Tema: Problema de optimizare neconstrânsă în învățarea aprofundată

Deadline: 27.04.2025
Punctaj: 10p
Scop: Familarizarea studentul cu noțiuni din domeniul învățării aprofundate, precum și cu rolul optimizării din acest domeniu la modă.

1 Noţiuni introductive în învățarea aprofundată

În cele ce urmează vom introduce noțiuni de bază din domeniul învățării profunde (*engl. deep learning*). Metodele de învățare din acest domeniu se bazează pe rețele neuronale artificiale (RNA). Aceste metode s-au inspirat din biologie și au la bază modelul neuronului artifical (a se vedea figura de mai jos):

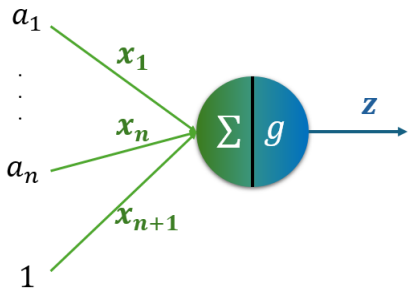

Figure 1: Modelul neuronului artificial: $a \in \mathbb{R}^n$ este vectorul de intrare în neuron, $x \in \mathbb{R}^{n+1}$ este vectorul de ponderi asociate neuronului, g este funcția de activare și z este ieșirea din neuron

Modelul neuronului se poate împărți în două componente:

- **componenta liniară**: $\bar{z} = \sum_{j=1}^{n} a_j x_j + x_{n+1}$
- **componenta neliniară**: $z = g(\bar{z})$, unde g este de obicei o funcție neliniară (exemple de funcții de activare se găsesc în secțiunea 4).

Așezarea mai multor neuroni pe același nivel va crea un strat de neuroni. O rețea neuronală artificială are cel puțin 3 straturi:

- un strat de intrare - nr. de neuroni este egal cu numărul de caracteristici ale intrării, plus unul pentru deplasare (*engl. bias*). În general neuroni sunt simpli, doar de trecere a informației.
- un strat intermediar - nr. de straturi intermediare determină *adâncimea* rețelei. O rețea cu un strat intermediar se numește o rețea superficială (*engl. shallow network*).

- un strat de ieșire - nr. de neuroni din acest strat este determinat de setul de date.

În cele ce urmează vom folosi o rețea superficială, cum este reprezentată în Figura 2.

