Inteligență Artificială

Bogdan Alexe
bogdan.alexe@fmi.unibuc.ro

Secția Tehnologia Informației, anul III, 2022-2023
Cursul 13

Recapitulare – cursul trecut

1. Căutare în jocuri cu incertitudine:
   - căutare ExpectiMax

2. Examenul final
   - organizarea examenului din acest an
   - subiectele din anii trecuți
   - rezolvările subiectelor din anii trecuți

Examen iarnă - evaluare și notare

Nota = min(Curs + Laborator + Proiect&Teme + BonusLab, 10)
4p 3p 3p maxim 1p

- Examen Curs (4 puncte) – scris, în sesiune, 2 ore + eventual bonus de la curs (nu puteți depăși 4p)

- Test Laborator (3 puncte) – în sesiune (în aceeași zi cu Examen Curs), 2 ore;

- Proiect (1,5 puncte) – proiect la prima parte (învățare automată, îl primiți în săptămâna 4), termen limită de predare – săptămâna 6, prezentarea proiectului în săptămâna 7.
- Teme (1,5 puncte) – teme la partea a doua (căutare informată și neinformată), prezentare în cadrul laboratorului.

Nu există praguri, note minime impuse. 4,99 înseamnă restanță.

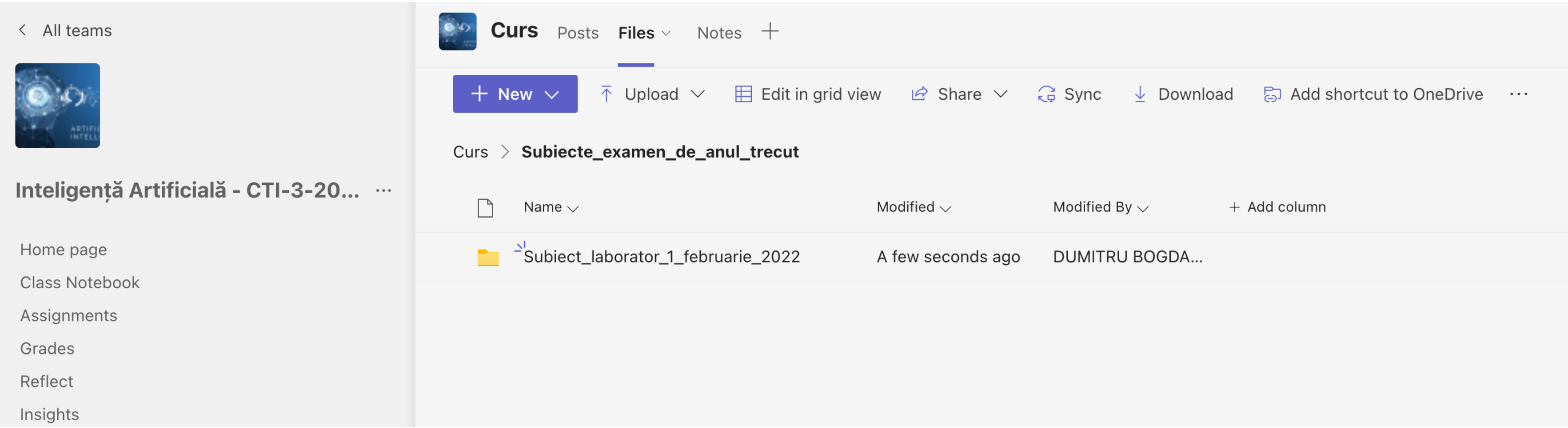
Organizarea examenului din acest an

- Examenul scris:
  - data 2 februarie, intre orele 9-11 in amf. Pompeiu si Titeica
  - open books vs closed books?

Inteligenta artificiala
Conf.dr. Alexe B.
Ora 9.00 – 11.00 Scris
Amf. Pompeiu si Titeica
Ora 12.00-14.00 – Laborator
Salile 201, 204, 303, 308

- Testul de laborator:
  - data 2 februarie, intre orele 12-14 in salile amf. Pompeiu si Titeica
  - veți primi un subiect cu două probleme (una din prima parte – învățare automată, una din a doua parte – căutare informată și neinformată)
  - aveți voie cu un stick cu resurse
  - nu aveți voie cu laptop, toată lumea dă examen pe calculatoarele din FMI
  - acces la internet restricționat

Organizarea examenului din acest an



O să afișăm până pe 1 februarie situația fiecărui student (notă proiect Kaggle, notă teme partea a doua, bonus laborator, bonus curs)

Subiectul din 27 ianuarie 2020

Examen la disciplina Inteligenţă Artificială
Seria 35, 27 ianuarie 2020

Subiectul 1. (3,5 puncte)

Considerăm o problemă de clasificare binară în care exemplele de antrenare sunt reprezentate de vectori bidimensionali cu etichetele binare 0 şi 1. Fie mulţimea de antrenare $S = \{((0,0)^T, 0), ((4,0)^T, 1), ((2,-1)^T, 1), ((2,2)^T, 1), ((5,3)^T, 0), ((-2,2)^T, 0)\}$. Fie mulţimea de testare $T = \{(1,1)^T, (5,5)^T\}$. Realizaţi următoarele:

(a) considerăm perceptronul din Figura 1a cu ponderile w_1 şi w_2 care ponderează intrările x_1 (abscisa) şi x_2 (ordonata) a unui punct, iar ponderea w_0 reprezintă deplasarea (bias-ul). Funcţia de activare a perceptronului este funcţia $\phi(x) = hardlim(x)$, unde $hardlim(x) = 1$, dacă $x \geq 0$ şi $hardlim(x) = 0$ dacă $x < 0$. Determinaţi matricea de confuzie a perceptonului cu ponderile $w_0 = -2, w_1 = 1, w_2 = 1$ care clasifică exemple din mulţimea S. **(1 punct)**

(b) mulţimea de antrenare S nu este liniar separabilă. Totuşi, dacă adăugăm la fiecare vector a treia coordonată, egală cu eticheta exemplului, obţinem o nouă mulţime de antrenare care acum este liniar separabilă. Arătaţi acest lucru, dând exemplu de un clasificator liniar care separă perfect exemplele de antrenare din noua mulţime. Puteţi folosi acest clasificator liniar pentru a eticheta exemple din mulţimea T? **(1 punct)**

(c) construiţi o reţea neuronală de perceptroni care să înveţe perfect mulţimea de antrenare. Etichetaţi punctele din mulţimea T folosind reţeaua astfel construită. **(1,5 puncte)**

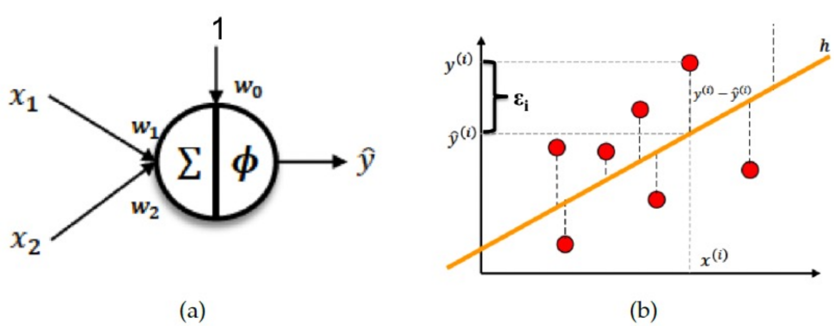
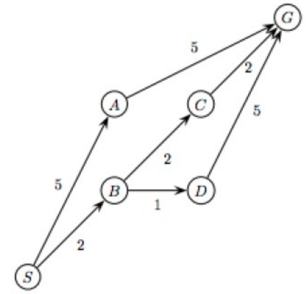


Figura 1: a. Arhitectura unui perceptron; b. Problema de regresie liniară prezentată la curs şi evidenţierea reziduului ε_i.



(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 2. (1 punct)

Fie $E = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$ o mulţime cu m exemple de antrenare, unde $x^{(i)}, y^{(i)}$ sunt numere reale pentru $i = 1, \ldots, m$. Regresia liniară simplă modelează legătura dintre variabila independentă x şi variabila dependentă y folosind dreapta de ecuaţie $h(x) = w_0 + w_1 \cdot x$. Parametri w_0 şi w_1 se determină folosind tehnica metodei celor mai mici pătrate aplicată pe mulţimea de antrenare E. La curs a fost arătat că parametri optimi au valorile următoare:

$$w_0 = \frac{1}{m}\sum_{i=0}^{m} y^{(i)} - w_1 \cdot \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$w_1 = \frac{\sum_{i=1}^{m}(x^{(i)} - \bar{x}) \cdot (y^{(i)} - \bar{y})}{\sum_{i=1}^{m}(x^{(i)} - \bar{x})^2},$$

unde $\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$, iar $\bar{y} = \frac{1}{m}\sum_{i=1}^{m} y^{(i)}$.

Pentru fiecare exemplu de antrenare din E notăm cu ε_i diferenţa dintre răspunsul corect $y^{(i)}$ şi predicţia $\hat{y}^{(i)}$ (vedeţi acest lucru ilustrat şi în Figura 1b). Cantitatea ε_i se mai numeşte *reziduu*. Arătaţi că suma tuturor reziduurilor este nulă, adică:

$$\sum_{i=1}^{m} \varepsilon_i = 0.$$

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. S este starea iniţială iar G este starea scop. O strategie de căutare care porneşte din starea S şi vrea să ajungă la starea G explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lăţime, starea A va fi explorată înaintea stării B). Considerăm euristicile h_0, h_1 şi h_2 date de tabelul din Figura 2b. Realizaţi următoarele:

(a) care este soluţia problemei de căutare folosind strategiile de căutare în lăţime şi căutare în adâncime? **(0,5 puncte)**

(b) care din euristicile h_0, h_1, h_2 din Figura 2b sunt admisibile? Justificaţi răspunsul **(0,5 puncte)**

(c) care este soluţia returnată de algoritmul A^* pentru fiecare euristică h_0, h_1, h_2 în parte? Specificaţi soluţiile parţiale construite până la obţinerea soluţiei finale. **(0,5 puncte)**

(d) care este soluţia returnată de algoritmul de căutare Greedy ce foloseşte euristica h_1? Specificaţi soluţiile parţiale construite până la obţinerea soluţiei finale. **(0,5 puncte)**

(e) care este soluţia returnată de algoritmul de căutare uniformă după cost (UCS)? Specificaţi soluţiile parţiale construite până la obţinerea soluţiei finale. **(0,5 puncte)**

Subiectul 4. (2 puncte)

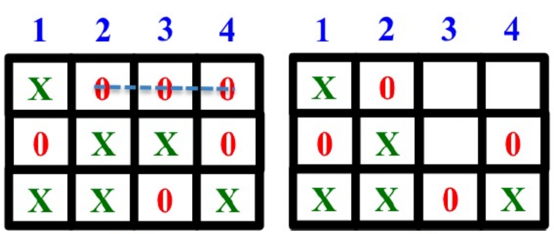
În jocul Conectează-3, jucătorii X şi 0 mută alternativ plasând simbolurile lor într-una din coloanele 1, 2, 3 sau 4 ale unui tablou cu 3 linii şi 4 coloane. Simbolurile se acumulează unul peste altul (Figura 3a). O coloană în care au fost plasate 3 simboluri devine plină şi prin urmare jucătorii nu mai pot plasa simboluri în ea. Câştigă jucătorul care realizează primul 3 simboluri dispuse pe orizontală, verticală sau diagonală. Dacă niciun jucător nu realizează acest lucru jocul se termină la egalitate. Jucătorul X mută primul.

(a) Care este numărul de mutări maxim (= factorul de ramificare) pe care îl are la dispoziţie fiecare jucător în orice moment al jocului? Justificaţi răspunsul. **(0,25 puncte)**

(b) Considerăm că după 9 mutări ajungem cu jocul în starea dată de Figura 3b. Jucătorul 0 este acum la mutare. Desenaţi arborele de joc asociat, considerând drept rădăcină starea actuală a jocului. **(0,75 puncte)**

(c) considerăm jucătorul X ca fiind MAX şi jucătorul 0 ca fiind MIN. Funcţia de utilitate asociată nodurilor terminale este următoarea: dacă jucătorul X câştigă atunci el primeşte k puncte iar jucătorul 0 pierde k puncte, dacă jocul se termină la egalitate atunci fiecare are 0 puncte. Valoarea k atunci când jucătorul X câştigă este stabilită astfel: $k = 3$ puncte pentru o linie realizată din simboluri dispuse pe diagonală, $k = 2$ pentru o linie realizată din simboluri dispuse pe verticală şi $k = 1$ pentru o linie realizată din simboluri dispuse pe orizontală. Valoarea k atunci când jucătorul 0 câştigă pe cazurile enumerate anterior este -3, -2, -1. Folosind algoritmul Minimax etichetaţi arborele de joc de la punctul anterior cu valorile minimax asociate fiecărui nod. **(0,5 puncte)**

(d) Folosiţi algoritmul de retezare Alfa-Beta pentru accelerarea calculului valorilor Minimax. Ce noduri din arbore de joc nu vor fi explorate? Puteţi reordona fiii fiecărui subarbore pentru a maximiza numărul de noduri care nu vor fi explorate de algoritmul Alfa-Beta? Justificaţi răspunsul. **(0,5 puncte)**



(a) (b)

Figura 3: a. Exemplu de stare finală (nod de tip frunză în arborele de joc) cu utilitatea -1 (câştigă jucătorul 0 ce realizează o linie pe orizontală); b. Stare curentă în care se ajunge după 9 mutări, acum jucătorul 0 este la mutare.

Oficiu: 1 punct. Timp de lucru: 2 ore.

Cursul de azi

1. Examenul final

    - rezolvările subiectelor din anii trecuți

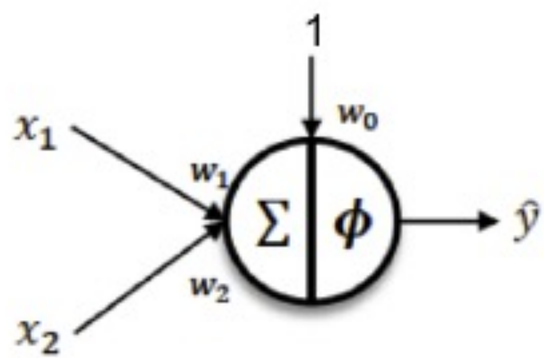
2. Prezentare curs opțional CAVA

Subiectul 1a - enunt

Subiectul 1. (3,5 puncte)

Considerăm o problemă de clasificare binară în care exemplele de antrenare sunt reprezentate de vectori bidimensionali cu etichetele binare 0 şi 1. Fie mulţimea de antrenare $S = \{((0,0)^T,0),((4,0)^T,1),((2,-1)^T,1),((2,2)^T,1),((5,3)^T,0),((-2,2)^T,0)\}$. Fie mulţimea de testare $T = \{(1,1)^T,(5,5)^T\}$. Realizaţi următoarele:

(a) considerăm perceptronul din Figura 1a cu ponderile w_1 şi w_2 care ponderează intrările x_1 (abscisa) şi x_2 (ordonata) a unui punct, iar ponderea w_0 reprezintă deplasarea (bias-ul). Funcţia de activare a perceptronului este funcţia $\phi(x) = hardlim(x)$, unde $hardlim(x) = 1$, dacă $x \geq 0$ şi $hardlim(x) = 0$ dacă $x < 0$. Determinaţi matricea de confuzie a perceptonului cu ponderile $w_0 = -2, w_1 = 1, w_2 = 1$ care clasifică exemple din mulţimea S. **(1 punct)**

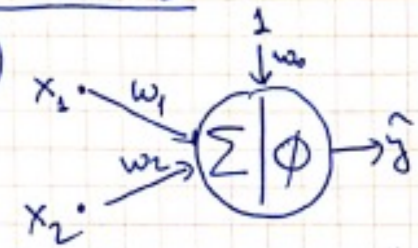


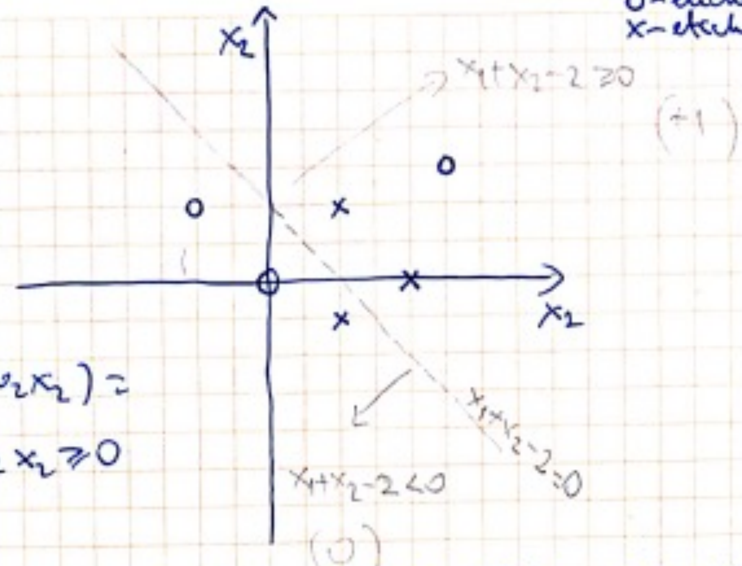
(a)

Subiectul 1

0 - eticheta 0
x - eticheta 1

a)





$$\hat{y} = \text{hardlim}(w_0 + w_1 \cdot x_1 + w_2 x_2) =$$

$$= \begin{cases} 1, & w_0 + w_1 x_1 + w_2 x_2 \geq 0 \\ 0, & \text{altfel} \end{cases}$$

$S = \{((0,0)^T, 0), ((4,0)^T, 1), \dots ((-2,2)^T, 0)\}$

$T = \{(1,1)^T, (5,5)^T\}$

$w_0 = -2,\ w_1 = 1,\ w_2 = 1$

$$\hat{y} = \begin{cases} 1, & x_1 + x_2 - 2 \geq 0 \\ 0, & x_1 + x_2 - 2 < 0 \end{cases}$$

$x_1 = 0$	$x_1 = 4$	$x_1 = 2$	$x_1 = 2$	$x_1 = 5$	$x_1 = -2$
$x_2 = 0$	$x_2 = 0$	$x_2 = -1$	$x_2 = 2$	$x_2 = 3$	$x_2 = 2$
$y = 0$	$y = 1$	$y = 1$	$y = 1$	$y = 0$	$y = 0$
$\hat{y} = 0$	$\hat{y} = 1$	$\hat{y} = 0$	$\hat{y} = 1$	$\hat{y} = 1$	$\hat{y} = 0$

etichete reale y \\ etichete prezise $\hat{y}$	0	1
0	2	1
1	1	2

Matricea de confuzie asociată perceptronului

Subiectul 1b - enunt

Subiectul 1. (3,5 puncte)

Considerăm o problemă de clasificare binară în care exemplele de antrenare sunt reprezentate de vectori bidimensionali cu etichetele binare 0 şi 1. Fie mulţimea de antrenare $S = \{((0,0)^T, 0), ((4,0)^T, 1), ((2,-1)^T, 1), ((2,2)^T, 1), ((5,3)^T, 0), ((-2,2)^T, 0)\}$. Fie mulţimea de testare $T = \{(1,1)^T, (5,5)^T\}$. Realizaţi următoarele:

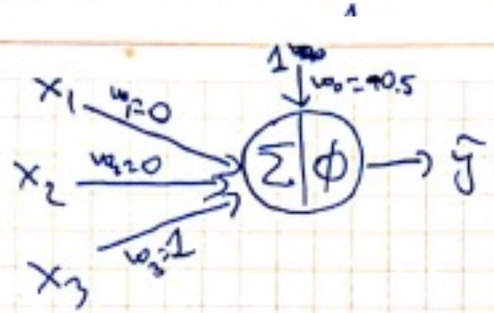
(b) mulţimea de antrenare S nu este liniar separabilă. Totuşi, dacă adăugăm la fiecare vector a treia coordonată, egală cu eticheta exemplului, obţinem o nouă mulţime de antrenare care acum este liniar separabilă. Arătaţi acest lucru, dând exemplu de un clasificator liniar care separă perfect exemplele de antrenare din noua mulţime. Puteţi folosi acest clasificator liniar pentru a eticheta exemple din mulţimea T? **(1 punct)**

Subiectul 1b - rezolvare

b) $S_{nou} = \{((1,0,0)^T,0),((4,0,1)^T,1),((2,-1,1)^T,1),((2,2,1)^T,1),$
$((5,3,0)^T,0),((-2,2,0)^T,0)\}$

Adaugăm practic dimensiunea x_3 = înălțime. Toate punctele pozitive primesc înălțimea 1, restul au înălțimea 0. Noua mulțime de antrenare este liniar separabilă, există o infinitate de hiperplanuri (planuri) care separă exemplele pozitive de cele negative.

