

# Inteligență Artificială

## Lucrare de laborator – Varianta 4

### 26 iunie 2023

În această lucrare veți dezvolta și antrena modele pentru generare de imagini condiționate de imaginile de intrare. Pentru fiecare imagine de intrare corespunde o imagine de ieșire, care, pusă una sub alta, formează împreună o singură imagine cu conținut coerent.

În directorul **curent**, veți găsi datele de antrenare (trainImages.npy), etichetele corespunzătoare (trainTargets.npy), clasele imaginilor de antrenare (trainClasses.npy), datele de testare (testImages.npy) și clasele imaginilor de testare (testClasses.npy).

În fișierul trainImages.npy se găsește un tensor ce conține 1200 de imagini care reprezintă imaginile de intrare. În fișierul trainTargets.npy se găsește un tensor ce conține 1200 de imagini care reprezintă imaginile de ieșire.

În fișierul testImages.npy se găsește un tensor care conține 500 de imagini de testare.

#### **Rezolvați următoarele cerințe:**

1. **(2p)** Implementați o funcție care să returneze imaginea mediană ce trebuie prezisă pentru fiecare clasă de imagini în parte. Pentru a netezi fiecare imagine rezultată, aplicați operație de convoluție / filtrare pe fiecare imagine medie în parte, folosind un filtru de dimensiune 3x3 obținut prin înmulțirea vectorului coloană [0.2; 0.6; 0.2] cu transpusul său. În funcție de clasa asociată exemplelor de test, folosiți imaginea medie aferentă clasei drept predicție. Generați un fișier (citiți observațiile pentru denumirea corectă a fișierului) cu predicțiile pentru toate datele de test.

2. **(2p)** Implementați metoda celor mai apropiați vecini cu  $k \geq 3$  folosind distanța Minkowski cu  $p=3$ . Ca reprezentare pentru un exemplu, veți folosi concatenarea între un vector format din valorile pixelilor imaginii de intrare și un vector de tipul „one-hot” (care conține 1 pentru indexul clasei corespunzătoare și 0 în rest). Pentru a obține imaginea prezisă pentru fiecare exemplu de test, trebui să calculați imaginea mediană peste vecini și să neteziți imaginea cu același filtru ca la punctul 1. Pentru a obține punctajul acordat, trebuie să implementați corect modelul și să generați un fișier (citiți observațiile pentru denumirea corectă a fișierului) cu predicțiile pentru datele de test.

**1p** – MSE maxim pe datele de test = 1310.0

**2p** – MSE maxim pe datele de test = 1250.0

3. **(2p)** Antrenați o rețea neuronală feed-forward cu maxim 3 straturi pe mulțimea de antrenare, folosind aceeași reprezentare vectorială ca la punctul 2. Pentru a obține punctajul maxim, trebuie să găsiți parametrii optimi pentru modelul dat și să generați maxim 3 submisii / fișiere cu predicțiile pe datele de test (citiți observațiile pentru denumirea corectă a fișierelor). În lipsa imaginilor de ieșire de test, puteți păstra o parte din mulțimea de antrenare pentru validare.

**1p** – MSE maxim pe datele de test = 1300.00

**2p** – MSE maxim pe datele de test = 1140.00

4. **(2p)** Antrenați un ansamblu de 5 rețele neuronale cu maxim 3 straturi pe mulțimea de antrenare, folosind aceeași reprezentare vectorială ca la punctul 2. Pentru predicția finală a modelelor din ansamblu, trebui să calculați imaginea mediană peste imaginile returnate de toate rețelele.

**1p** – MAE maxim pe datele de test = 1100.0

**2p** – MAE maxim pe datele de test = 1050.0

6. **(1p)** Creați un raport al experimentelor însoțit de evaluarea pe un set de validare a diferite combinații de hiperparametri, atât pentru modelul de la punctul 2 cât și pentru cel de la punctul 3. Raportul poate conține tabele sau grafice.

**1p - Oficiu**

***Observații importante:***

După implementarea cerințelor de mai sus, trebuie să trimiteți într-un folder denumit {Nume}\_{Prenume}\_{Grupa}\_{Varianta}:

a) Cel mult 1 submisie pentru setul de testare cu metodele de la punctele 1 și 2; cel mult 3 submisii pentru setul de testare cu fiecare din metodele de la punctele 3 și 4. O submisie constă într-un fișier .npz denumit:

{Nume}\_{Prenume}\_{Grupa}\_subiect{i}\_solutia\_{j}.npz

unde i este numărul subiectului (1, 2, 3 sau 4) și j este numărul submisiei (1, 2 sau 3), în care se află un tensor ce conține imaginile de ieșire pentru toate exemplele de test.

b) Codul aferent pentru antrenarea modelelor și obținerea soluțiilor trimise. Pentru fiecare submisie, codul trebuie organizat într-un singur fișier .py denumit:

{Nume}\_{Prenume}\_{Grupa}\_subiect{i}\_solutia\_{j}.py

unde i este numărul subiectului (1, 2, 3 sau 4) și j este numărul submisiei (1, 2 sau 3).

c) Raportul de la punctul 5.