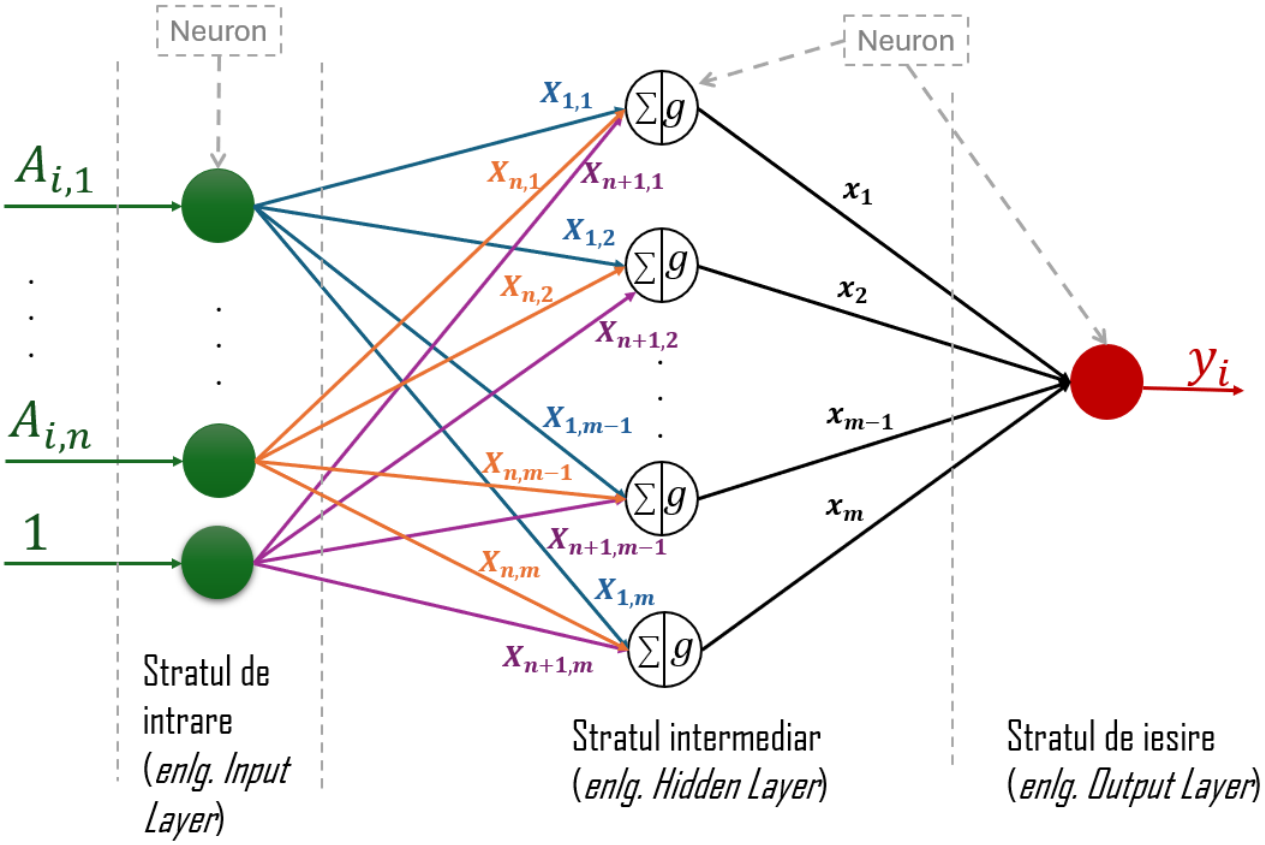

Figure 2: Exemplu de rețea superficială: n intrări, m neuroni pe stratul ascuns şi o ieşire.

Modelul de predicţie ce rezultă din reţeaua neurală din Figura 2 este de forma:

$$\mathcal{M}_{x,X}(a) = g(a^T X)x, \quad \text{cu } a \text{ un exemplu din baza de date.}$$

Pentru ca sarcina de învățare să aibă loc, la modelul rețelei se adaugă o funcție de pierderi (*engl. loss function*), ce trebuie minimizată. Există diferite funcții de pierderi, alese în conformitate cu tipul de învățare: supervizat (avem acces la etichete) și nesupervizat (fară acces la etichete). Există și o clasă de metode semi-supervizate, când setul de date este împărțit într-o mulțime mică etichetată și una mare netichetată. Mai departe oferim 3 exemple de funcții de pierdere uzuale pentru metodele supervizate, clasa metodelor pe care o vom aborda mai departe:

- Eroare medie pătrată (utilizată în *probleme de regresie*): $L(e, y) = \frac{1}{2N}\|e - y\|^2 = \frac{1}{2N}\sum_{i=1}^{N}(e_i - y_i)^2$
- Eroarea medie absolută (utilizată în *probleme de regresie*): $L(e, y) = \frac{1}{N}\sum_{i=1}^{N}|e_i - y_i|$
- Entropie încrucișată binară (utilizată în probleme de *clasificare binară*) :
  $L(e, y) = -\frac{1}{N}\sum_{i=1}^{N} e_i \log y_i + (1 - e_i)\log(1 - y_i)$,

unde $e \in \mathbb{R}^N$ este vectorul de etichete (cunoscut din setul de date) și $y \in \mathbb{R}^N$ este vectorul de predicții (ieșirea din rețea), i.e., $y_i = \mathcal{M}_{x,X}(a_i) = g(a_i^T X)x$ pentru $i = 1 : N$. *Observaţi că prima şi a treia funcţie de pierdere sunt diferenţiabile.* În secțiunea următoare, formulăm problema de optimizare fără constrângeri.

2 Formularea problemei

Considerăm următoarele notații:

- $A \in \mathbb{R}^{N \times n}$ baza de date, unde N este nr. de exemple și n este nr. de caracteristici ale unui exemplu
- $e \in \mathbb{R}^N$ vectorul referință (etichete).
- $\mathbb{1}_N \in \mathbb{R}^N$ vector plin cu unu.