O posibilitate este $w_0=-0.5$, $w_1=0$, $w_2=0$, $w_3=1$: $\boxed{x_3-0.5\geq 0}$



$$\hat{y} = hardlim(x_3-0.5) = \begin{cases} 1, & x_3 \geq 0.5 \\ 0, & x_3 < 0.5 \end{cases}$$

Nu putem folosi acest clasificator pentru a eticheta exemplele din mulțimea T întrucât lipsește a treia componentă (care reprezintă eticheta pe care vrem să o prezicem).

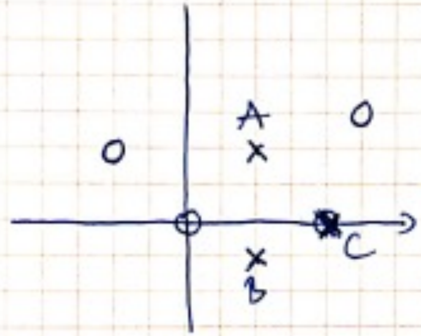
Subiectul 1c - enunt

Subiectul 1. (3,5 puncte)

Considerăm o problemă de clasificare binară în care exemplele de antrenare sunt reprezentate de vectori bidimensionali cu etichetele binare 0 şi 1. Fie mulţimea de antrenare $S = \{((0,0)^T, 0), ((4,0)^T, 1), ((2,-1)^T, 1), ((2,2)^T, 1), ((5,3)^T, 0), ((-2,2)^T, 0)\}$. Fie mulţimea de testare $T = \{(1,1)^T, (5,5)^T\}$. Realizaţi următoarele:

(c) construiţi o reţea neuronală de perceptroni care să înveţe perfect mulţimea de antrenare. Etichetaţi punctele din mulţimea T folosind reţeaua astfel construită. **(1,5 puncte)**

c) $A(2,2)$, $B(2,-1)$, $C(4,0)$



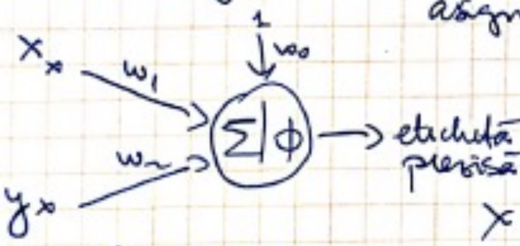
AB: $\frac{x-x_A}{y-y_A} = \frac{x_B-x_A}{y_B-y_A}$

$\frac{x-2}{y-2} = \frac{2-2}{-1-2} \Leftrightarrow x=2.$

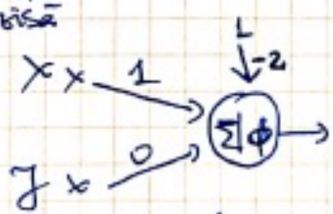
$x-2=0.$

Pentru dreapta AB am două posibile ecuații $\begin{cases} x-2=0 \\ -x+2=0. \end{cases}$

Pe care o aleg? (soluția mea presupune a construi o rețea care asignează 1 pentru toate punctele din triunghiul ABC).



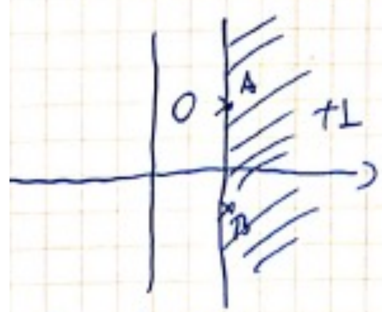
Dacă aleg $x-2=0$



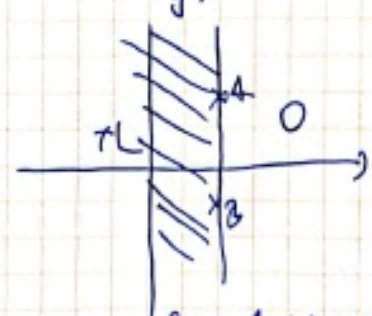
etichetă prezisă $= \begin{cases} 1, & x-2\geq 0 \\ 0, & x-2<0 \end{cases}$

Dacă aleg $-x+2=0$

etichetă prezisă $= \begin{cases} 1, & -x+2\geq 0 \\ 0, & -x+2<0 \end{cases}$



Cazul $x-2=0$



Cazul $-x+2=0$.

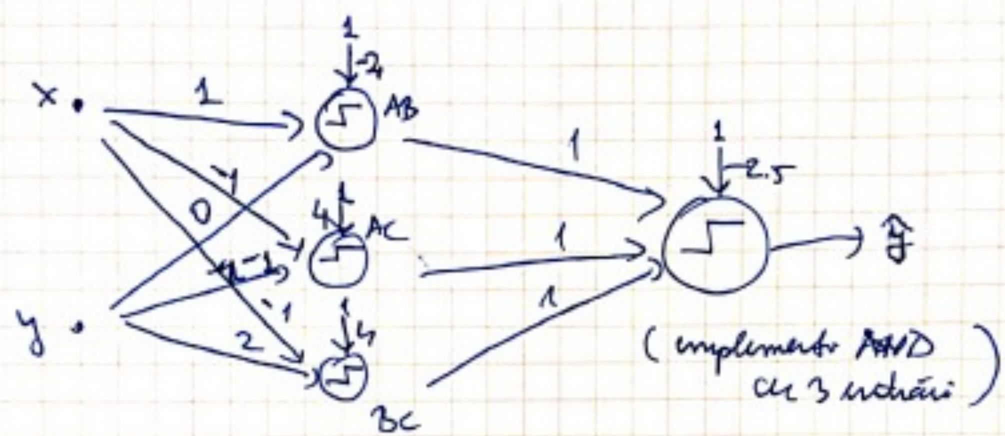
Subiectul 1c - rezolvare

Vreau ca punctul $C(4,0)$ să primească eticheta 1. Aleg varianta cu $\boxed{x-2=0}$

Analog, obținem ecuațiile pentru dreptele AC și BC

AC: $x+y-4=0$ $\boxed{-x-y+4=0}$

BC: $x-2y-4=0$ $\boxed{-x+2y+4=0}$



$T=\{(1,1)^T, (5,5)^T\}$. Ambele puncte vor fi etichetate ca 0.

Subiectul 2 - enunt

Subiectul 2. (1 punct)

Fie $E = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$ o mulţime cu m exemple de antrenare, unde $x^{(i)}, y^{(i)}$ sunt numere reale pentru $i = 1, \ldots, m$. Regresia liniară simplă modelează legătura dintre variabila independentă x şi variabila dependentă y folosind dreapta de ecuaţie $h(x) = w_0 + w_1 \cdot x$. Parametri w_0 şi w_1 se determină folosind tehnica metodei celor mai mici pătrate aplicată pe mulţimea de antrenare E. La curs a fost arătat că parametri optimi au valorile următoare:

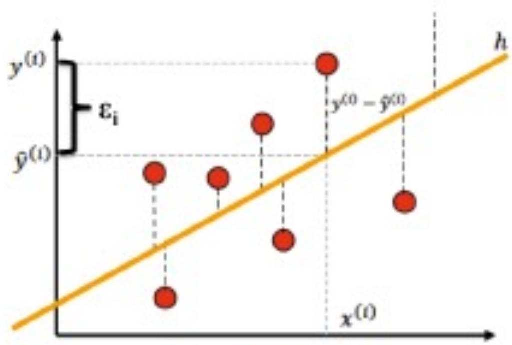
$$w_0 = \frac{1}{m}\sum_{i=0}^{m} y^{(i)} - w_1 \cdot \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$w_1 = \frac{\sum_{i=1}^{m}(x^{(i)} - \bar{x}) \cdot (y^{(i)} - \bar{y})}{\sum_{i=1}^{m}(x^{(i)} - \bar{x})^2},$$

unde $\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$, iar $\bar{y} = \frac{1}{m}\sum_{i=1}^{m} y^{(i)}$.

Pentru fiecare exemplu de antrenare din E notăm cu ε_i diferenţa dintre răspunsul corect $y^{(i)}$ şi predicţia $\hat{y}^{(i)}$ (vedeţi acest lucru ilustrat şi în Figura 1b). Cantitatea ε_i se mai numeşte *reziduu*. Arătaţi că suma tuturor reziduurilor este nulă, adică:

$$\sum_{i=1}^{m} \varepsilon_i = 0.$$



(b)

Subiectul 2 - rezolvare

Subiectul 2

$$E = \{ (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)}) \}$$

Regresia liniară ne conduce la dreapta de ecuație $h(x) = w_0 + w_1 x$

$$w_0 = \frac{1}{m} \sum_{i=1}^{m} y^{(i)} - w_1 \cdot \frac{1}{m} \sum_{i=1}^{m} x^{(i)} = \bar{y} - w_1 \cdot \bar{x}$$

$$w_1 = \frac{\sum_{i=1}^{m} (x^{(i)} - \bar{x}) \cdot (y^{(i)} - \bar{y})}{\sum_{i=1}^{m} (x^{(i)} - \bar{x})^2}$$

$$\bar{x} = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}, \quad \bar{y} = \frac{1}{m} \sum_{i=1}^{m} y^{(i)}$$

$$\varepsilon_i = y^{(i)} - \hat{y}^{(i)} = y^{(i)} - (w_0 + w_1 x^{(i)})$$

Arătați că $\sum_{i=1}^{m} \varepsilon_i = 0$.

$$\sum_{i=1}^{m} \varepsilon_i = \sum_{i=1}^{m} (y^{(i)} - w_0 - w_1 x^{(i)}) = \sum_{i=1}^{m} y^{(i)} - m \cdot w_0 - w_1 \cdot \sum_{i=1}^{m} x^{(i)}$$

$$= m \cdot \bar{y} - m(\bar{y} - w_1 \cdot \bar{x}) - w_1 \cdot m \cdot \bar{x}$$

$$= m\bar{y} - m\bar{y} + m\, w_1 \cdot \bar{x} - w_1 \cdot m \cdot \bar{x}$$

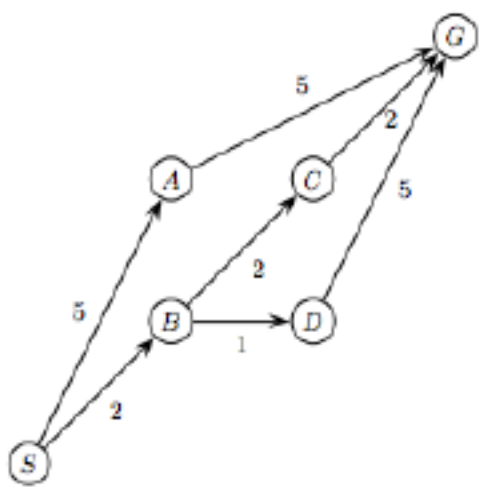
$$= 0.$$

Subiectul 3a – enunț

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. *S* este starea inițială iar *G* este starea scop. O strategie de căutare care pornește din starea *S* și vrea să ajungă la starea *G* explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lățime, starea *A* va fi explorată înaintea stării *B*). Considerăm euristicile h_0, h_1 și h_2 date de tabelul din Figura 2b. Realizați următoarele:

(a) care este soluția problemei de căutare folosind strategiile de căutare în lățime și căutare în adâncime? **(0,5 puncte)**



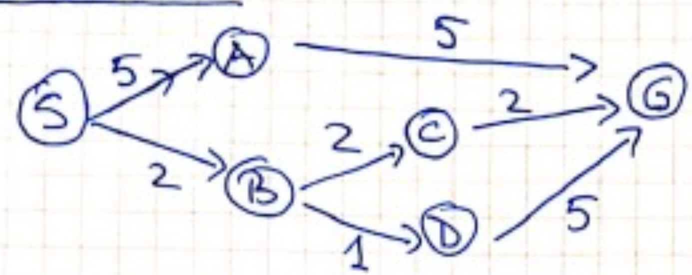
(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 3



Euristică / Stare	h_0	h_1	h_2	h^*
S	0	5	6	6
A	0	3	5	5
B	0	4	2	4
C	0	2	5	2
D	0	5	3	5
G	0	0	0	0

a) $F = \{S\}$ → frontiera inițială (conține soluțiile parțiale)

Pas 1: $F = \{\underline{S \to A}, S \to B\}$

(am expandat nodul S în ordine lexicografică pt căutarea în lățime)

Pas 2: $F = \{\underbrace{S \to A \to G}_{\text{soluție}}, S \to B\}$

Soluția căutării în lățime este $S \to A \to G$

Căutarea în adâncime:

$F = \{S\}$

Pas 1 $F = \{\underline{S \to A}, S \to B\}$

$F = \{\underbrace{S \to A \to G}_{\text{soluție}}, S \to B\}$

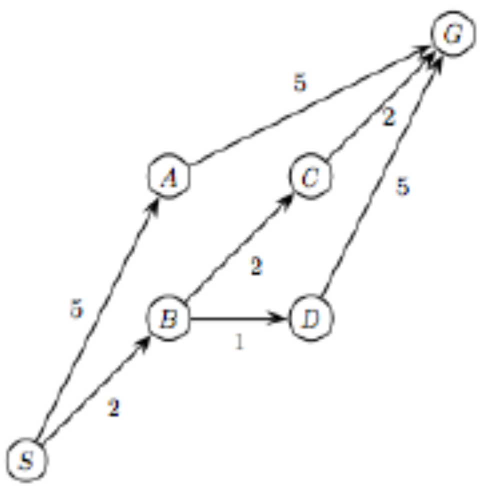
soluția căutării în adâncime este $S \to A \to G$

Subiectul 3b – enunț

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. S este starea inițială iar G este starea scop. O strategie de căutare care pornește din starea S și vrea să ajungă la starea G explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lățime, starea A va fi explorată înaintea stării B). Considerăm euristicile h_0, h_1 și h_2 date de tabelul din Figura 2b. Realizați următoarele:

(b) care din euristicile h_0, h_1, h_2 din Figura 2b sunt admisibile? Justificați răspunsul **(0,5 puncte)**



(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 3b – rezolvare

b) O euristică h este admisibilă dacă ea subestimează pentru orice nod n costul real al nodului n către cel mai apropiat nod scop

$$0 \le h(n) \le h^*(n)$$, unde $h^*(n)$ este costul real către cel mai apropiat nod scop.

Calculăm h^* pt fiecare nod:

$h^*(S)=6$, $h^*(A)=5$, $h^*(B)=4$, $h^*(C)=2$, $h^*(D)=5$, $h^*(G)=0$

Adaugăm coloana h^* în tabelul de sus.

Observăm că h_0 și h_1 sunt euristici admisibile, dar h_2 nu este admisibilă întrucât $h_2(C)=5 > h_2^*(C)=2$.

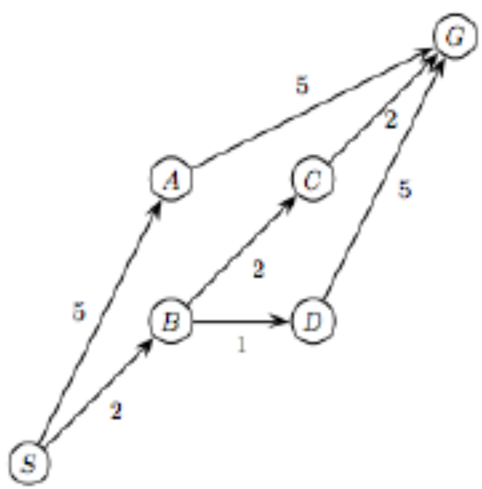
Euristică / Stare	h_0	h_1	h_2	h^*
S	0	5	6	6
A	0	3	5	5
B	0	4	2	4
C	0	2	5	2
D	0	5	3	5
G	0	0	0	0

Subiectul 3c – enunț

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. S este starea iniţială iar G este starea scop. O strategie de căutare care porneşte din starea S şi vrea să ajungă la starea G explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lăţime, starea A va fi explorată înaintea stării B). Considerăm euristicile h_0, h_1 şi h_2 date de tabelul din Figura 2b. Realizaţi următoarele:

(c) care este soluţia returnată de algoritmul A^* pentru fiecare euristică h_0, h_1, h_2 în parte? Specificaţi soluţiile parţiale construite până la obţinerea soluţiei finale. **(0,5 puncte)**



(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 3c – rezolvare

c) Pentru h_0 (= euristica banală) A^* se comportă ca UCS

$$f(n) = \underbrace{g(n)}_{\text{cost exact cât am făcut}} + \underbrace{h(n)}_{\text{cost estimat pe baza euristicii cât urmează să fac}}$$

Pas 0: $F = \{\underset{f(S)=0}{S}\}$

Pas 1: $F = \{\underset{f(S\to B)=2+0=2}{S\to B},\ \underset{f(S\to A)=5+0=5}{S\to A}\}$

Pas 2: $F = \{\underset{f(n)=3}{S\to B\to D},\ \underset{f(n)=4}{S\to B\to C},\ \underset{f(n)=5}{S\to A}\}$

Pas 3: $F = \{\underset{f(n)=8}{S\to B\to D\to G},\ \underset{f(n)=4}{S\to B\to C},\ \underset{f(n)=5}{S\to A}\}$

Pas 4: $F = \{\underset{f(n)=8}{S\to B\to D\to G},\ \underset{f(n)=6}{S\to B\to C\to G},\ \underset{f(n)=5}{S\to A}\}$

Pas 5: $F = \{\underset{f(n)=8}{S\to B\to D\to G},\ \underset{f(n)=6}{S\to B\to C\to G},\ \underset{f(n)=10}{S\to A\to G}\}$

Soluția este $S\to B\to C\to G$

Subiectul 3c – rezolvare

Pentru h_1

Pas 0: $F = \{S\}$

$f(n) = 0+5 = 5$

Pas 1: $F = \{S \to A,\ S \to B\}$

$f(n) = 5+3 = 8$, $\underline{f(n) = 2+4 = 6}$

Pas 2: $F = \{S \to A,\ S \to B \to C,\ S \to B \to D\}$

$f(n) = 8$, $\underline{f(n) = 4+2 = 6}$, $f(n) = 3+5 = 8$

Pas 3: $F = \{S \to A,\ S \to B \to C \to G,\ S \to B \to D\}$

$f(n) = 8$, $\underline{f(n) = 6+0 = 6}$, $f(n) = 3+5 = 8$

soluție

Soluția este $S \to B \to C \to G$

Subiectul 3c – rezolvare

Pentru h_2

Pas 0: $F = \{S\}$

$f(n) = 0+6 = 6$

Pas 1: $F = \{S \to A,\ S \to B\}$

$f(n) = 5+5 = 10$, $f(n) = 2+2 = 4$

Pas 2: $F = \{S \to A,\ S \to B \to C,\ S \to B \to D\}$

$f(n) = 10$, $f(n) = 4+5 = 9$, $f(n) = 3+3 = 6$

Pas 3: $F = \{S \to A,\ S \to B \to C,\ S \to B \to D \to G\}$

$f(n) = 10$, $f(n) = 9$, $f(n) = 8+0 = 8$

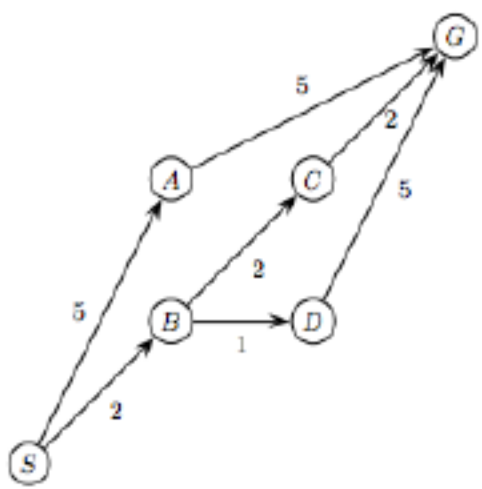
Soluția este $S \to B \to D \to G$.

