

# **Invatare Automata - Tema 3**

## **Recunoastere faciala**

Teodor-Stefan Dutu

Universitatea Politehnica Bucuresti  
Facultatea de Automatica si Calculatoare  
Grupa 341C3

# Cuprins

1	Introducere .....	3
2	Cerinta 1 - MLP-ul .....	4
3	Cerinta 2 - CNN-ul antrenat pe intreg setul de date .....	5
4	Cerinta 3 - CNN-ul antrenat pe setul de date redus .....	6
5	Cerinta 4 - MLP-ul imbunatatit .....	7
6	Bonus - VGG + SVC .....	9

## 1 Introducere

Scopul temei este de a compara performantele unor *MLP-uri* cu cele ale unui *CNN* folosind setul de date *Labeled Faces in the Wild* [1], atat in varianta completa, cat si intr-una redusa, care contine doar cele mai numeroase 5 clase.

## 2 Cerinta 1 - MLP-ul

Modelul initial pe care l-am ales este prezentat in Tabelul 1. Optimizatorul este *Adam*, deoarece acesta antreneaza retelele in batch-uri, facand antrenarea mai rapida pe seturi de date mari, cum este *LFW*. Activarile sunt de tip *ReLU*, ca sa evit problema *vanishing gradients*.

Tip	Dimensiune
Fully connected	8742 x 128
ReLU	-
Fully connected	128 x 128
ReLU	-
Fully connected	128 x 5
ReLU	-
SoftMax	-

Table 1: Modelul *MLP* de baza

Acest model a fost antrenat si testat pe setul redus de date, iar performantele sale, pe seturile de testare si antrenare se pot vedea in Figurile 1 si 2. Acuratetea pe seturile de testare si antrenare este slaba din cauza gradului mare de varianta a datelor din setul de antrenare.

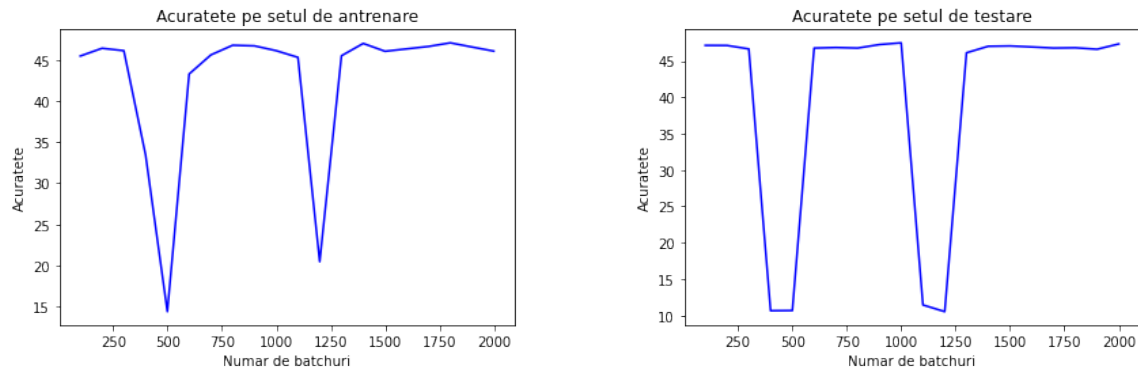
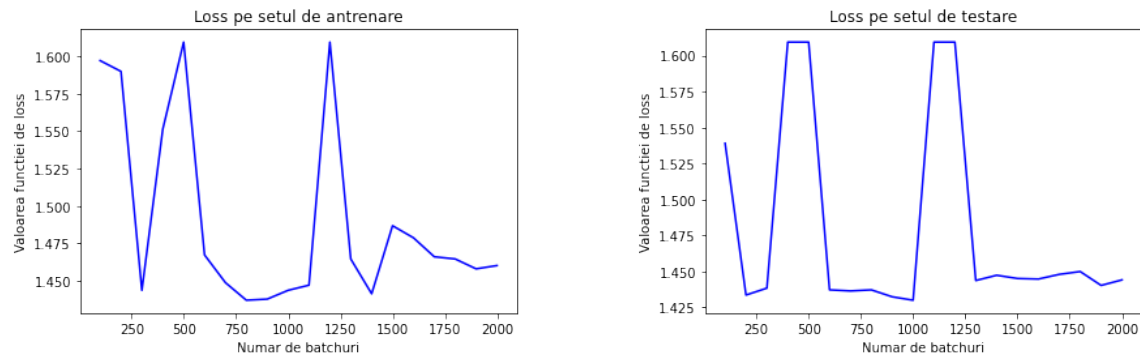
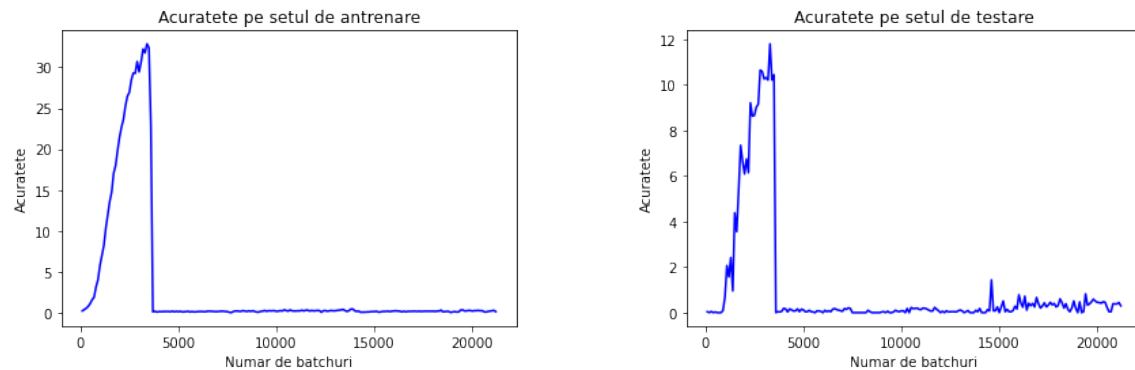