Scopul este gasirea parametrilor ce descriu rețeaua neuronală cu un singur strat din Figura 2, i.e. găsirea parametrilor optimi $x \in \mathbb{R}^m$ şi $X \in \mathbb{R}^{(n+1)\times m}$, unde m este numărul de neuroni de pe stratul ascuns. Gasirea parametrilor optimi se realizează rezolvând problema de minimizare fără constrângeri:

$$\min_{x \in \mathbb{R}^m, X \in \mathbb{R}^{(n+1)\times m}} L(e, g\left(\bar{A}X\right)x), \tag{1}$$

unde $\bar{A} = [A, \mathbb{1}_N]$ şi L este una din funcţiiile de pierdere de mai sus. De exemplu, selectând eroarea medie pătrată ca funcție de pierderi, obținem următoarea problemă de optimizare:

$$\min_{x \in \mathbb{R}^m, X \in \mathbb{R}^{(n+1)\times m}} \frac{1}{2N} \|g\left(\bar{A}X\right)x - e\|^2 \tag{2}$$

3 Cerinţe

- Selectați o baza de date adecvată învățării supervizate (i.e. problema clasificării sau regresiei). Exemple de site-uri cu baze de date: UC Irvine Machine Learning Repository, Kaggle datasets, etc. Baza de date ar trebui să aibă cel puțin 100 de exemple (din care selectati 80% pentru antrenare si 20% pentru testare).
- Stabiliți un număr de neuroni $m \geq 10$ pentru stratul ascuns şi folosiți funcția de activare din sectiunea 4, conform tabelului de pe moodle.
- Rezolvați problema de optimizare (1), implementând cel puțin doua metode de optimizare de la curs. Exemple de metode: Metoda gradient, Metoda gradient stocastica, Metoda Newton, Metoda quasi-Newton, Metoda Gauss-Newton, Metoda Levenberg-Marquardt, Metoda de Optimizare Alternativa.
- Comparați metodele alese în termeni de timp și iterații analizând norma gradientului, funcția obiectiv dea lungul iteratiilor si timp.
- Analizați performanța sarcinii de învățare aleasă (clasificare sau regresie) folosind metrici potrivite cum ar fi precizia clasificatorului (matrice de confuzie), coeficientul de determinare R_2, etc.

3.1 Livrabile

Studenţii vor pregăti şi încărca pe moodle următoarele fişiere:

- Un fişier PDF, denumit *grupa_nume_prenume.pdf*, în care sunt menţionate: sarcina de învăţare (clasificare sau regresie), detalii despre baza de date utilizată, detalii despre algoritmii de optimizare implementaţi, rezultate şi comentarii. Într-o anexă se va adăuga codul matlab.
- Fişierele matlab.

3.2 Indicatori de performanţă

Într-o problemă de clasificare, performanţa modelului este evaluată cu ajutorul matricei de confuzie (comanda matlab *confusionmat*). Pe baza acestei matrice se calculează indicatorii de performanţă după cum se poate observa in Figura 3.