Subiectul 3d – enunț

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. S este starea inițială iar G este starea scop. O strategie de căutare care pornește din starea S și vrea să ajungă la starea G explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lățime, starea A va fi explorată înaintea stării B). Considerăm euristicile h_0, h_1 și h_2 date de tabelul din Figura 2b. Realizați următoarele:

(d) care este soluția returnată de algoritmul de căutare Greedy ce folosește euristica h_1? Specificați soluțiile parțiale construite până la obținerea soluției finale. **(0,5 puncte)**



(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 3d – rezolvare

d) Greedy cu h_1

Pas 0: $F = \{S\}$

$h_1(S) = 5$

Pas 1: $F = \{\underset{h_1(n)=3}{S \to A}, \underset{h_1(n)=4}{S \to B}\}$

Pas 2: $F = \{\underset{h_1(n)=0}{S \to A \to G}, \underset{h_1(n)=4}{S \to B}\}$

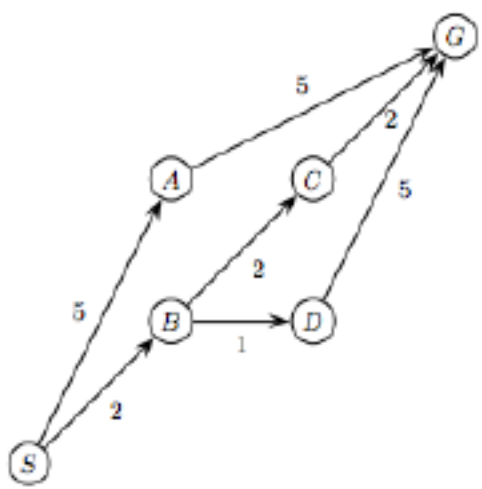
soluție

Subiectul 3e – enunț

Subiectul 3. (2.5 puncte)

Considerăm problema de căutare dată de graful din Figura 2a. S este starea inițială iar G este starea scop. O strategie de căutare care pornește din starea S și vrea să ajungă la starea G explorează stările într-o anumită ordine. Implicit, se preferă explorarea stărilor în ordine lexicografică (spre exemplu în căutarea în lățime, starea A va fi explorată înaintea stării B). Considerăm euristicile h_0, h_1 și h_2 date de tabelul din Figura 2b. Realizați următoarele:

(e) care este soluția returnată de algoritmul de căutare uniformă după cost (UCS)? Specificați soluțiile parțiale construite până la obținerea soluției finale. **(0,5 puncte)**



(a)

Stare	h_0	h_1	h_2
S	0	5	6
A	0	3	5
B	0	4	2
C	0	2	5
D	0	5	3
G	0	0	0

(b)

Figura 2: a. Graful asociat problemei de căutare; b. Euristicile h_0, h_1, h_2.

Subiectul 3e – rezolvare

e) UCS = A^* cu h_0 = punctul c) prima parte

c) Pentru h_0 (= euristica banală) A^* se comportă ca UCS

$f(n) = g(n) + h(n)$

$g(n)$: cost exact cât am făcut

$h(n)$: cost estimat pe baza euristicii cât urmează să fac

Pas 0: $F = \{ \underset{f(S)=0}{S} \}$

Pas 1: $F = \{ \underset{f(S\to B)=2+0=2}{S \to B},\ \underset{f(S\to A)=5+0=5}{S \to A} \}$

Pas 2: $F = \{ \underset{f(n)=3}{S \to B \to D},\ \underset{f(n)=4}{S \to B \to C},\ \underset{f(n)=5}{S \to A} \}$

Pas 3: $F = \{ \underset{f(n)=8}{S \to B \to D \to G},\ \underset{f(n)=4}{S \to B \to C},\ \underset{f(n)=5}{S \to A} \}$

Pas 4: $F = \{ \underset{f(n)=8}{S \to B \to D \to G},\ \underset{f(n)=6}{S \to B \to C \to G},\ \underset{f(n)=5}{S \to A} \}$

Pas 5: $F = \{ \underset{f(n)=8}{S \to B \to D \to G},\ \underset{f(n)=6}{S \to B \to C \to G},\ \underset{f(n)=10}{S \to A \to G} \}$

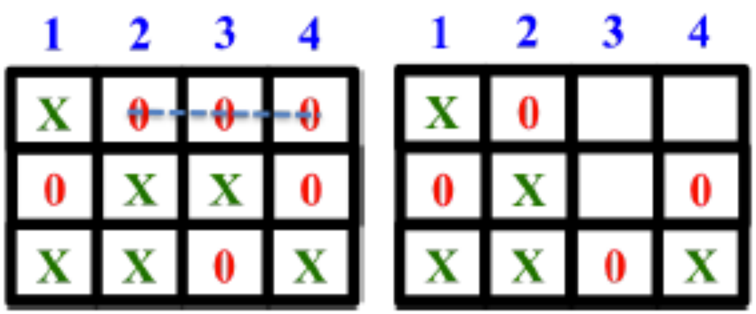
Soluția este $S \to B \to C \to G$

Subiectul 4a – enunț

Subiectul 4. (2 puncte)

În jocul Conectează-3, jucătorii X și 0 mută alternativ plasând simbolurile lor într-una din coloanele 1, 2, 3 sau 4 ale unui tablou cu 3 linii și 4 coloane. Simbolurile se acumulează unul peste altul (Figura 3a). O coloană în care au fost plasate 3 simboluri devine plină și prin urmare jucătorii nu mai pot plasa simboluri în ea. Câștigă jucătorul care realizează primul 3 simboluri dispuse pe orizontală, verticală sau diagonală. Dacă niciun jucător nu realizează acest lucru jocul se termină la egalitate. Jucătorul X mută primul.

(a) Care este numărul de mutări maxim (= factorul de ramificare) pe care îl are la dispoziție fiecare jucător în orice moment al jocului? Justificați răspunsul. **(0,25 puncte)**



(a) (b)

Figura 3: a. Exemplu de stare finală (nod de tip frunză în arborele de joc) cu utilitatea −1 (câștigă jucătorul 0 ce realizează o linie pe orizontală); b. Stare curentă în care se ajunge după 9 mutări, acum jucătorul 0 este la mutare.

Subiectul 4a – rezolvare

Subiectul 4

a) Numărul maxim de mutări (= factorul de ramificare) pe care îl are la dispoziție fiecare jucător = numărul de coloane libere. Inițial, toate coloanele sunt libere, deci $b = 4$, spre sfârșitul jocului se poate întâmpla ca numai o singură coloană să fie liberă, deci $b = 1$.

Subiectul 4b – enunț

Subiectul 4. (2 puncte)

În jocul Conectează-3, jucătorii X şi 0 mută alternativ plasând simbolurile lor într-una din coloanele 1, 2, 3 sau 4 ale unui tablou cu 3 linii şi 4 coloane. Simbolurile se acumulează unul peste altul (Figura 3a). O coloană în care au fost plasate 3 simboluri devine plină şi prin urmare jucătorii nu mai pot plasa simboluri în ea. Câştigă jucătorul care realizează primul 3 simboluri dispuse pe orizontală, verticală sau diagonală. Dacă niciun jucător nu realizează acest lucru jocul se termină la egalitate. Jucătorul X mută primul.

(b) Considerăm că după 9 mutări ajungem cu jocul în starea dată de Figura 3b. Jucătorul 0 este acum la mutare. Desenaţi arborele de joc asociat, considerând drept rădăcină starea actuală a jocului. **(0,75 puncte)**

1	2	3	4
X	0	0	0
0	X	X	0
X	X	0	X

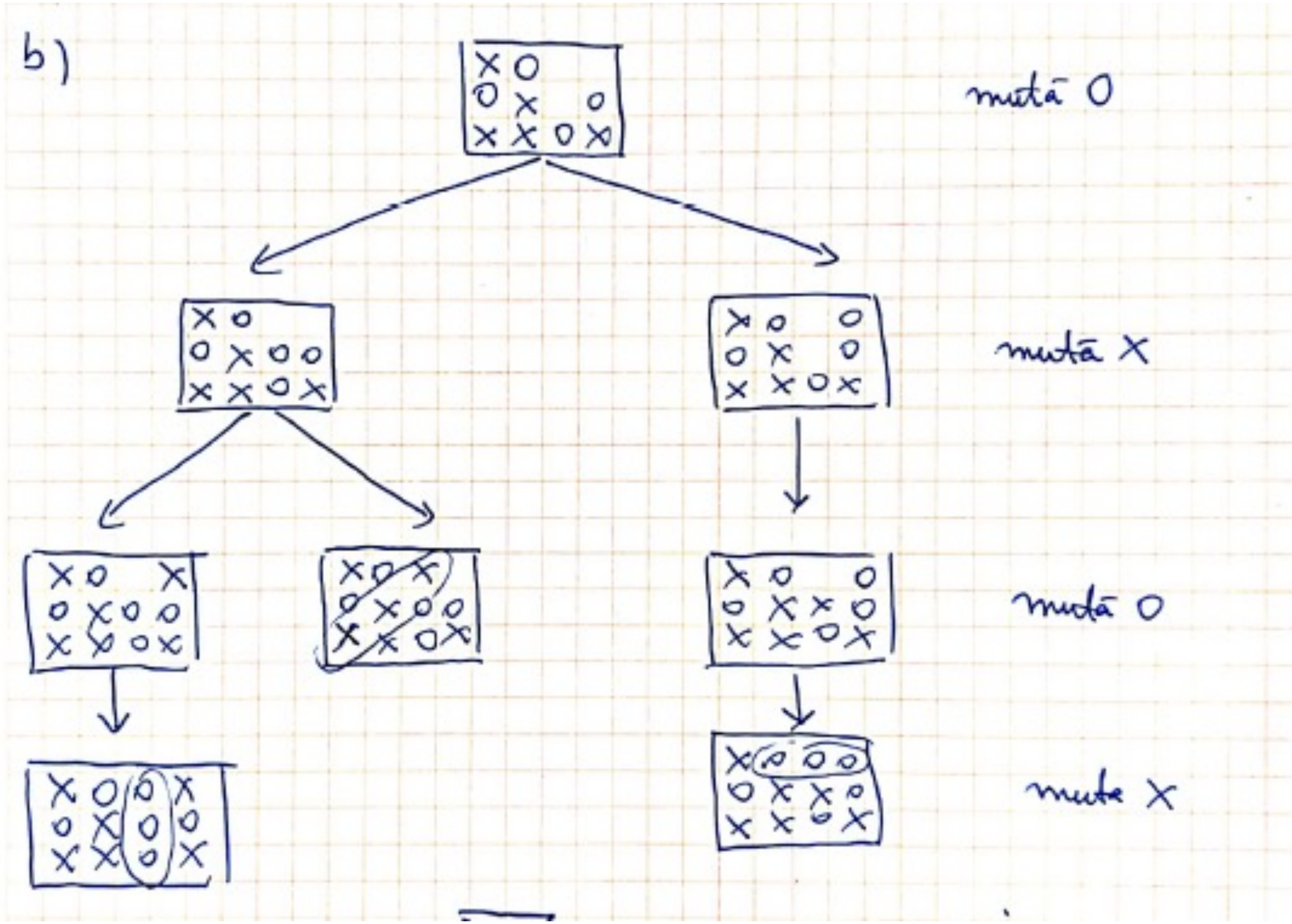
(a)

1	2	3	4
X	0		
0	X		0
X	X	0	X

(b)

Figura 3: a. Exemplu de stare finală (nod de tip frunză în arborele de joc) cu utilitatea −1 (câştigă jucătorul 0 ce realizează o linie pe orizontală); b. Stare curentă în care se ajunge după 9 mutări, acum jucătorul 0 este la mutare.

Subiectul 4b – rezolvare

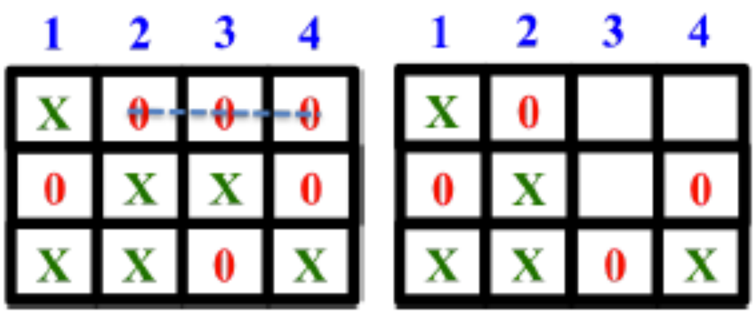


Subiectul 4c – enunț

Subiectul 4. (2 puncte)

În jocul Conectează-3, jucătorii X şi 0 mută alternativ plasând simbolurile lor într-una din coloanele 1, 2, 3 sau 4 ale unui tablou cu 3 linii şi 4 coloane. Simbolurile se acumulează unul peste altul (Figura 3a). O coloană în care au fost plasate 3 simboluri devine plină şi prin urmare jucătorii nu mai pot plasa simboluri în ea. Câştigă jucătorul care realizează primul 3 simboluri dispuse pe orizontală, verticală sau diagonală. Dacă niciun jucător nu realizează acest lucru jocul se termină la egalitate. Jucătorul X mută primul.

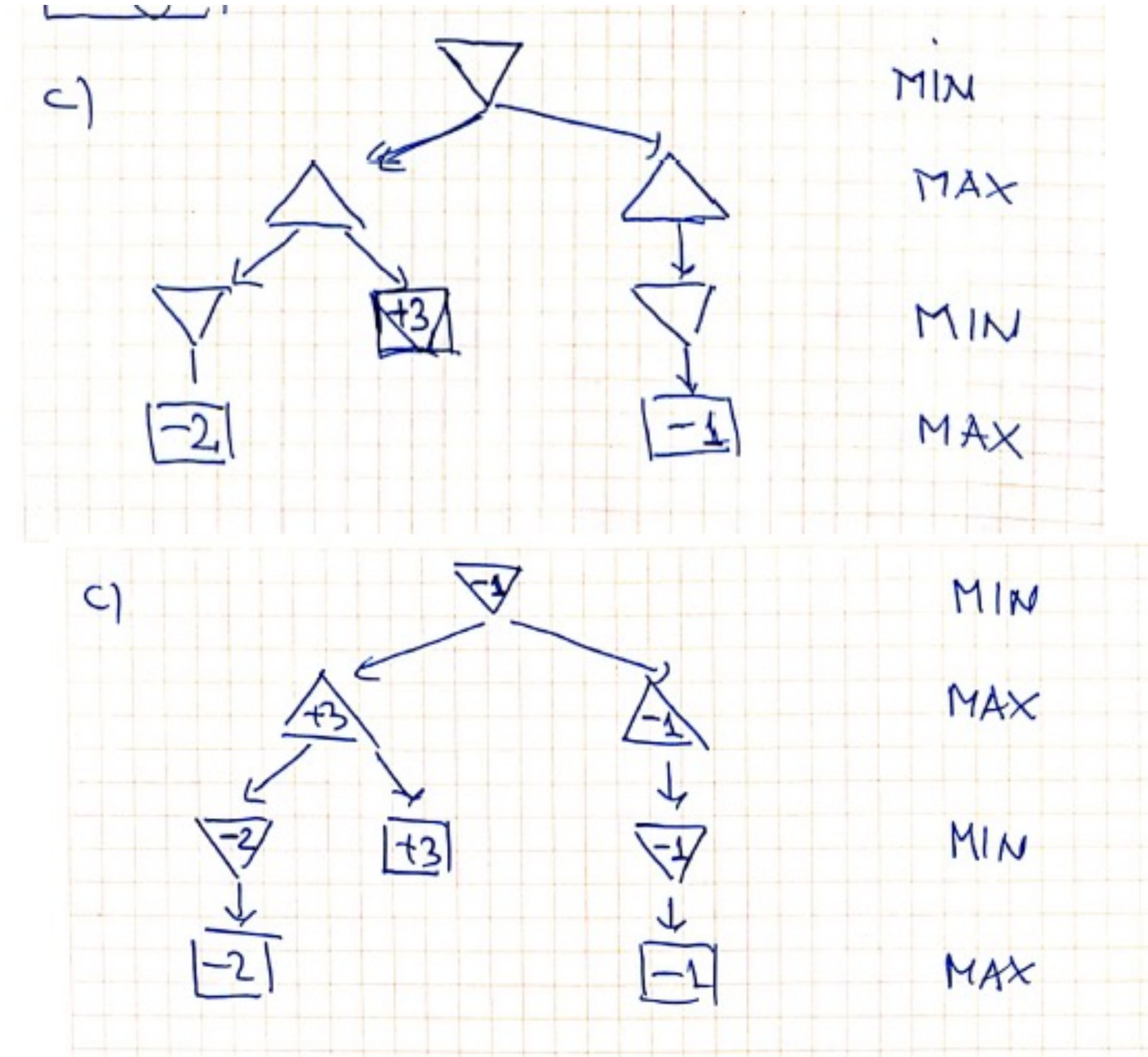
(c) considerăm jucătorul X ca fiind MAX şi jucătorul 0 ca fiind MIN. Funcţia de utilitate asociată nodurilor terminale este următoarea: dacă jucătorul X câştigă atunci el primeşte k puncte iar jucătorul 0 pierde k puncte, dacă jocul se termină la egalitate atunci fiecare are 0 puncte. Valoarea k atunci când jucătorul X câştigă este stabilită astfel: $k = 3$ puncte pentru o linie realizată din simboluri dispuse pe diagonală, $k = 2$ pentru o linie realizată din simboluri dispuse pe verticală şi $k = 1$ pentru o linie realizată din simboluri dispuse pe orizontală. Valoarea k atunci când jucătorul 0 câştigă pe cazurile enumerate anterior este -3, -2, -1. Folosind algoritmul Minimax etichetaţi arborele de joc de la punctul anterior cu valorile minimax asociate fiecărui nod. **(0,5 puncte)**



(a) (b)

Figura 3: a. Exemplu de stare finală (nod de tip frunză în arborele de joc) cu utilitatea -1 (câştigă jucătorul 0 ce realizează o linie pe orizontală); b. Stare curentă în care se ajunge după 9 mutări, acum jucătorul 0 este la mutare.

Subiectul 4c – rezolvare

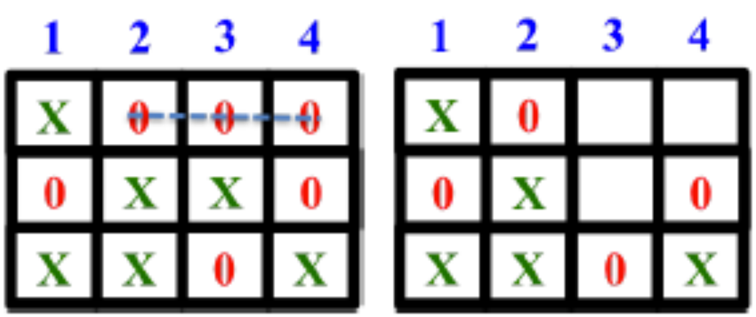


Subiectul 4d – enunț

Subiectul 4. (2 puncte)

În jocul Conectează-3, jucătorii X și 0 mută alternativ plasând simbolurile lor într-una din coloanele 1, 2, 3 sau 4 ale unui tablou cu 3 linii și 4 coloane. Simbolurile se acumulează unul peste altul (Figura 3a). O coloană în care au fost plasate 3 simboluri devine plină și prin urmare jucătorii nu mai pot plasa simboluri în ea. Câștigă jucătorul care realizează primul 3 simboluri dispuse pe orizontală, verticală sau diagonală. Dacă niciun jucător nu realizează acest lucru jocul se termină la egalitate. Jucătorul X mută primul.