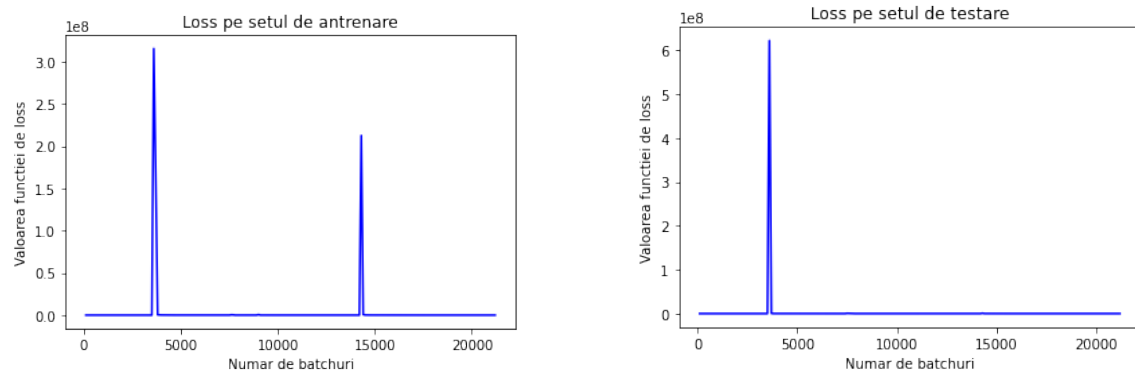
Fig. 1: Acuratetele modelului *MLP* initial