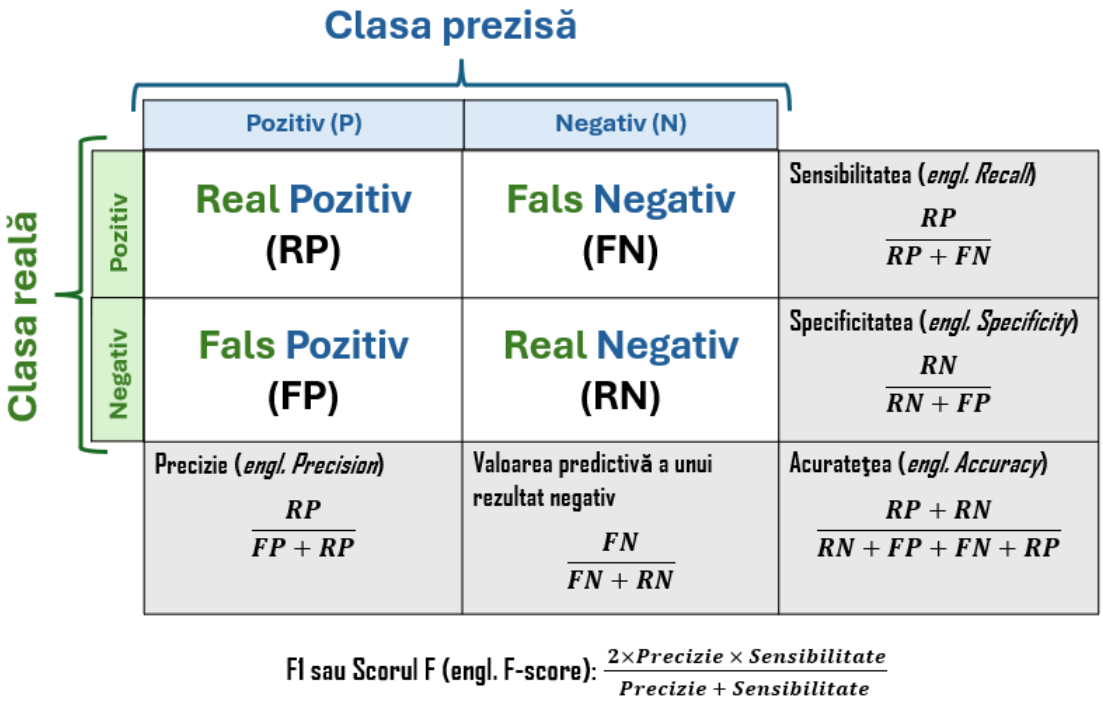

Figure 3: Matricea de confuzie şi indicatorii de performanţă asociaţi: sensibilitatea, acurateţea, ş.a.m.d.

Observaţii:

- **Scorul F (F1)** este un alt indicator des utilizat în exemplificarea performanţei clasificatorului. Un scor F ridicat indică, în general, o performanță bine echilibrată, demonstrând că modelul poate atinge în același timp o precizie ridicată și o sensibilitate mare.
- Când clasele nu sunt bine proporţionate, adică într-o clasă există cu mult mai multe exemple decât în cealaltă, performanţa clasificatorului este mai bine reprezentată de indicatorul **precizie** decât de **acurateţe**.

Pentru probleme de regresie avem următorii indicatori de performanţă:

- **Scorul** R^2 sau coeficientul de determinare indică proporția variației din variabila dependentă (cea pe care încercăm să o prezicem, e) care este explicată de modelul de regresie, în comparație cu variația totală a variabilei dependente.
$$R^2 = 1 - \frac{\sum_{i=1}^{N}(e_i - y_i)^2}{\sum_{i=1}^{N}(e_i - \mu_e)^2},$$
unde $\mu_e = \frac{1}{N}\sum_{i=1}^{N} e_i$ este media. Interpretare: $R^2 = 1$ indică faptul că modelul explică întreaga variație a datelor, ceea ce înseamnă că se potrivește perfect; $R^2 \in (0, 1)$ indică proporția variației din date care este explicată de model; iar $R^2 = 0$ indică faptul că modelul nu explică deloc variația datelor, ceea ce înseamnă că nu se potrivește deloc.
- Eroarea medie absolută: $MAE = \frac{1}{N}\sum_{i=1}^{N} |e_i - y_i|$.
- Eroarea medie pătratică: $MSE = \frac{1}{N}\sum_{i=1}^{N}(e_i - y_i)^2$

4 Funcții de activare

Lista de funcţii de activare a fost preluată din [1]. Funcţiile ce au parametrii se aleg de către studenţi. Atenţie! A se evita cazul în care se recuperează funcţia originală. De exemplu pentru funcția de activare nr. 2 se vor evita valorile $a = 1$ şi $b = 0$.

0. **Funcția sigmoid:** $\sigma(z) = \frac{1}{1+e^{-z}}$
1. Functia sigmoid cu deplasare si scalare: $g(z) = \frac{1}{1+e^{-a(z-b)}}$, unde $a, b \in \mathbb{R}$ fixaţi.
2. Funcția VFS: $g(z) = a\sigma(bz) - c = \frac{a}{1+e^{-bz}} - c$, unde $a, b, c \in \mathbb{R}$ fixaţi.
3. Funcția **tangenta hiperbolica (tanh)**: $g(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} = 2\sigma(2z) - 1$
4. Funcția tanh scalata: $g(z) = a\tanh(bz)$, unde $a, b \in \mathbb{R}$ fixaţi.
5. Funcția SigLin: $g(z) = \sigma(z) - az$, unde $a \in \mathbb{R}$ fixat.
6. Funcția SiLU: $g(z) = z\sigma(z)$
7. Funcția TS-sigmoid: $g(z) = \frac{1}{1+e^{-z}}\left(\frac{1}{1+e^{-z}} + \frac{1}{1+e^{-z+a}} + \frac{1}{1+e^{-z+b}}\right)$, unde $a, b \in \mathbb{R}$ fixaţi.
8. Funcția soft-clipping (SC): $g(z) = \frac{1}{a}\ln\left(\frac{1+e^{az}}{1+e^{a(z-1)}}\right)$, unde $a \in \mathbb{R}$ fixat.
9. Funcția **secanta hiperbolică**-sigmoid (**Sech**Sig): $g(z) = (z + a\ \mathrm{sech}(z + a))\sigma(z)$, unde $a \in \mathbb{R}$ fixat şi $\mathrm{sech}(z) = \frac{2e^z}{e^{2z}+1}$.
10. Funcția TanhSig: $g(z) = (z + a\tanh(z + a))\sigma(z)$, unde $a \in \mathbb{R}$ fixat.
11. Funcția Rootsig: $g(z) = \frac{az}{1+\sqrt{1+a^2z^2}}$, unde $a \in \mathbb{R}$ fixat.
12. Funcția exponenţială swish: $g(z) = e^{-z}\sigma(z)$