(d) Folosiți algoritmul de retezare Alfa-Beta pentru accelerarea calcului valorilor Minimax. Ce noduri din arbore de joc nu vor fi explorate? Puteți reordona fiii fiecărui subarbore pentru a maximiza numărul de noduri care nu vor fi explorate de algoritmul Alfa-Beta? Justificați răspunsul. **(0,5 puncte)**

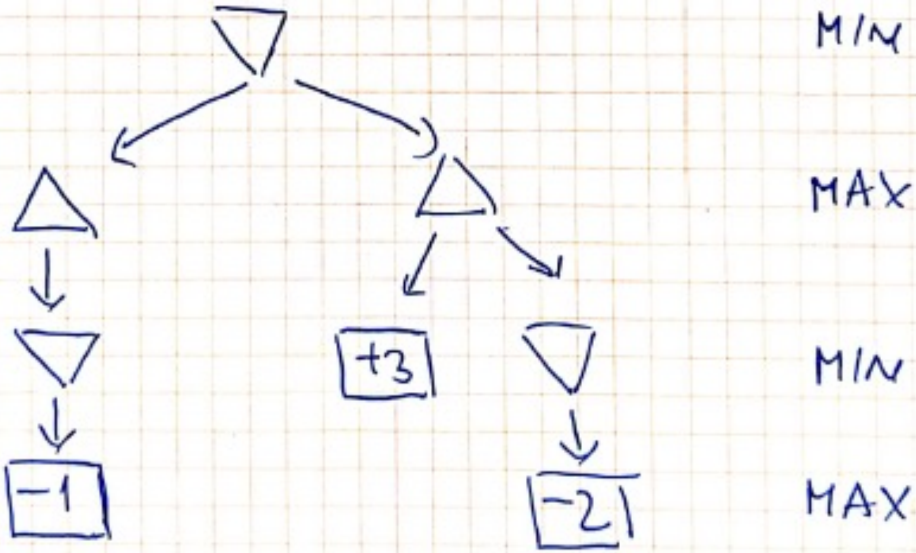


(a) (b)

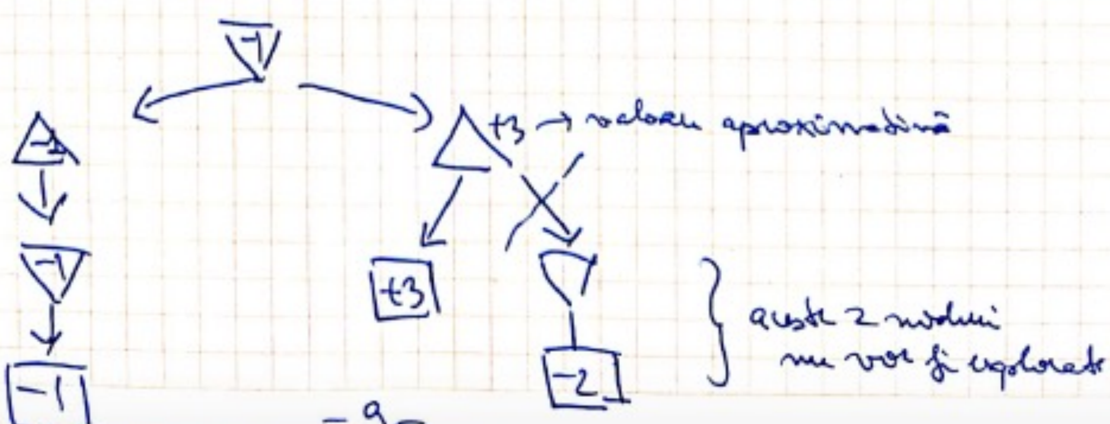
Figura 3: a. Exemplu de stare finală (nod de tip frunză în arborele de joc) cu utilitatea −1 (câștigă jucătorul 0 ce realizează o linie pe orizontală); b. Stare curentă în care se ajunge după 9 mutări, acum jucătorul 0 este la mutare.

d) Cu Alfa-Beta, în configurația actuală nu se poate face nicio retezare.

Rearanjăm arborele astfel:



Aplicăm Alfa Beta



3 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram clasificatorul binar bazat pe metoda celor mai apropiati k-vecini cu k=3. Fie multimea S de exemple de antrenare etichetate cu 6 puncte din plan unde S = {((3,0),0), ((3,-2),0), ((5,-2),0), ((7,-1),1), ((7,2),1), ((7,4),1)}, unde fiecare exemplu de antrenare e reprezentat de vectorul dat de coordonatele sale = (abscisa, ordonata) si de eticheta corespunzatoare (0 sau 1).

1. Dati exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare.
2. Dati exemplu de un punct care va fi clasificat cu eticheta 0 folosind distanta Manhattan. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare
3. Puteti da exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana si cu eticheta 0 folosind distanta Manhattan? Daca raspusnul e pozitiv dati exemplul, altfel justificati de ce nu puteti da exemplul.

3 întrebare

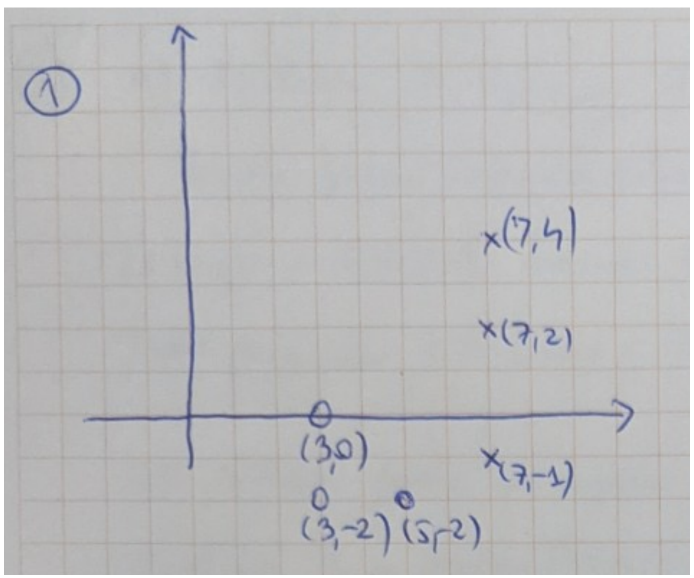
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram clasificatorul binar bazat pe metoda celor mai apropiati k-vecini cu k=3. Fie multimea S de exemple de antrenare etichetate cu 6 puncte din plan unde S = {((3,0),0), ((3,-2),0), ((5,-2),0), ((7,-1),1), ((7,2),1), ((7,4),1)}, unde fiecare exemplu de antrenare e reprezentat de vectorul dat de coordonatele sale = (abscisa, ordonata) si de eticheta corespunzatoare (0 sau 1).

1. Dati exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare.
2. Dati exemplu de un punct care va fi clasificat cu eticheta 0 folosind distanta Manhattan. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare
3. Puteti da exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana si cu eticheta 0 folosind distanta Manhattan? Daca raspusnul e pozitiv dati exemplul, altfel justificati de ce nu puteti da exemplul.



3 întrebare

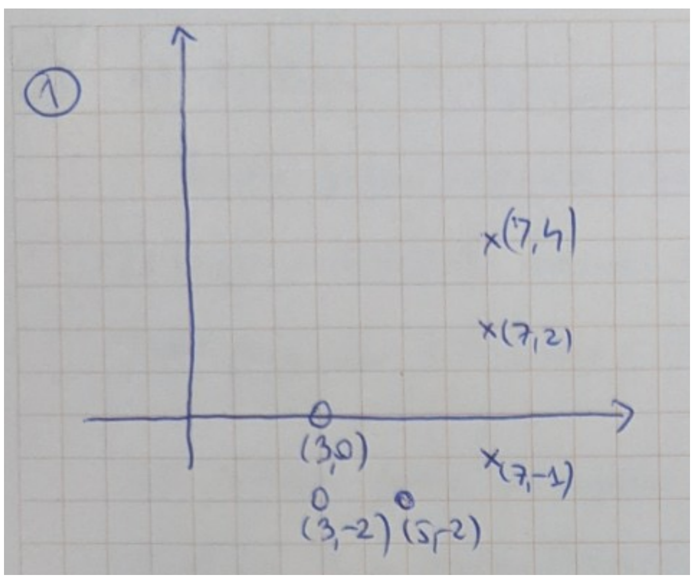
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram clasificatorul binar bazat pe metoda celor mai apropiati k-vecini cu k=3. Fie multimea S de exemple de antrenare etichetate cu 6 puncte din plan unde S = {((3,0),0), ((3,-2),0), ((5,-2),0), ((7,-1),1), ((7,2),1), ((7,4),1)}, unde fiecare exemplu de antrenare e reprezentat de vectorul dat de coordonatele sale = (abscisa, ordonata) si de eticheta corespunzatoare (0 sau 1).

1. Dati exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare.
2. Dati exemplu de un punct care va fi clasificat cu eticheta 0 folosind distanta Manhattan. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare
3. Puteti da exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana si cu eticheta 0 folosind distanta Manhattan? Daca raspunsul e pozitiv dati exemplul, altfel justificati de ce nu puteti da exemplul.



1. Consideram punctul (7,0).

Cei mai apropiati 3 vecini sunt:

- (7,-1) distanta 1 (distanta euclidiana)
- (7,2) distanta 1
- (5,-2) distanta $\sqrt{8}$ ($= \sqrt{(7-5)^2+(0-(-2))^2}$)

3 întrebare

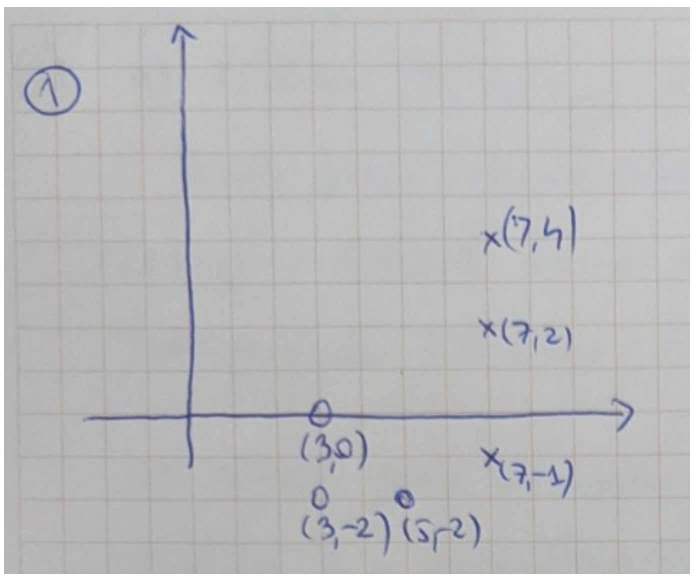
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram clasificatorul binar bazat pe metoda celor mai apropiati k-vecini cu k=3. Fie multimea S de exemple de antrenare etichetate cu 6 puncte din plan unde S = {((3,0),0), ((3,-2),0), ((5,-2),0), ((7,-1),1), ((7,2),1), ((7,4),1)}, unde fiecare exemplu de antrenare e reprezentat de vectorul dat de coordonatele sale = (abscisa, ordonata) si de eticheta corespunzatoare (0 sau 1).

1. Dati exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare.
2. Dati exemplu de un punct care va fi clasificat cu eticheta 0 folosind distanta Manhattan. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare
3. Puteti da exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana si cu eticheta 0 folosind distanta Manhattan? Daca raspusnul e pozitiv dati exemplul, altfel justificati de ce nu puteti da exemplul.



2. Considerăm punctul (3,0)

Cei mai apropiați 3 vecini în sensul distanței Manhattan sunt:

- (3,0) distanța Manhattan 0
- (3,-2) distanța Manhattan 2 (|3-3|+|0-(-2)|)
- (5,-2) distanța Manhattan 4 (|3-5|+|0-(-2)|)

3 întrebare

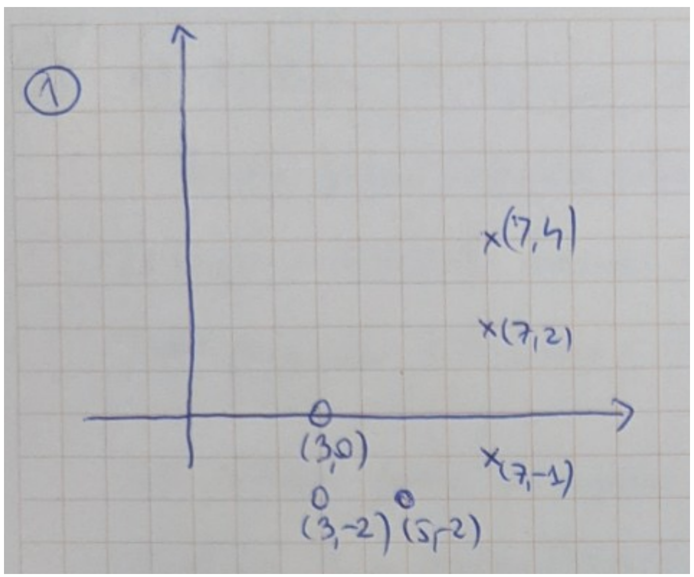
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram clasificatorul binar bazat pe metoda celor mai apropiati k-vecini cu k=3. Fie multimea S de exemple de antrenare etichetate cu 6 puncte din plan unde S = {((3,0),0), ((3,-2),0), ((5,-2),0), ((7,-1),1), ((7,2),1), ((7,4),1)}, unde fiecare exemplu de antrenare e reprezentat de vectorul dat de coordonatele sale = (abscisa, ordonata) si de eticheta corespunzatoare (0 sau 1).

1. Dati exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare.
2. Dati exemplu de un punct care va fi clasificat cu eticheta 0 folosind distanta Manhattan. Justificati raspunsul scriind care sunt cei mai apropiati 3 vecini si scriind distantele corespunzatoare
3. Puteti da exemplu de un punct care va fi clasificat cu eticheta 1 folosind distanta euclidiana si cu eticheta 0 folosind distanta Manhattan? Daca raspusnul e pozitiv dati exemplul, altfel justificati de ce nu puteti da exemplul.



3. Punctul (5,1) ;

- folosind distanta Euclidiana cei 3 vecini sunt (3,0) - eticheta 0 ($\sqrt{5}$)
  (7,2) - eticheta 1 ($\sqrt{5}$)
  (7,-1) - eticheta 1 ($\sqrt{8}$)
  => va fi clasificat cu eticheta 1

- folosind distanta Manhattan cei 3 vecini sunt: (3,0) - eticheta 0 (3)
  (7,2) - eticheta 1 (3)
  (5,-2) - eticheta 0 (3)
  => va fi etichetat cu eticheta 0

2 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram problema clasificarii literelor mici (clasele ‘a’, ‘b,’…,’z’) scrise de mana din alfabetul englez (in total 26 de clase) pe baza modelului Naïve Bayes. Datele de antrenare constau in 100 de exemple de antrenare cu imagini 28 x 28 pixeli pentru fiecare litera in parte. In total aveti 2600 exemple de antrenare. Consideram ca stim probabilitatea a-priori de aparitie a fiecarei litere pe baza studiului frecventei de aparitie a literelor in limba engleza. Pentru estimarea probabilitatii likelihood folosim modelul Naïve Bayes care considera ca fiecare pixel din imagine are o distributie normala specifica pozitiei pixelului din imagine si clasei literei.

1. cati parametri trebuie sa invete modelul vostru pentru a putea clasifica o litera 28 x 28?
2. cum ati invata parametri corespunzatori pixelului de pe linia 14, coloana 14 (centrul imaginii) din clasa ‘a’?

2 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram problema clasificarii literelor mici (clasele 'a', 'b,'…,'z') scrise de mana din alfabetul englez (in total 26 de clase) pe baza modelului Naïve Bayes. Datele de antrenare constau in 100 de exemple de antrenare cu imagini 28 x 28 pixeli pentru fiecare litera in parte. In total aveti 2600 exemple de antrenare. Consideram ca stim probabilitatea a-priori de aparitie a fiecarei litere pe baza studiului frecventei de aparitie a literelor in limba engleza. Pentru estimarea probabilitatii likelihood folosim modelul Naïve Bayes care considera ca fiecare pixel din imagine are o distributie normala specifica pozitiei pixelului din imagine si clasei literei.

1. cati parametri trebuie sa invete modelul vostru pentru a putea clasifica o litera 28 x 28?
2. cum ati invata parametri corespunzatori pixelului de pe linia 14, coloana 14 (centrul imaginii) din clasa 'a'?

2) a) 26 litere 'a', 'b', 'c', — 'z'

Pentru fiecare litera (in total 26) invatam pentru fiecare pozitie din cele 784 (28×28) o distributie normala $\mathcal{N}(\mu, \sigma)$

Total parametri: 26 litere × 784 pozitii × 2 parametri

b) Selectez toate exemplele (in numar de 100) cu litera 'a'. Selectez valorile pixelilor de pe pozitia (14,14) in total 100 de valori. Calculez apoi media μ si deviatia standard σ.

1 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Fie P = {((0,2),1), ((2,0),1), ((4,-2),1), ((1,5),0), ((3,3),0), ((5,1),0)} o multime de 6 exemple de antrenare de puncte din plan. Fiecare punct are o abscisa si o ordonata si o eticheta (0 sau 1).

1. Este multimea P liniar separabila? Justificati raspunsul.
2. Daca proiectam fiecare punct pe prima coordonata (abscisa) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Ox) liniar separabila? Justificati raspunsul.
3. Daca proiectam fiecare punct pe a doua coordonata (ordonata) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Oy) liniar separabila? Justificati raspunsul.

1 întrebare

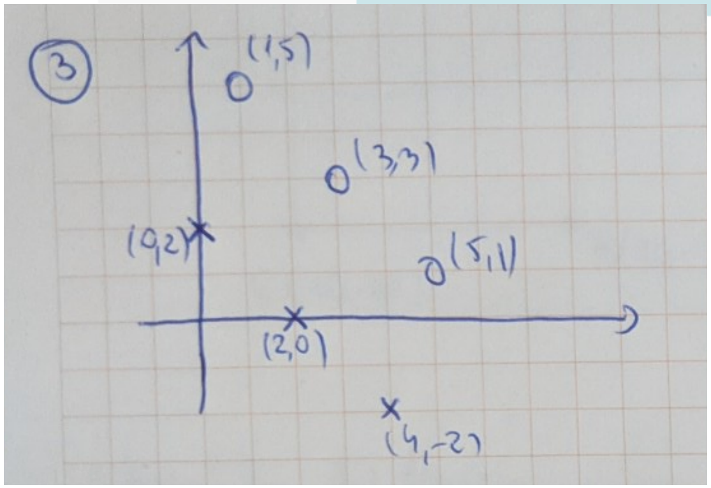
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Fie P = {((0,2),1), ((2,0),1), ((4,-2),1), ((1,5),0), ((3,3),0), ((5,1),0)} o multime de 6 exemple de antrenare de puncte din plan. Fiecare punct are o abscisa si o ordonata si o eticheta (0 sau 1).

1. Este multimea P liniar separabila? Justificati raspunsul.
2. Daca proiectam fiecare punct pe prima coordonata (abscisa) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Ox) liniar separabila? Justificati raspunsul.
3. Daca proiectam fiecare punct pe a doua coordonata (ordonata) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Oy) liniar separabila? Justificati raspunsul.



1 întrebare

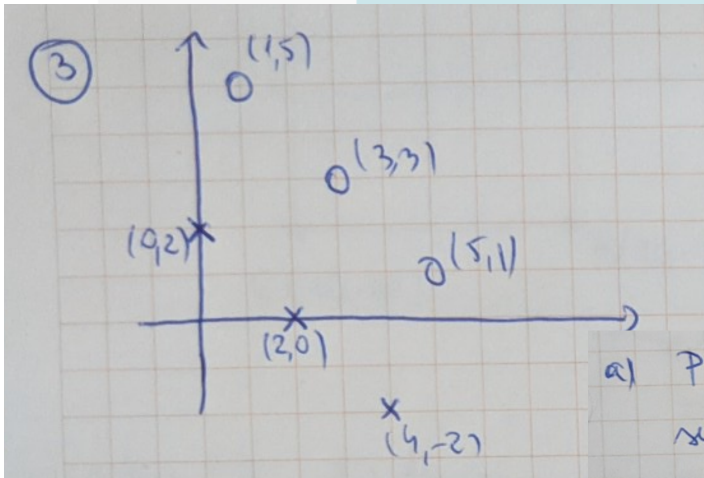
Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

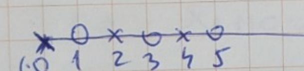
Fie P = {((0,2),1), ((2,0),1), ((4,-2),1), ((1,5),0), ((3,3),0), ((5,1),0)} o multime de 6 exemple de antrenare de puncte din plan. Fiecare punct are o abscisa si o ordonata si o eticheta (0 sau 1).