Fig. 2: Lossurile modelului *MLP* initial

### 3 Cerinta 2 - CNN-ul antrenat pe intreg setul de date

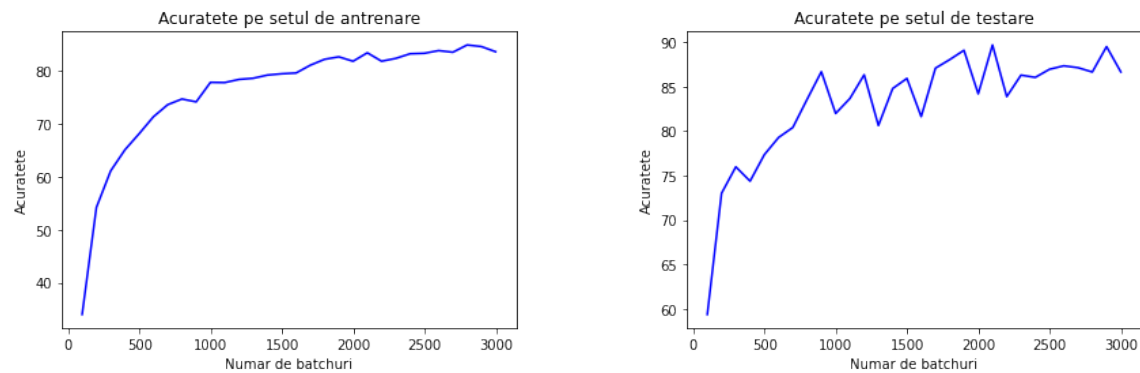
In acest caz se observa cum acuratetea modelului creste initial, dupa care scade foarte mult. Acest lucru este cauzat de faptul ca fiecare clasa are un numar foarte mic de exemple de antrenament (procentual) si cand ajunge sa invete mai multe, nu mai poate distinge intre ele. In acest sens, acuratetile atat pe seturile de testare cat si pe cel de antrenare tind la 0, dupa cum se vede in Figurile 3 si 4, modelul fiind probabil prea simplu pentru a distinge intre atatea clase si atatea exemple.

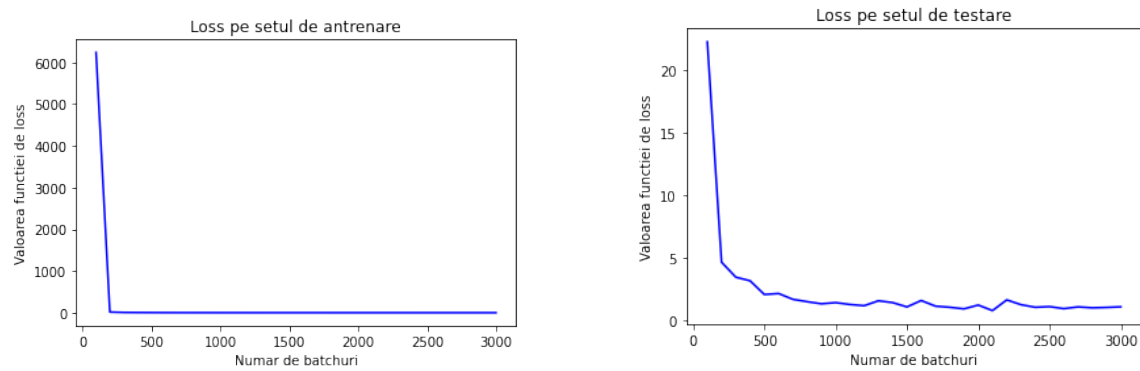
Fig. 3: Acuratetile modelului *CNN* pe seturile complete de testare si antrenare

Fig. 4: Lossurile modelului *CNN* pe seturile complete de testare si antrenare

#### 4 Cerinta 3 - CNN-ul antrenat pe setul de date redus

In schimb, cand am antrenat reseaua convolutionala pe setul de date cu 5 clase, acuratetea a ajuns aproape de 90% (antrenand timp de 300 de epoci), datorita ponderii crescute a fiecărei clase in setul de date. Figurile 5 si 6 indica o antrenare cu un progres relativ constant.