13. Funcția sigmoid derivată: $g(z) = e^{-z}(\sigma(z))^2$
14. Funcția Gish: $g(z) = z \ln\left(2 - e^{(-e^z)}\right)$
15. Funcția Logish: $g(z) = z \ln(1 + \sigma(z))$
16. Funcția LiSHT: $g(z) = z \tanh(z)$
17. Funcția TSReLU: $g(z) = z \tanh(\sigma(z))$
18. Funcția TBSReLU: $g(z) = z \tanh\left(\frac{1-e^{-z}}{1+e^{-z}}\right)$
19. Funcția Log-Sigmoid: $g(z) = ln(\sigma(z))$
20. Funcția dSiLU: $g(z) = \sigma(z)(1 + z(1 - \sigma(z)))$
21. Funcția MSiLU: $g(z) = z\sigma(z) + \frac{e^{-z^2-1}}{4}$
22. Funcția secanta hiperbolica rectificată: $g(z) = z \operatorname{sech}(z)$
23. Funcția Mish: $g(z) = z \tanh\left(\ln\left(1 + e^z\right)\right)$
24. Funcția TanhExp: $g(z) = z \tanh(e^z)$
25. Funcția SinSig: $g(z) = z \sin(\frac{\pi}{2}\sigma(z))$
26. Funcția Softplus: $g(z) = \ln(e^z + 1)$
27. Funcția SoftPlus parametrizat: $g(z) = a(\ln(e^z + 1) - b)$, unde $a, b \in \mathbb{R}$ fixaţi
28. Funcția Soft++: $g(z) = \ln(1 + e^{az}) + \frac{z}{b} - \ln(2)$, unde $a, b \in \mathbb{R}$ fixaţi
29. Funcția Aranda-Ordaz: $g(z) = 1 - (1 + ae^z)^{\frac{1}{a}}$, unde $a > 0$ fixat
30. Funcția sine: $g(z) = \sin(\pi z)$
31. Funcția cosine: $g(z) = 1 - \cos(z)$
32. Funcția cosid: $g(z) = \cos(z) - z$
33. Funcția Sinp: $g(z) = \sin(z) - az$, unde $a \in \mathbb{R}$ fixat
34. Funcția GCU: $g(z) = z \cos(z)$
35. Funcția ASU: $g(z) = z \sin(z)$
36. Funcția polyexp: $g(z) = (az^2 + bz + c)e^{-dz^2}$, $a, b, c, d \in \mathbb{R}$ fixaţi
37. Funcția exponenţială: $g(z) = e^{-z}$
38. Funcția E-Tanh: $g(z) = ae^z \tanh(z)$
39. Funcția undă (wave): $g(z) = (1 - z^2)e^{-az^2}$

40. Funcția NCU: $g(z) = z - z^3$
41. Funcția polinom de ordinul 3: $g(z) = az^3$, unde $a \in \mathbb{R}$ fixat
42. Funcția SQU: $g(z) = z^2 + z$
43. Funcția SoftSign: $g(z) = \frac{z}{1+|z|}$

References

[1] Kunc, V., & Kléma, J., *Three Decades of Activations: A Comprehensive Survey of 400 Activation Functions for Neural Networks*. arXiv preprint arXiv:2402.09092, 2024.