1. Este multimea P liniar separabila? Justificati raspunsul.
2. Daca proiectam fiecare punct pe prima coordonata (abscisa) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Ox) liniar separabila? Justificati raspunsul.
3. Daca proiectam fiecare punct pe a doua coordonata (ordonata) este multimea de exemple de antrenare nou obtinuta (cu puncte de pe axa Oy) liniar separabila? Justificati raspunsul.



a) P este liniar separabilă, spre exemplu dreapta de ecuație $y+x-4=0$ separă perfect clasele

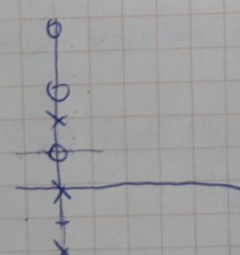
b) Obținem



$P' = \{(0,1), (2,1), (4,1), (1,0), (3,0), (5,0)\}$

Mulțimea P' nu mai este liniar separabilă, nu putem avea un hiperplan (=punct) care să separe perfect cele 2 clase

c) Obținem



$P'' = \{(2,1), (0,1), (-2,1), (5,0), (3,0), (1,0)\}$

Mulțimea P'' nu este liniar separabilă, nu putem avea un hiperplan (=punct) care să separe perfect cele 2 clase

10 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram o problema de clasificare binara cu puncte din **R** (dreapta reala). Fie multimea de exemple de antrenare P = {(-1,1), (0,0), (1,0), (2,0), (3,1), (4,1)} unde fiecare exemplu de antrenare e reprezentat de coordonata sa = abscisa si de eticheta corespunzatoare (0 sau 1). Scrieti ponderile w_0 (biasul), w_1 (ponderea pentru abscisa) a unui perceptron cu functia de activare hardlim care obtine acuratetea cea mai mare pe multimea P.

10 întrebare

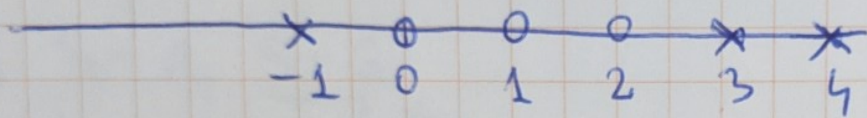
Nu a primit răspuns încă

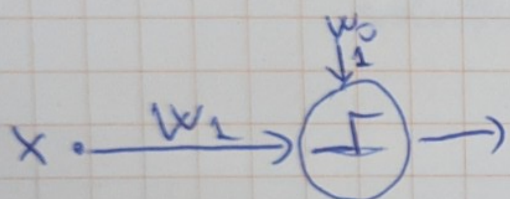
Marcat din 1,00

Întrebare cu flag

Consideram o problema de clasificare binara cu puncte din **R** (dreapta reala). Fie multimea de exemple de antrenare P = {(-1,1), (0,0), (1,0), (2,0), (3,1), (4,1)} unde fiecare exemplu de antrenare e reprezentat de coordonata sa = abscisa si de eticheta corespunzatoare (0 sau 1). Scrieti ponderile w_0 (biasul), w_1 (ponderea pentru abscisa) a unui perceptron cu functia de activare hardlim care obtine acuratetea cea mai mare pe multimea P.

4





$$y = \text{hardlim}(w_1 \cdot x + w_0) = \begin{cases} 1, & w_1 x + w_0 \geq 0 \\ 0, & \text{altfel} \end{cases}$$

$w_1 = 1$

$w_0 = -2.5$

$$\Rightarrow \quad y = \begin{cases} 1, & x \geq 2.5 \\ 0, & \text{altfel} \end{cases}$$

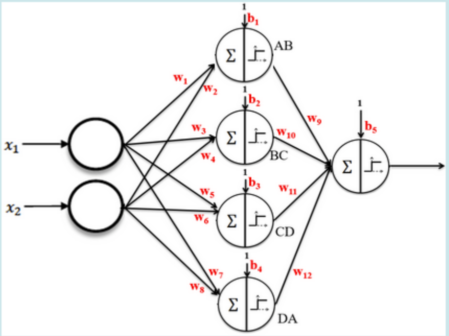
7 întrebare

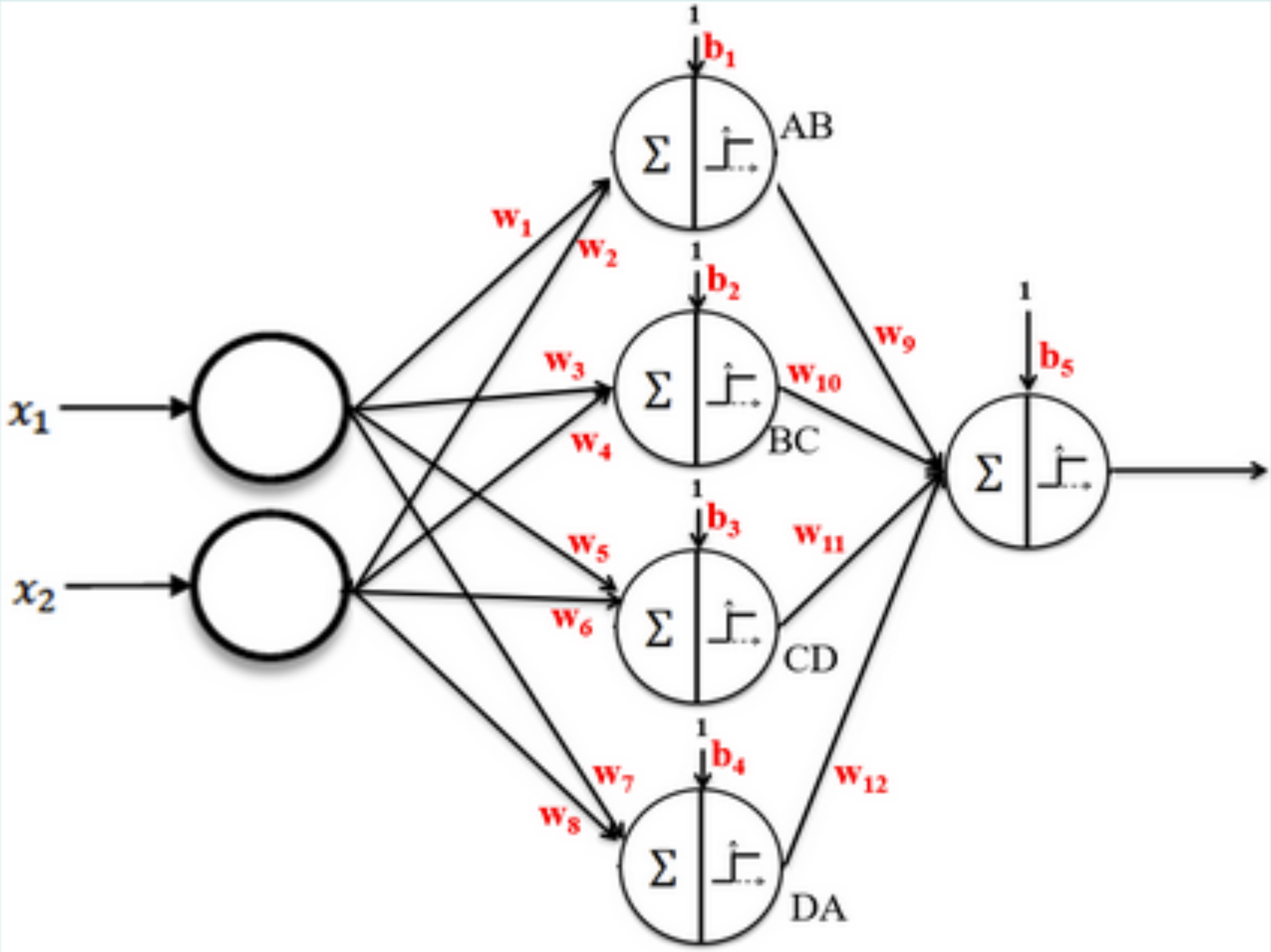
Nu a primit răspuns încă

Marcat din 1,00

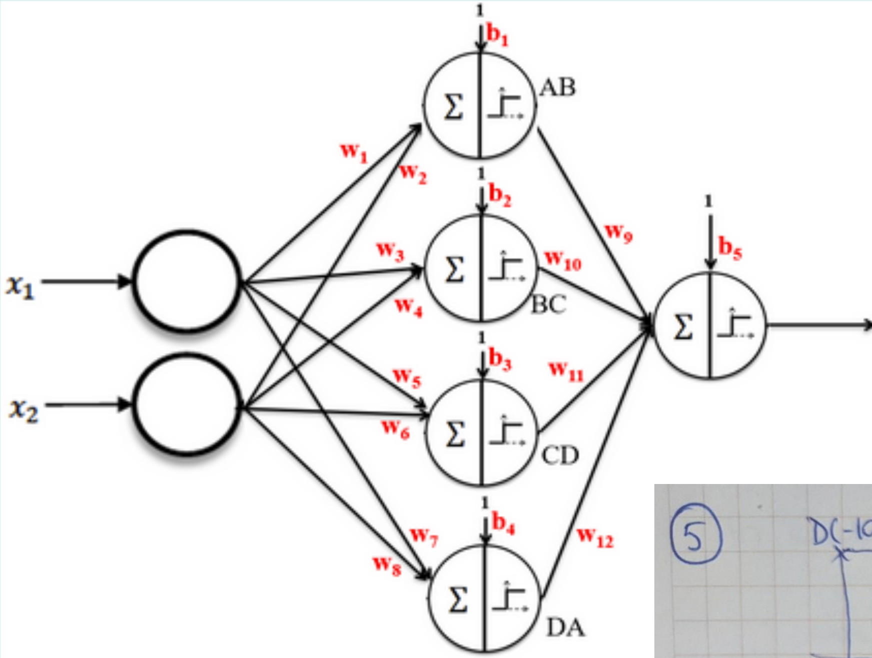
Întrebare cu flag

Consideram patratul de varfuri A(10,10), B(10,-10), C(-10,-10), D(-10,10) si reteaua neuronala din figura de mai jos care doreste a implementa functia indicator a patratului ABCD: pentru fiecare punct din interiorul sau de pe frontiera patratului reteaua ar trebui sa aiba iesirea 1, in rest iesirea retelei ar trebui sa fie 0. Reteaua neuronala de perceptroni are doua straturi ascunse. Pe primul strat ascuns sunt patru perceptroni care implementeaza ecuatiile dreptelor AB, BC, CD, DA astfel incat punctele din patrat primesc eticheta 1 din partea tututor celor 4 perceptroni. Al doilea strat ascuns are un singur perceptron care implementeaza functia AND de 4 variabile (iesirile perceptronilor de pe primul strat ascuns). Un punct este in patrat daca toti cei 4 perceptroni de pe primul strat au iesirea 1. Un punct din exteriorul patratului este caracterizat de faptul ca cel putin un perceptron de pe primul strat ascuns va avea iesirea 0. Functia de activare a tuturor perceptronilor din desen este hardlim. Scrieti un set de valori pentru ponderile w_1, w_2, …, w_12 si bias-urile b_1, b_2,…, b_5 astfel incat reteaua neuronala obtinuta sa implementeze functia indicator a patratului ABCD dupa logica descrisa.



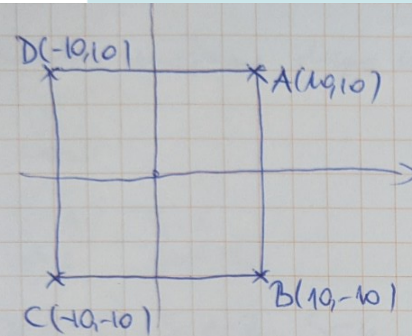


x_1
x_2
w_1
w_2
w_3
w_4
w_5
w_6
w_7
w_8
w_9
w_{10}
w_{11}
w_{12}
b_1
b_2
b_3
b_4
b_5
1
AB
BC
CD
DA
Σ



reste a implementa functia indicator a patratului ABCD: pentru fiecare punct din interiorul sau de pe frontiera patratului reteaua ar trebui sa aiba iesirea
sunt patru perceptroni care implementeaza ecuatiile dreptelor AB, BC, CD, DA astfel incat punctele din patrat primesc eticheta 1 din partea tututor celor
ptronilor de pe primul strat ascuns). Un punct este in patrat daca toti cei 4 perceptroni de pe primul strat au iesirea 1. Un punct din exteriorul patratului
perceptronilor din desen este hardlim. Scrieti un set de valori pentru ponderile w_1, w_2, …, w_12 si bias-urile b_1, b_2,…, b_5 astfel incat reteaua

5



AB: $x_1=10$, sau $-x_1=-10$

$x_1-10=0$, $-x_1+10=0$.

Perceptronul care implementeaza dreapta AB trebuie sa dea output-ul 1 pt toate pct din interiorul patratului. Toate punctele la stanga lui 10 trebuie sa primeasca 1 ($X \leq 10$). Deci ecuatia corecta este

$-x_1+10=0$ ($w_1=-1, w_2=0, b_1=10$).

Similar obtinem BC: $x_2+10=0$ ($w_3=0, w_4=1, b_2=10$)

CD: $x_1+10=0$ ($w_5=1, w_6=0, b_3=10$)

AD: $-x_2+10=0$ ($w_7=0, w_8=-1, b_4=10$)

Perceptronul de pe stratul 2 implementeaza AND, deci pot considera

$w_9=w_{10}=w_{11}=w_{12}=1$, $b_5=-3.5$ (sau -3.6, -3.7, -3.1, etc)

8 întrebare

Nu a primit răspuns încă

Marcat din 1,00

Întrebare cu flag

Consideram problema regresiei liniare simple pe multimea de antrenare S = {(3,0), (4,2), (6,0), (7,2)}. Notam cu w = (w_0,w_1) astfel incat y = w_0 + w_1*x este dreapta solutie a regresiei liniare. Cat este w?

8 întrebare

Nu a primit răspuns încă

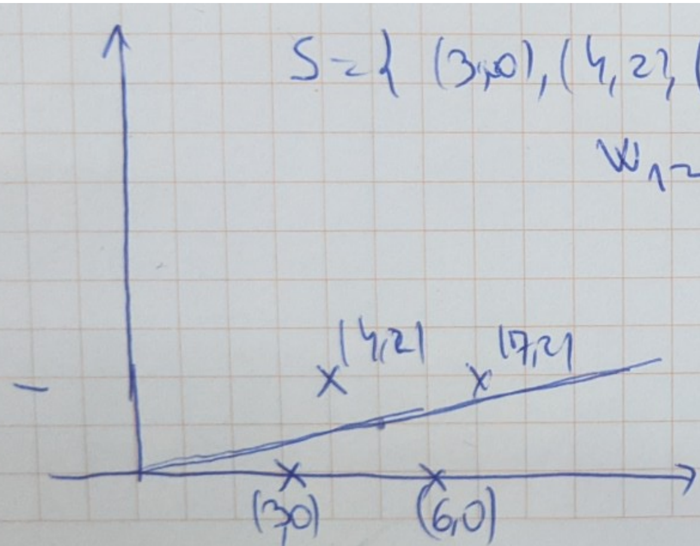
Marcat din 1,00

Întrebare cu flag

Consideram problema regresiei liniare simple pe multimea de antrenare S = {(3,0), (4,2), (6,0), (7,2)}. Notam cu w = (w_0,w_1) astfel incat y = w_0 + w_1*x este dreapta solutie a regresiei liniare. Cat este w?

6

$S = \{(3,0),(4,2),(6,0),(7,2)\}$ $w = (w_0, w_1)$ $\bar{x} = 5,\ \bar{y} = 1$



$$w_1 = \frac{\sum_i (x^i - \bar{x})(y^i - \bar{y})}{\sum_i (x^i - \bar{x})^2} = \frac{(3-5)(0-1) + \ldots + (7-5)\cdot(2-1)}{(3-5)^2 + \ldots (7-5)^2}$$

$$= \frac{2}{10} = \frac{1}{5}$$

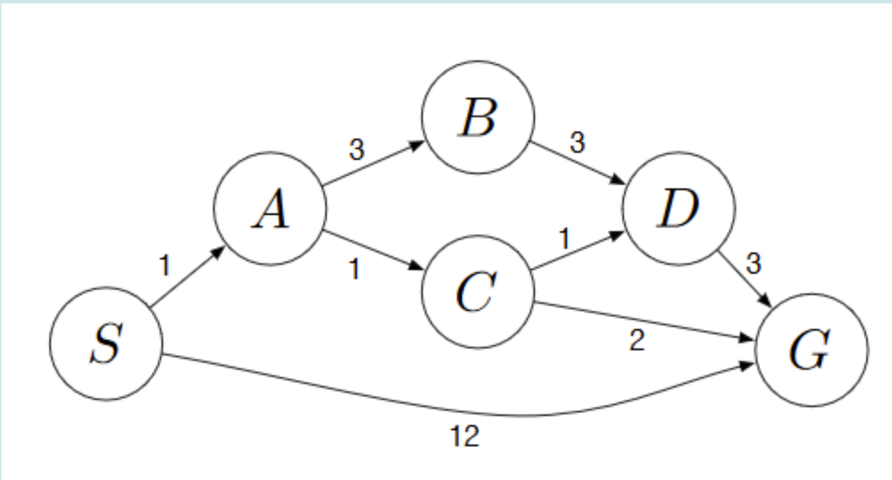
$$w_0 = \frac{1}{m}\left[\sum y^i - w_1 \sum x^i\right] = \bar{y} - w_1 \bar{x} =$$

$$= 1 - \frac{1}{5} \cdot 5 = 0$$

Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica. Realizati urmatoarele:

1. Care este solutia problemei de cautare folosind strategia de cautare in latime? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.
2. Care este solutia problemei de cautare folosind strategia de cautare uniforma dupa cost? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

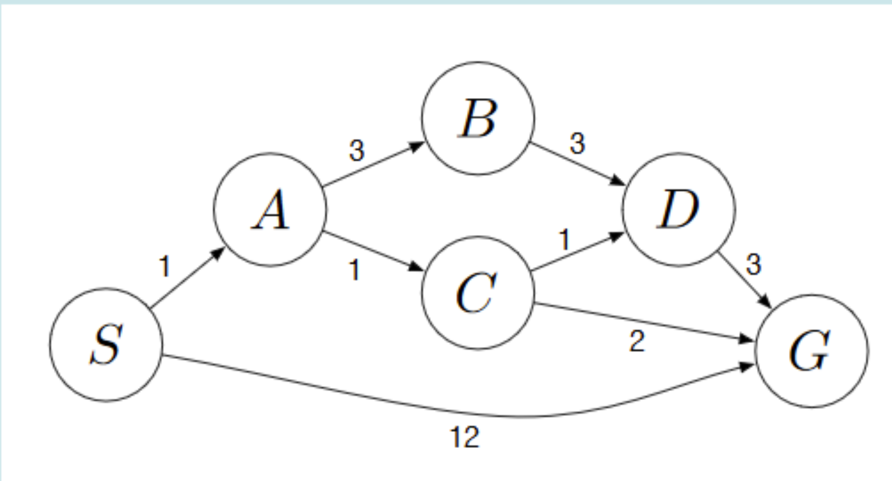
(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie partiala invalida pentru cazul de fata)



Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica. Realizati urmatoarele:

1. Care este solutia problemei de cautare folosind strategia de cautare in latime? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.
2. Care este solutia problemei de cautare folosind strategia de cautare uniforma dupa cost? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie partiala invalida pentru cazul de fata)



(17) a) Cautarea in latime:

S,

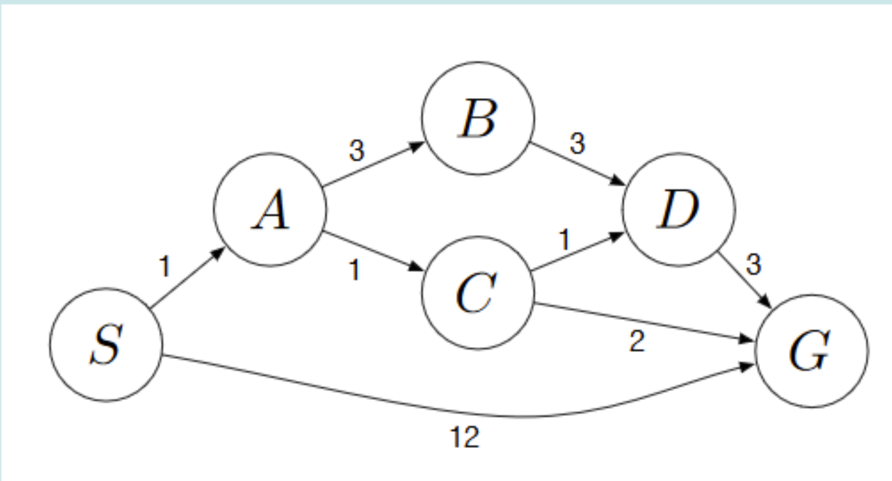
S-A, S-G

solutie

Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica. Realizati urmatoarele:

