această idee nu este bună.

În mod concret, tu îl convingi pe Ionuţ dându-i două valori: α, care reprezintă minimul valorilor posibile pentru expresia (1), şi β, care este valoarea lui K atunci când expresia (1) îşi atinge valoarea minimă.

a. Cât este valoarea expresiei $\alpha + \beta$ în cazul în care $n = 100$?

b. Presupunem că datele de clusterizat este format din 4 instanţe situate pe axa reală, şi anume $x_1 = 1$, $x_2 = 2$, $x_3 = 5$ şi $x_4 = 7$, iar K are valoarea 3. Cât este în acest caz valoarea optimă a funcţiei obiectiv (1)?

4. [1.25p] (Distribuţia Bernoulli: estimare în sensul verosimiltăţii maxime)

La curs am spus că la estimarea de verosimilitate maximă (engl., Maximum Likelihood Estimation, MLE), definim

$$\hat{\theta}_{MLE} = \arg\max_{\theta} P(\mathcal{D}|\theta) = \arg\max_{\theta} \ln P(\mathcal{D}|\theta),$$

unde $\mathcal{D}$ este setul de date pe care se estimează parametrul θ al distribuţiei de probabilitate P.

Imaginează-ţi că eşti un *data scientist* care lucrează pentru o firmă de publicitate (engl., advertisement). Această firmă a făcut recent o campanie de publicitate [care constă în publicarea unui anunţ publicitar pe ecranul calculatorului / telefonului unor anumite persoane], iar ţie ţi se cere să evaluezi succesul acestei campanii de publicitate.

În cadrul acestei campanii, au fost targetate / vizate N persoane. Variabila aleatoare Y_i primeşte valoarea $y_i = 1$ dacă persoana i a făcut "click" pe anunţul publicitar şi $y_i = 0$ în cazul contrar. În total, $\sum_{i=1}^{N} y_i = k$ persoane au decis să facă "click" pe anunţul publicitar. Vom presupune că probabilitatea ca persoana i să facă "click" pe anunţul publicitar respectiv este θ, iar probabilitatea ca persoana i să nu facă "click" pe anunţul publicitar este $1 - \theta$.

Desigur, $P(\mathcal{D}|\theta) = P(Y_1, \ldots, Y_N|\theta)$.

a. Dedu expresia analitică pentru calcularea lui $\hat{\theta}_{MLE}$.

b. Presupunem că $N = 100$ şi $k = 10$. Calculează $\hat{\theta}_{MLE}$.

5. [2.85p] **(O problemă à la C. Do şi S. Batzoglou: rezolvarea unei mixturi de doi vectori de distribuţii Bernoulli i.i.d. identificarea parametrilor şi a variabilelor neobservabile; scrierea funcţiei de log-verosimilitate a datelor complete)**

Să zicem că avem două monede, A şi B. Sarcina ta este aceea de a afla (a „estima") θ_A şi θ_B, probabilităţile de apariţie a feţei *stemă* (engl., head) pentru fiecare dintre cele două monede. Însă eu sunt cam răutacios şi nu-ţi dau voie să arunci tu monedele. În schimb, decid să procedez astfel: voi arunca eu însumi monedele şi după aceea îţi voi comunica ţie rezultatele aruncărilor. În mod concret, îţi voi spune ceva de forma următoare: am ales una dintre cele două monede (nu-ţi spun care anume),[1] am aruncat-o de 10 ori şi am obţinut în total de 7 ori stema şi de 3 ori banul. Apoi am ales din nou una dintre cele două monede (poate aceeaşi cu cea dinainte, poate nu), am aruncat-o de 10 ori şi am obţinut în total de 5 ori stema şi de 5 ori banul. În total, îţi comunic de N ori câte o astfel de informaţie. (Aşadar, la final vei dispune de rezultatele a $10\,N$ aruncări ale monedelor.)

Facem *presupunerea* că la fiecare dintre aceste N serii cele 10 aruncări sunt independente unele de altele şi că am reţinut ordinea / succesiunea rezultatelor obţinute în urma acestor aruncări şi ţi-o comunic.

a. Formulează aceasta ca pe o problemă de tip EM (Expectation-Maximization).
i. **Care sunt datele observabile?**
ii. **Care sunt variabilele neobservabile / ascunse / latente (engl., hidden variables)?**
iii. **Care sunt parametrii modelului (adică parametrii distribuţiilor probabiliste folosite)?**

b. Calculează [adică, stabileşte pas cu pas care este] expresia funcţiei de *log-verosimilitate a datelor complete* la iteraţia t, pentru această problemă.

(Atenţie! Nu ţi se cere să rezolvi efectiv problema EM.)

[1] Vom considera că moneda A este aleasă întotdeauna cu probabilitatea π, iar moneda B cu probabilitatea $1 - \pi$.