1. Care este solutia problemei de cautare folosind strategia de cautare in latime? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.
2. Care este solutia problemei de cautare folosind strategia de cautare uniforma dupa cost? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie partiala invalida pentru cazul de fata)



17 a) Cautarea in latime:

S,

S-A, S-G
solutie

b) cautare uniforma dupa cost

S
cost=0

S-A, S-G
cost 1, cost 12

S-A-C, S-A-B, S-G
2, 4, 12

S-A-C-D, S-A-C-G, S-A-B, S-G
3, 4, 4, 12

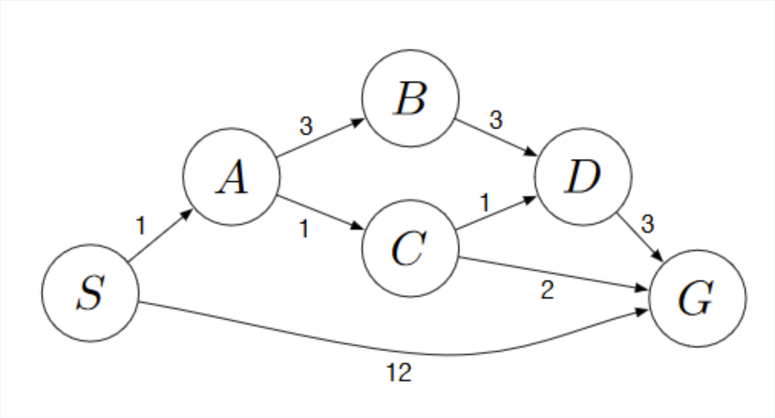
S-A-C-D-G, S-A-C-G, S-A-B, S-G
6, 4, 4

S-A-C-D-G, S-A-C-G, S-A-B-D, S-G
6, 4, 7, 12
solutie

Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica (la costuri egale). Consideram euristicile h_1 si h_2 din tabelul de mai jos. Realizati urmatoarele:

1. Care din euristicile h_1 si h_2 este admisibila? Justificati raspunsul.
2. Care este solutia problemei de cautare folosind algoritmul A* cu euristica h_1? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie invalida pentru cazul de fata)

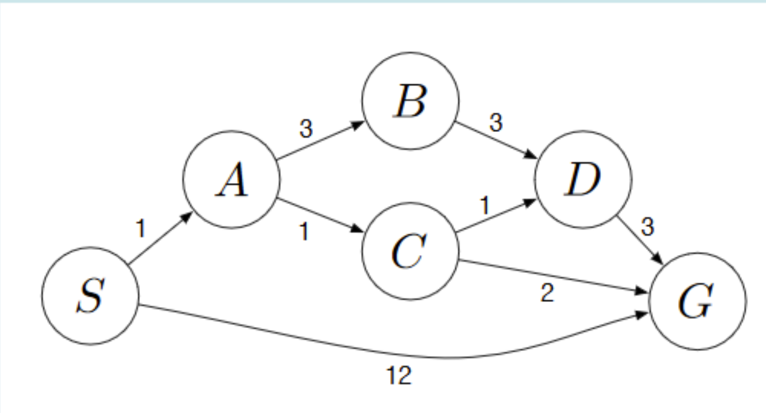


State	h_1	h_2
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	0

Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica (la costuri egale). Consideram euristicile h_1 si h_2 din tabelul de mai jos. Realizati urmatoarele:

1. Care din euristicile h_1 si h_2 este admisibila? Justificati raspunsul.
2. Care este solutia problemei de cautare folosind algoritmul A* cu euristica h_1? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie invalida pentru cazul de fata)



State	h_1	h_2
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	0

8 a) h este admisibilă dacă subestimează costul optim al fiecărui nod

$h^*(S) = 4$

$h_1(S) = 5 \Rightarrow h_1$ nu este admisibilă

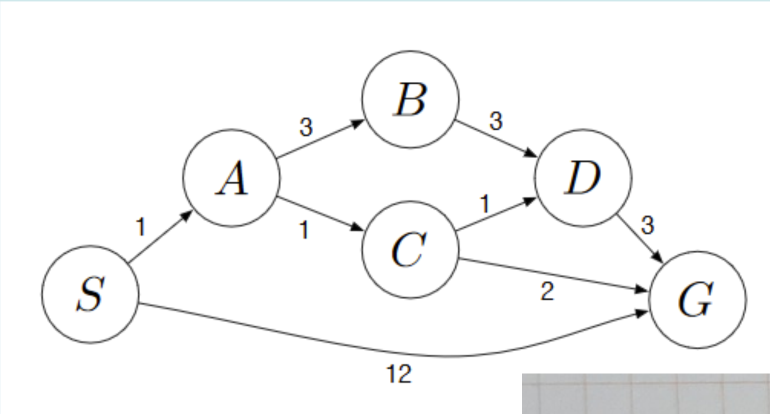
h_2 verifică condiția $h_2(N) \le h^*(N)$ pentru orice nod N

$\Rightarrow h_2$ este admisibilă

Consideram problema de cautare data de graful din figura de mai jos. S este starea initiala, G este starea scop. O strategie de cautare care porneste din starea S si vrea sa ajunga la starea G exploreaza starile intr-o anumita ordine. Implicit, se prefera explorarea starilor in ordine lexicografica (la costuri egale). Consideram euristicile h_1 si h_2 din tabelul de mai jos. Realizati urmatoarele:

1. Care din euristicile h_1 si h_2 este admisibila? Justificati raspunsul.
2. Care este solutia problemei de cautare folosind algoritmul A* cu euristica h_1? Specificati solutiile partiale construite in frontiera pana la obtinerea solutiei finale.

(Cand raspundeti la intrebari folositi ca notatie pentru o solutie partiala urmatoare notatie: S-A-D-G - solutie invalida pentru cazul de fata)



State	h_1	h_2
S	5	4
A	3	2
B	6	6
C	2	1
D	3	3
G	0	0

b) $f(n) = g(n) + h(n)$

traseu vecin
(exact) (estimat)

S (5)
g=0, h=5

S-A (4), S-G (12)
g=1, h=3 g=12, h=0

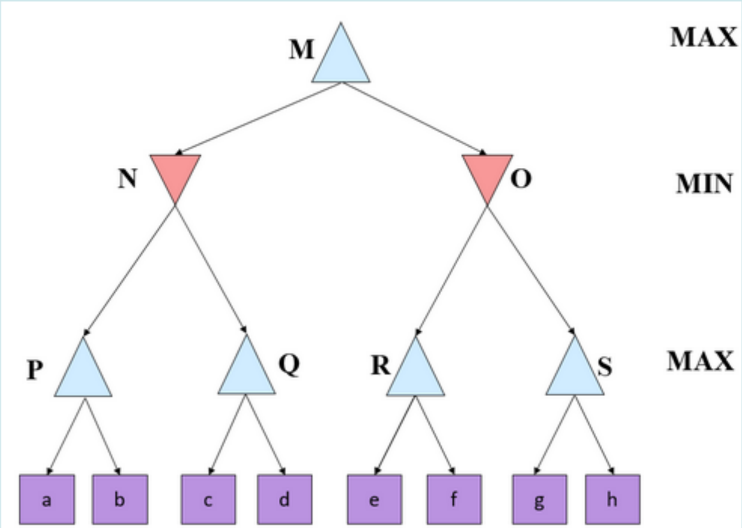
S-A-C (4), S-A-B (10), S-G (12)
g=2, h=2 g=4, h=6 g=12, h=0

S-A-C-D (6), S-A-C-G (4), S-A-B (10), S-G (12)
g=3, h=3 g=4, h=0 g=4, h=6 g=12, h=0

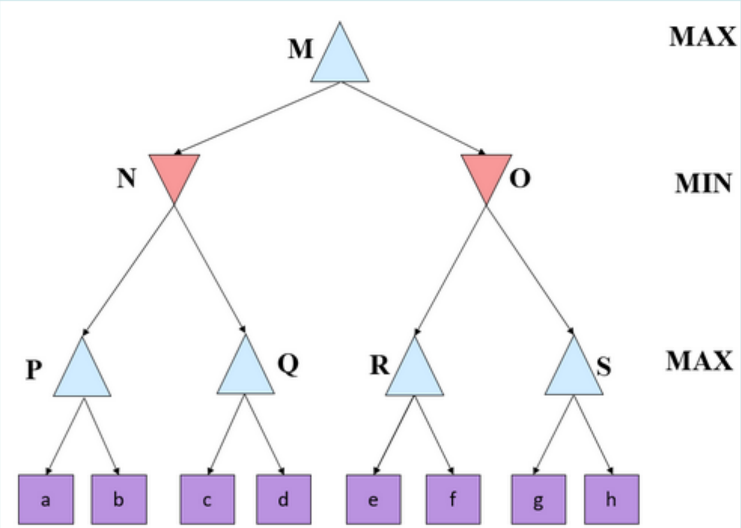
Solutie

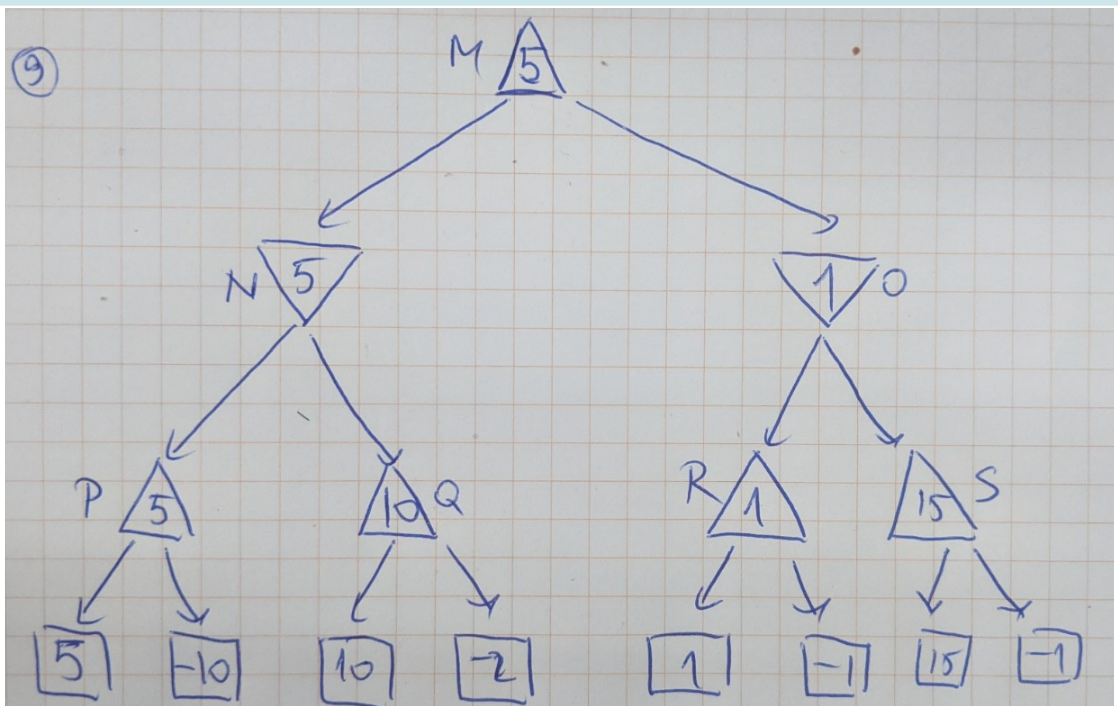
Stare	h_1
S	5
A	3
B	6
C	2
D	3
G	0

Consideram arborele de cautare din figura de mai jos, in care jucatorul la mutare este jucatorul MAX (vrea sa isi maximizeze castigul). Jucatorul MIN vrea sa minimizeze castigul lui MAX. Nodurile terminale au valorile a = 5, b= -10, c= 10, d = -2, e = 1, f = -1, g = 15, h =-1. Care sunt valorile Minimax ale nodurilor M, N, O, P, Q, R, S?

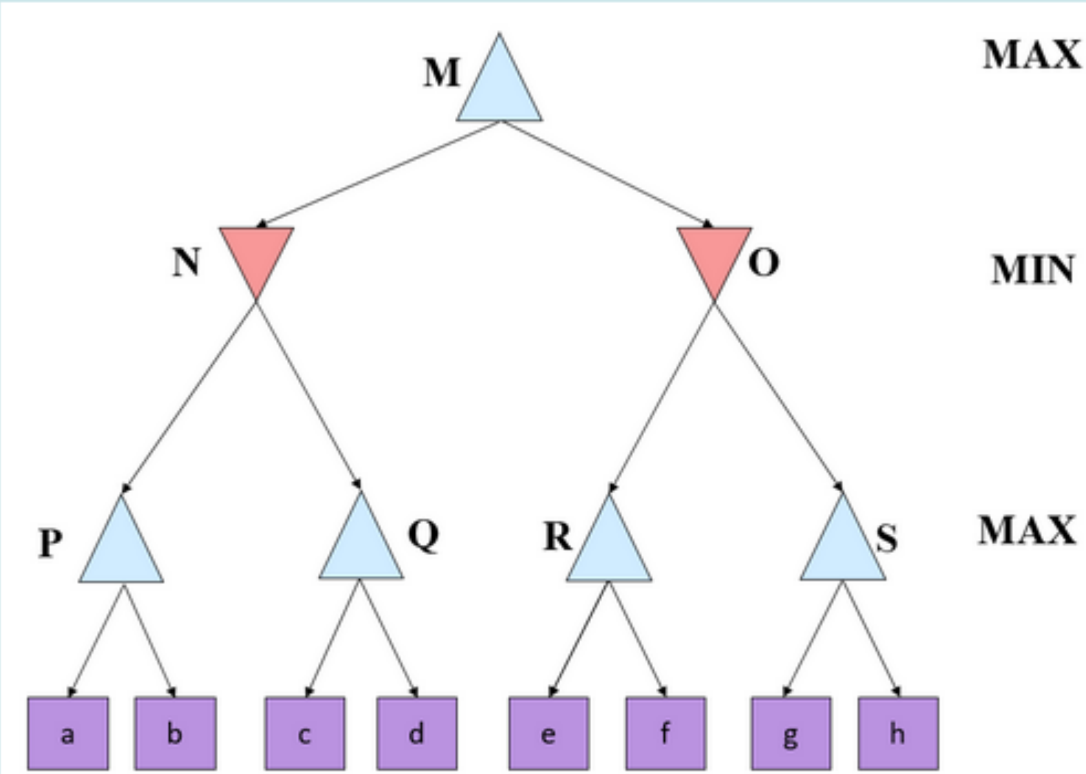


Consideram arborele de cautare din figura de mai jos, in care jucatorul la mutare este jucatorul MAX (vrea sa isi maximizeze castigul). Jucatorul MIN vrea sa minimizeze castigul lui MAX. Nodurile terminale au valorile a = 5, b= -10, c= 10, d = -2, e = 1, f = -1, g = 15, h =-1. Care sunt valorile Minimax ale nodurilor M, N, O, P, Q, R, S?

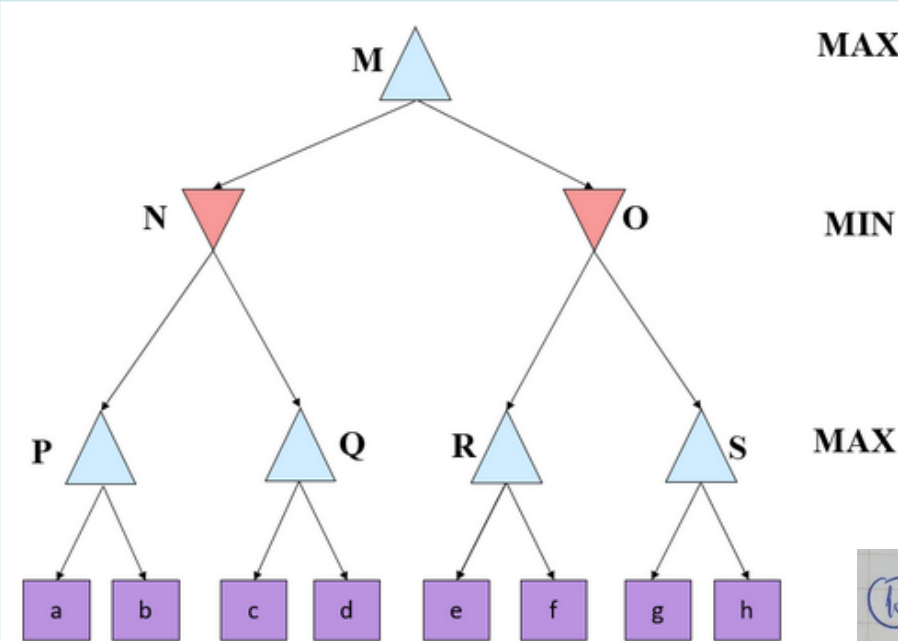


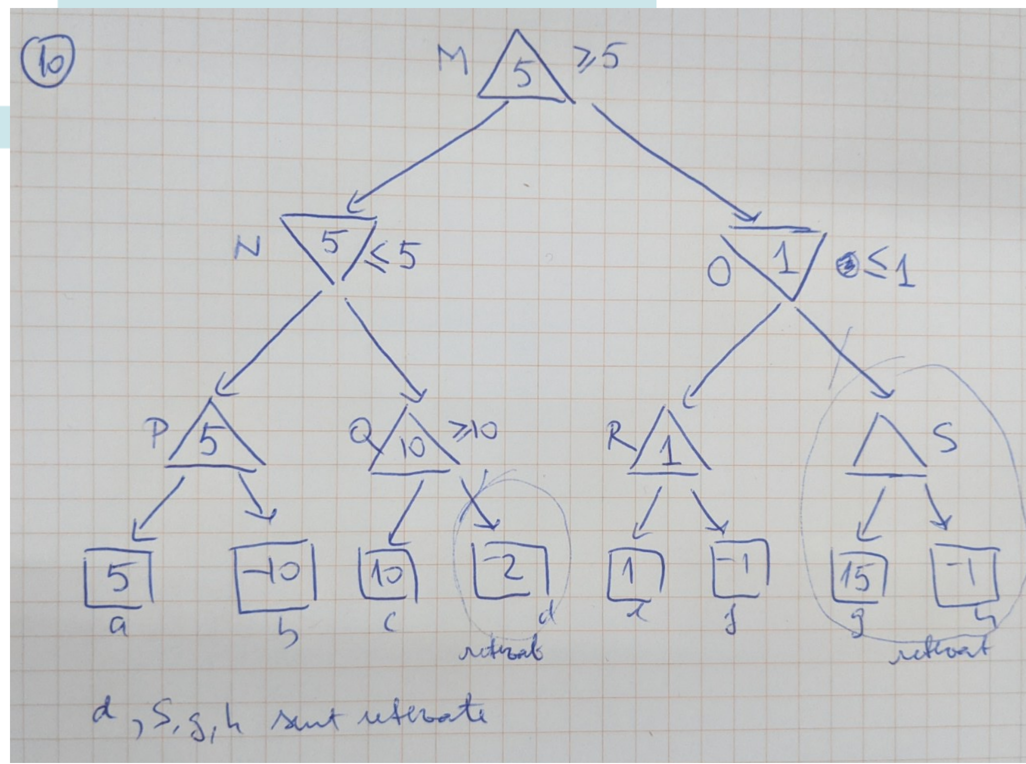


Consideram arborele de joc din figura de mai jos, in care jucatorul la mutare este jucatorul MAX (vrea sa isi maximizeze castigul). Jucat
Aplicati algoritmul alfa-beta in calcul valorii Minimax a radacinii M. Ce noduri sunt retezate (nu sunt vizitate)? Justificati raspunsul.



Consideram arborele de joc din figura de mai jos, in care jucatorul la mutare este jucatorul MAX (vrea sa isi maximizeze castigul). Jucat
Aplicati algoritmul alfa-beta in calcul valorii Minimax a radacinii M. Ce noduri sunt retezate (nu sunt vizitate)? Justificati raspunsul.





Cursuri opționale CTI anul 4

CTI, anul 4, semestrul 1	
DD1	Metode moderne de calcul si simulare
DD2	Prelucrarea semnalelor
DC1	Comunicare si relatii publice
DC2	Management
DS1	Cloud Computing
DS2	Tehnici Web
DS3	Scalarea retelelor de calculatoare
DS4	Calcul Numeric în Informatică
DS5	Concepte și aplicații în Vederea Artificială
DS6	Introducere in Robotica
DS7	Tehnici de optimizare
DS8	Robotic Process Automation (RPA) folosind platforma UiPath

CTI, anul 4, semestrul 2	
DD1	Sistemul de operare Linux
DD2	Algoritmi paraleli
DC1	Managementul operatiunilor
DC2	Marketing
DS1	Criptografie si securitate
DS2	Programarea pe dispozitive mobile
DS3	Conectarea retelelor la nivel WAN
DS4	Blockchain
DS5	Elemente de securitate si logica aplicata
DS6	Introducere in prelucrarea limbajului natural
DS7	Învățare automată în Vedere Artificială
DS8	Metode Formale in Ingineria Software
DS9	Production engineering
DS10	Strategii de planificare a unei echipe de roboti (SPER)
DS11	Inițiere în cercetare și bioinformatică
DS12	Algoritmica avansata a grafurilor

Cursuri opționale CTI anul 4

CTI, anul 4, semestrul 1	
DD1	Metode moderne de calcul si simulare
DD2	Prelucrarea semnalelor
DC1	Comunicare si relatii publice
DC2	Management
DS1	Cloud Computing
DS2	Tehnici Web
DS3	Scalarea retelelor de calculatoare
DS4	Calcul Numeric în Informatică
DS5	Concepte și aplicații în Vederea Artificială
DS6	Introducere in Robotica
DS7	Tehnici de optimizare
DS8	Robotic Process Automation (RPA) folosind platforma UiPath

CTI, anul 4, semestrul 2	
DD1	Sistemul de operare Linux
DD2	Algoritmi paraleli
DC1	Managementul operatiunilor
DC2	Marketing
DS1	Criptografie si securitate
DS2	Programarea pe dispozitive mobile
DS3	Conectarea retelelor la nivel WAN
DS4	Blockchain
DS5	Elemente de securitate si logica aplicata
DS6	Introducere in prelucrarea limbajului natural
DS7	Învățare automată în Vedere Artificială
DS8	Metode Formale in Ingineria Software
DS9	Production engineering
DS10	Strategii de planificare a unei echipe de roboti (SPER)
DS11	Inițiere în cercetare și bioinformatică
DS12	Algoritmica avansata a grafurilor

Prezentare curs opțional: Concepte și Aplicații în Vederea Artificială

Bogdan

Curs (50%)

Radu

Curs (50%)

Alexandra

Laborator (100%)

-cursul opțional se adresează studenților de anul 3 Mate-Info, anul 3 Info, anul 4 CTI

Ce este vederea artificială?

- Înzestrarea computerelor cu un sistem vizual asemănător cu sistemul vizual uman
- Scrierea de programe pentru calculator care pot interpreta imagini/video-uri

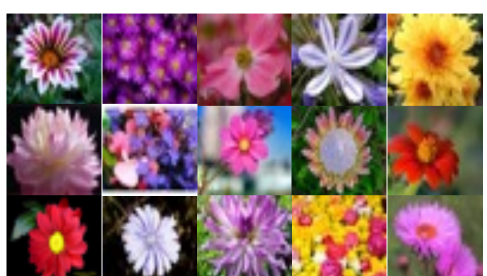
Structura cursului

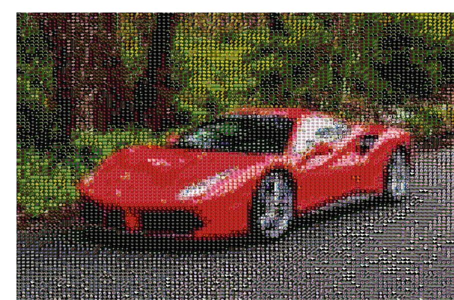
0. Formarea imaginilor

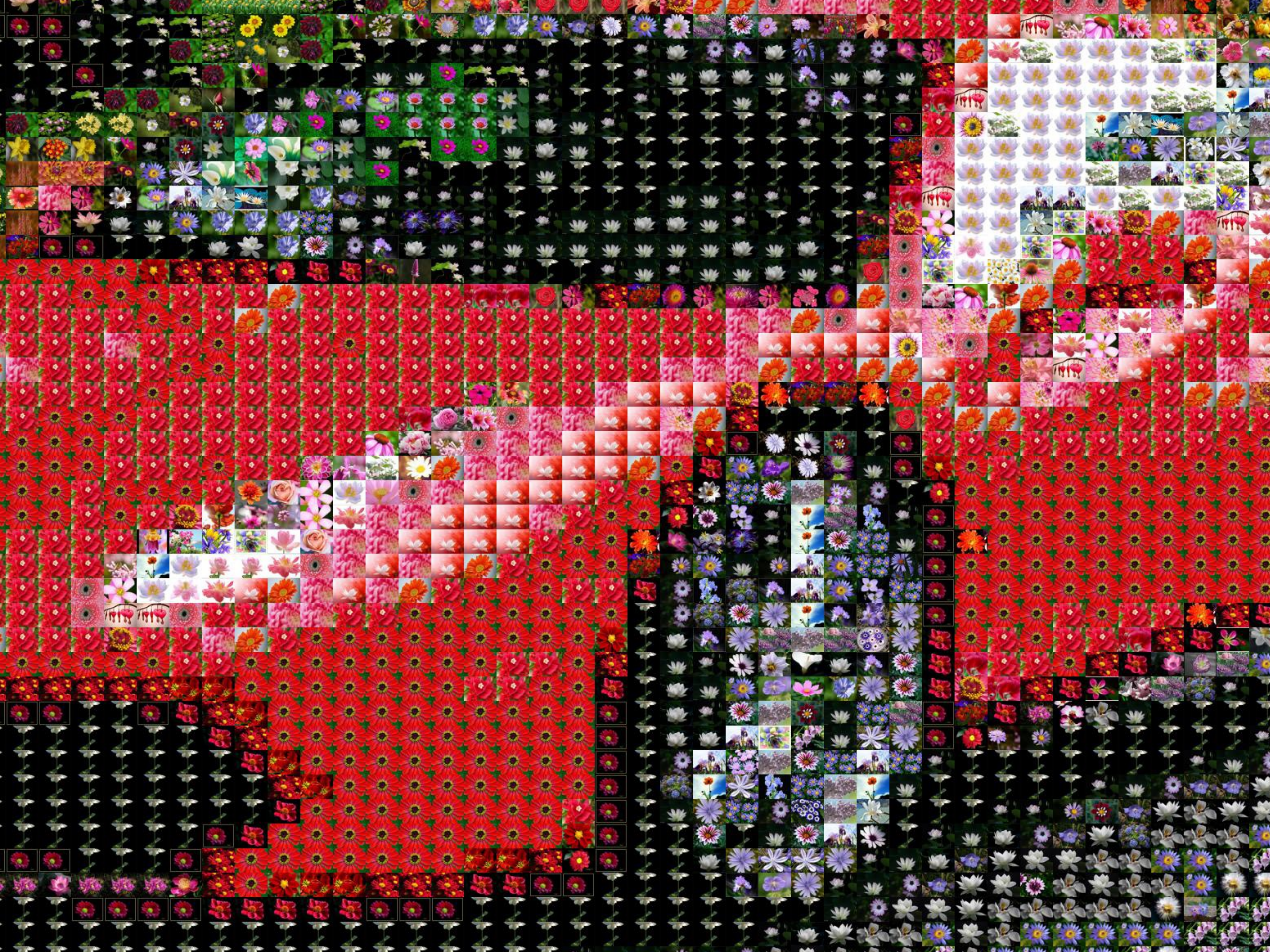
- generalități despre lucrul cu imagini

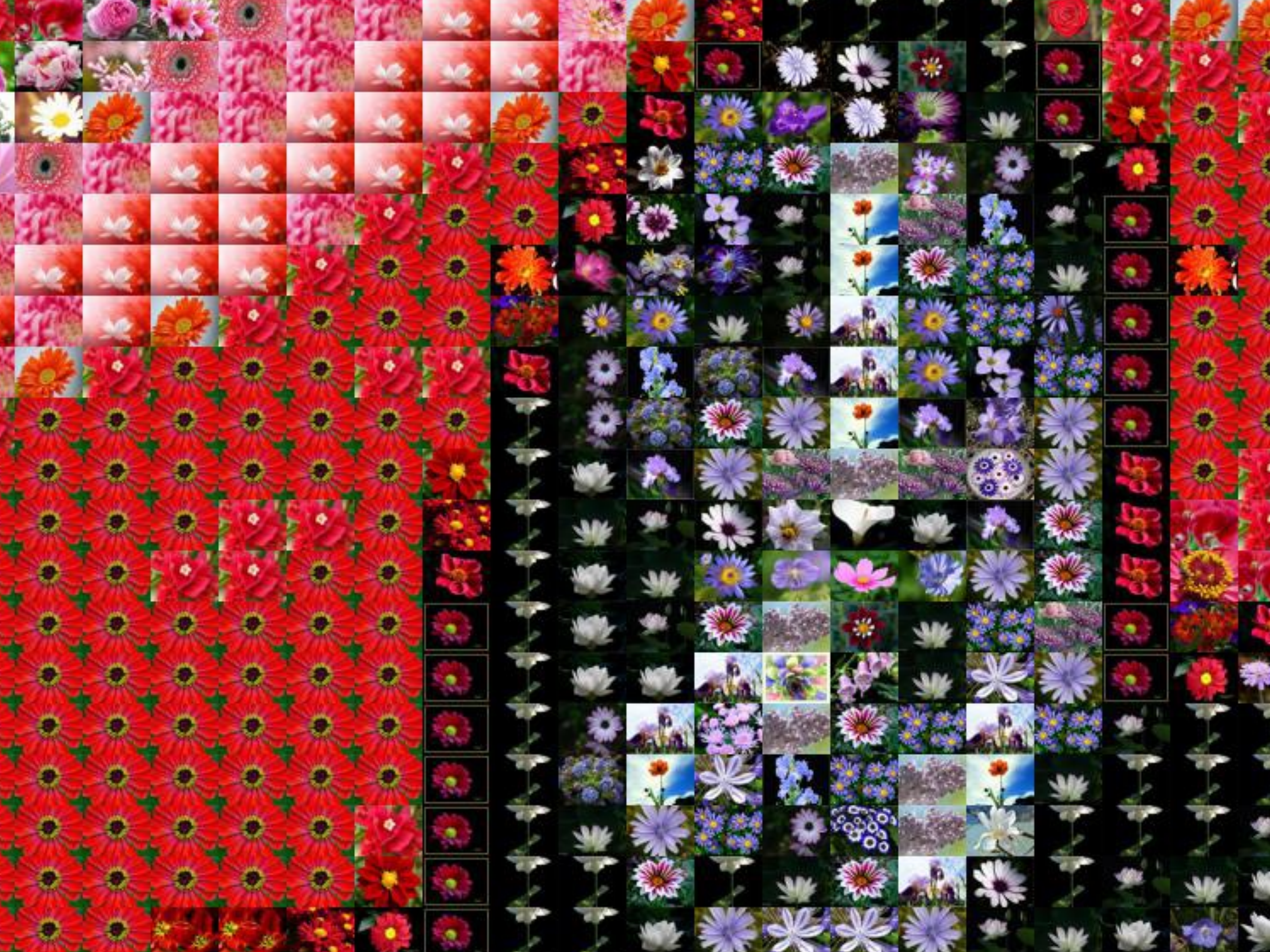
Laborator: realizarea de mozaicuri

imagine de referință + colecție de imagini (piese) de dimensiuni reduse + algoritm codat de voi = imagine mozaic

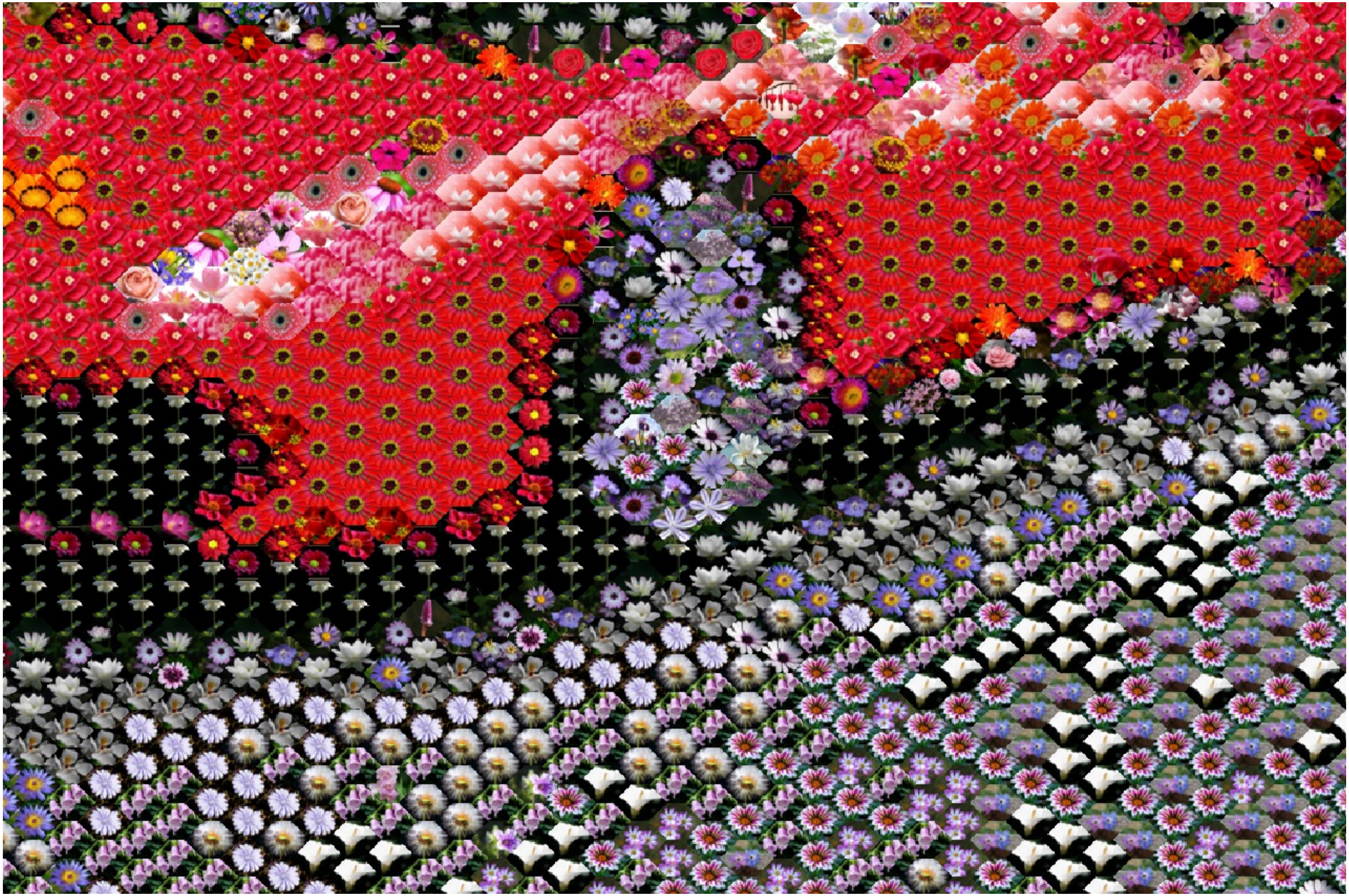








Piese hexagonale

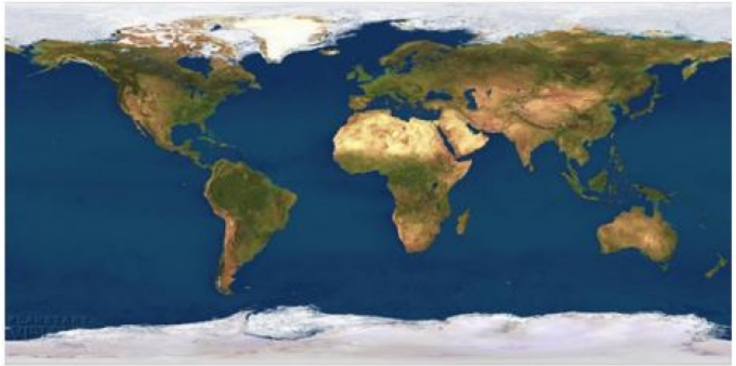


Piese hexagonale

Structura cursului

1. Caracteristici ale imaginilor
- filtre, gradienți, muchii, textură

Laborator: redimensionarea imaginilor cu păstrarea conținutului



Redimensionare uzuală (imresize)

Redimensionare cu păstrarea conținutului

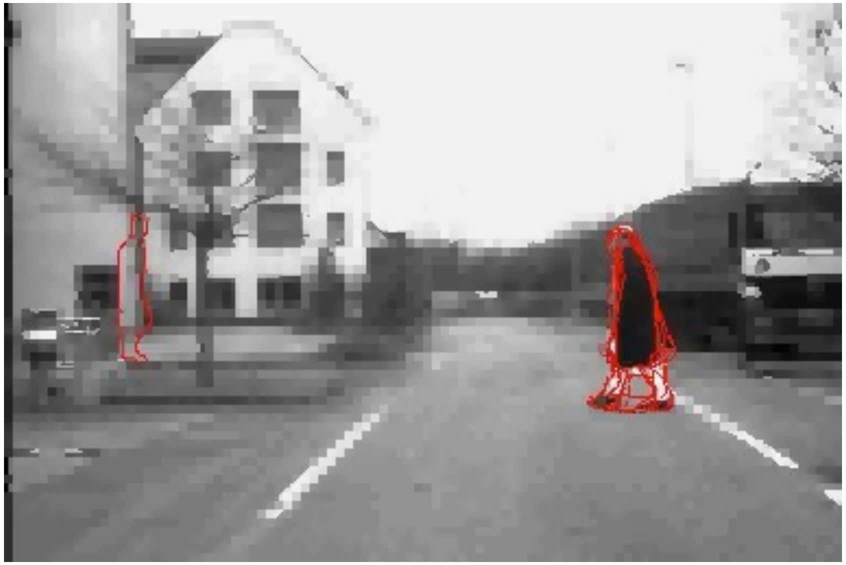
Structura cursului opțional

1. Caracteristici ale imaginilor

- filtre, gradienți, muchii, textură

Laborator: detectare de obiecte

Detectare semne de circulație



Detectare pietoni

Structura cursului opțional

2. Descriptori vizuali

- puncte de interes, descriptori SIFT, descriptori HOG

Laborator: detectare facială

Structura cursului opțional

3. Recunoaștere de obiecte

- modelul K-nearest neighbours, metode kernel, rețele neuronale convoluționale, modelul bag of visual words

Laborator: clasificarea imaginilor, colorarea imaginilor în tonuri de gri (grayscale)

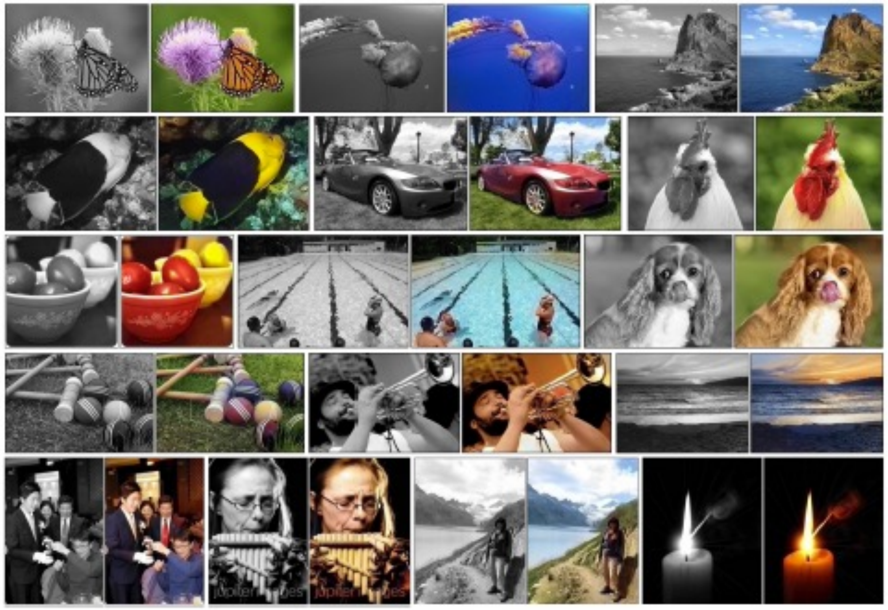
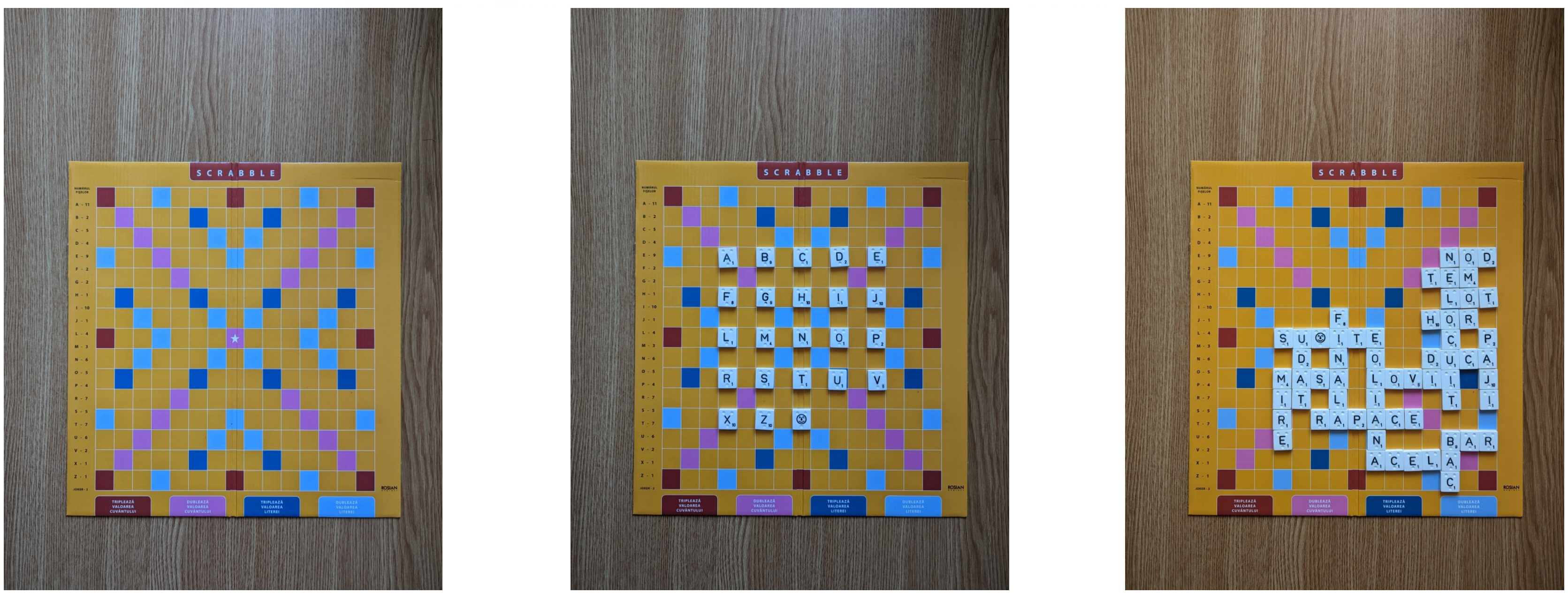


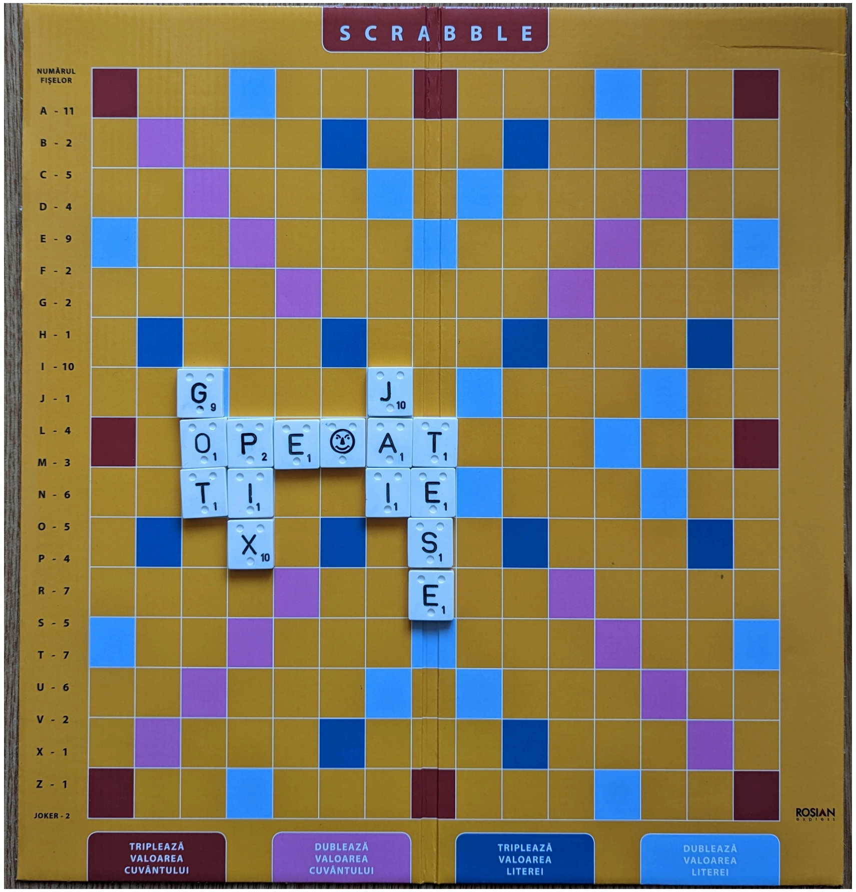
Figura 1: **Rezultatele colorării imaginilor în tonuri de gri.**

Tema 1 din acest an: Calculator automat de scor

Scrabble este un joc în care participanții formează cuvinte prin plasarea de litere pe o tablă având dimensiunile de 15×15 pătrate (Figura 1). Cuvintele se pot forma pe orizontală sau pe verticală, precum la cuvinte încrucișate, iar punctajul obținut este mai mare atunci când literele folosite sunt mai rare (mai valoroase) sau când sunt plasate pe pătrate divers colorate care dau diferite bonificații ale tablei de joc. Cuvintele sunt valide doar dacă corespund dicționarelor acceptate oficial. Fiecare jucător concurează pentru a obţine un scor cât mai mare folosind diferite combinaţii de litere plasându-le cât mai bine pe tablă astfel încât să profite de pătratele premium.



Exemple de formare a cuvintelor + calcul scor



COOPERAT: (1(C) + 1(O) + 1(O) + 2(P) + 1(E) + 0(?=R) + 1(A) + 1(T))*3(STC) = 24

TOTAL: 24

SDL = SCOR DUBLU LITERA
STL = SCOR TRIPLU LITERA

SDC = SCOR DUBLU CUVANT
STC = SCOR TRIPLU CUVANT

Tema 2 din acest an: Detectarea și recunoașterea facială a personajelor din serialul de desene animate **Viața cu Louie**

Viaţa cu Louie este un serial de desene animate lansat în anul 1994 şi care are 3 sezoane, a câte 13 episoade. Serialul prezintă întâmplări din viața lui Louie Anderson, un copil de 8 ani care trăiește în Cedar Knoll, Wisconsin (SUA). Multe episoade sunt inspirate din viața adevăratului Louie Anderson, comedian american care apare la începutul episoadelor și care le vorbește telespectatorilor, acesta regizând, de altfel, și toate episoadele.

Task 1 – detectare facială

Prima problemă pe care o aveţi de rezolvat constă în detectarea facială a *tuturor* feţelor personajelor care apar în imagini. Pentru fiecare imagine de intrare algoritmul vostru trebuie să returneze o mulţime de detecţii asociate (fereastră dreptunghiulară şi scor) ce localizează *toate* feţele dintr-o imagine. Figura 1 arată câteva exemple din mulţimea de antrenare şi adnotările corespunzătoare, ce constau în ferestre dreptunghiulare de culoare roşie ce încadrează perfect *fiecare faţă*.

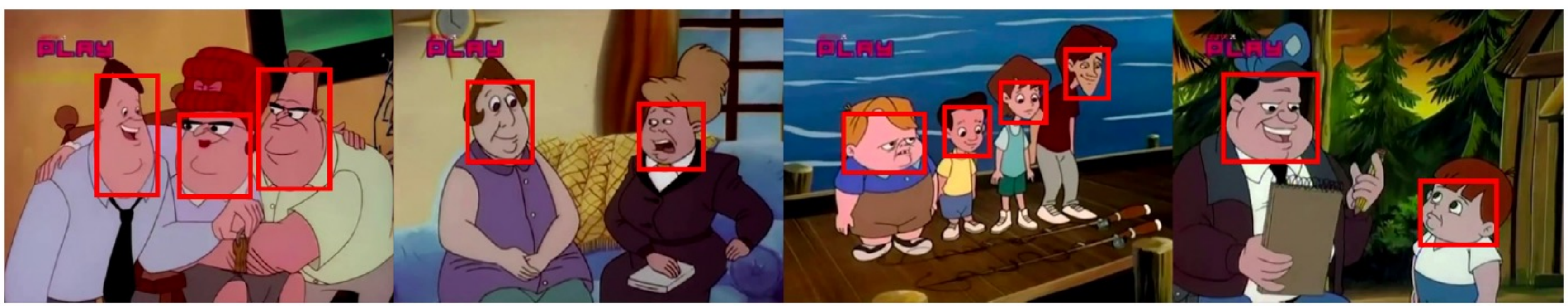
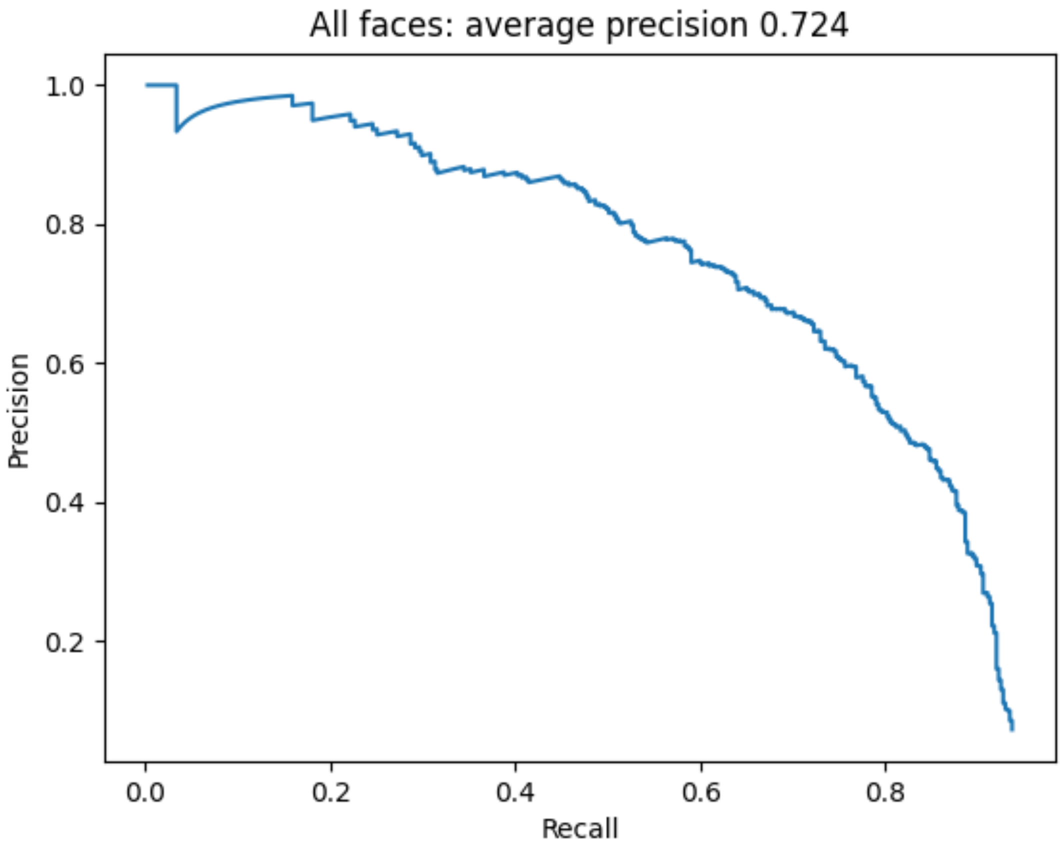


Figura 1: *Detectare facială a personajelor din Viaţa cu Louie: fiecare faţă de interes este adnotată cu o ferestră dreptunghiulară de culoare roşie.*

Evaluare task 1 – AP pe datele de test



(performanța soluției noastre pe datele de validare = 200 de imagini)

Task 2 – recunoaștere facială

A doua problemă pe care o aveţi de rezolvat constă în recunoaşterea facială a numai anumitor personaje. Alături de Louie, personajul principal din serial, apar cu preponderență alte trei personaje: Andy - tatăl său, Ora - mama sa, Tommy - fratele său. Vom considera problema recunoaşterii faciale numai pentru aceste patru personaje, Andy, Ora, Louie şi Tommy. Pentru fiecare imagine de intrare algoritmul vostru trebuie să returneze o mulţime de detecţii asociate (numele personajului, fereastră dreptunghiulară şi scor) ce localizează feţele celor patru personaje de interes (Andy, Ora, Louie, Tommy) din imagine. Figura 2 arată câteva exemple din mulţimea de antrenare şi adnotările corespunzătoare, ce constau în ferestre dreptunghiulare ce încadrează perfect *feţele de interes* (pentru Andy, Ora, Louie şi Tommy). Fiecare detecţie are o culoare specifică clasei personajului (albastru - Andy, galben - Ora, verde - Louie, violet - Tommy).

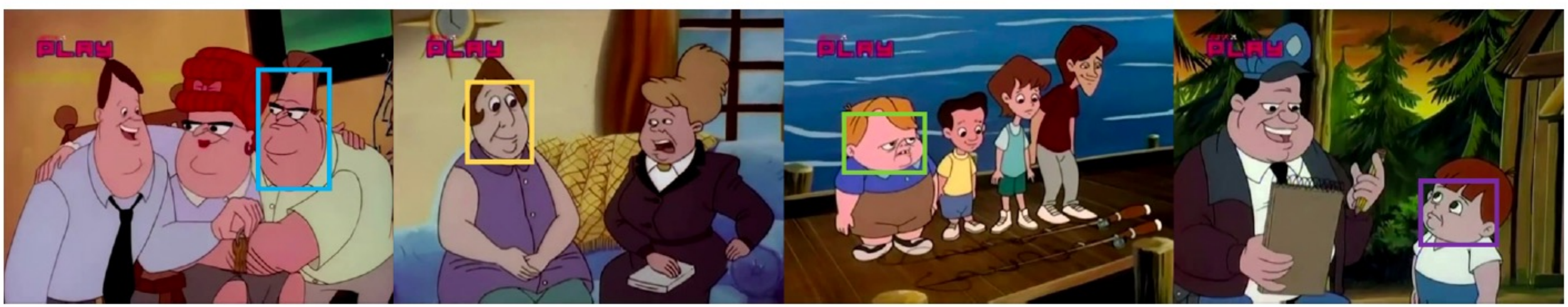
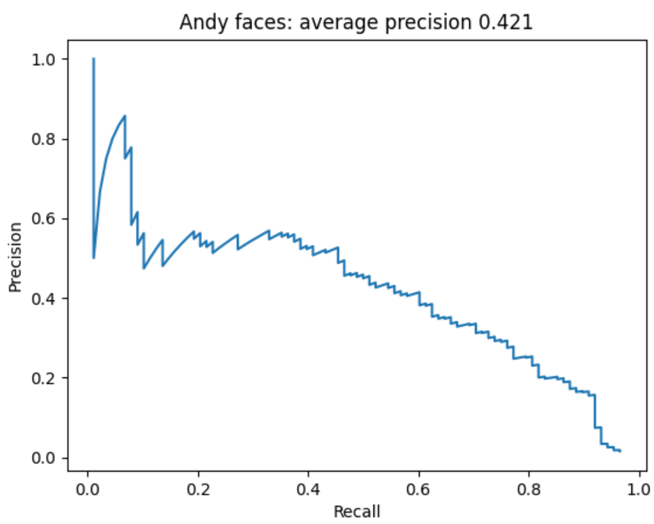
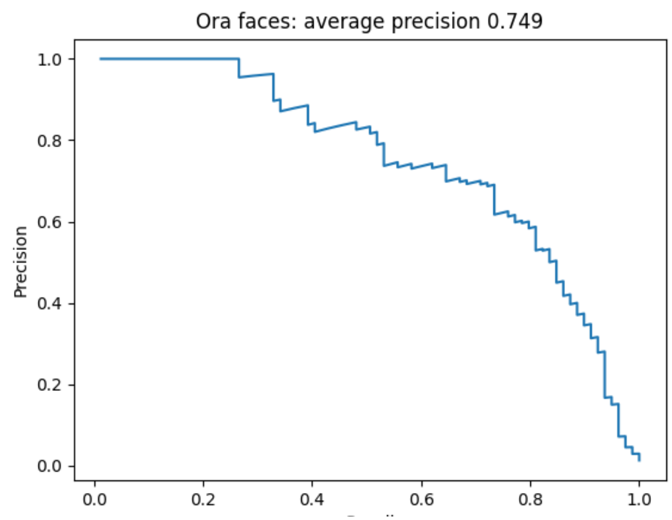
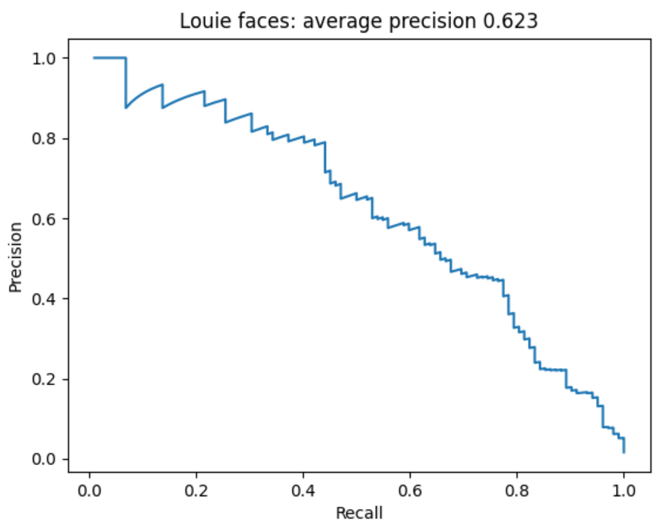


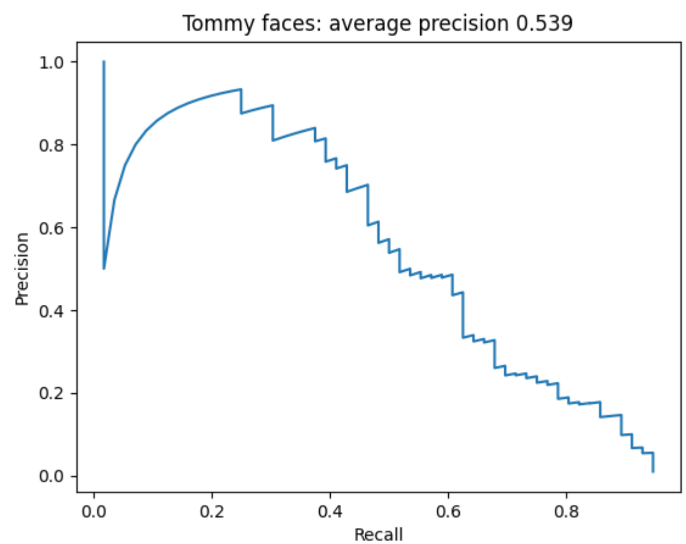
Figura 2: *Recunoaşterea facială a personajelor din Viaţa cu Louie: fiecare faţă de interes este adnotată cu o ferestră dreptunghiulară de culoare specifică clasei personajului (albastru - Andy, galben - Ora, verde - Louie, violet - Tommy).*

Evaluare task 2 – mAP pe datele de test









(performanța soluției noastre pe datele de validare = 200 de imagini)