Fig. 5: Acuratetele modelului *CNN* pe seturile reduse de testare si antrenare

Fig. 6: Lossurile modelului *CNN* pe seturile reduse de testare si antrenare

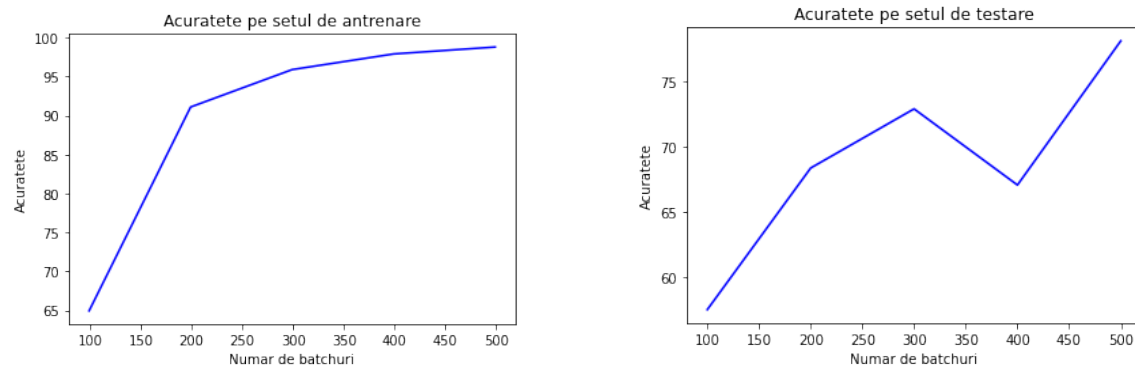
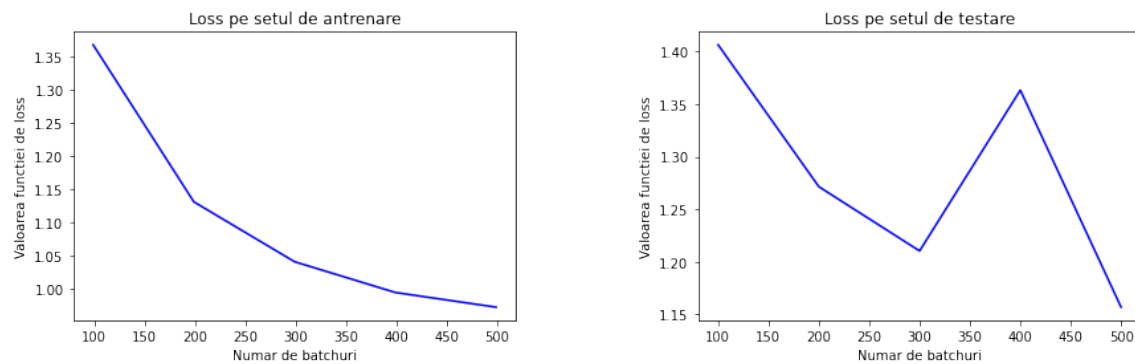
## 5 Cerinta 4 - MLP-ul imbunatatit

Initial observasem ca MLP-ul parea prea complex ca sa invete. Asadar, am incercat sa reduc numarul de straturi, iar acuratetea la antrenare ajungea rapid la 99%, dar cea de testare scadea chiar pana la 20%, deci un fenomen puternic de overfitting. Pentru a rezolva aceasta problema, am redus si mai mult numarul de straturi pana la unul singur, care s-a dovedit a fi suficient pentru ca acuratetea pe setul de testare sa ajunga la 80%. O alta imbunatatire a constat in normarea iesirilor stratului fully connected, in loc de activarea ReLU.

Totusi, antrenand perceptronul un numar mai mare de epoci, aceasta ultima acuratete incepe si ea sa scada, reseaua intrand iar in regim de overfitting. Din acest motiv, am redus si numarul de epoci de antrenare de la 200 la 50. Performantele modelului final se pot observa in Figurile 7 si 8, iar arhitectura sa este ilustrata in Tabelul 2.

Tip	Dimensiune
Fully connected	8742 x 5
BatchNorm	-
ReLU	-
SoftMax	-

Table 2: Modelul *MLP* imbunatatit

Fig. 7: Acuratetele modelului *MLP* imbunatatitFig. 8: Lossurile modelului *MLP* imbunatatit



## 6 Bonus - VGG + SVC

Am folosit reseaua reziduala preantrenata *VGG v2*, cu care am obtinut cate 512 caracteristici pentru fiecare imagine. Pe acestea le-am dat ca input unui *SVC*, cu care am clasificat imaginile. Astfel, modelul in acest caz e format din 2 "submodele", a caror invatare este "intrerupta" in punctul in care iesirea *VGG-ului* este preluata de *SVC*. "Intreruperea" aceasta vine si cu o intrerupere a antrenarii. Astfel, antrenarea *SVC-ului* nu influenteaza *VGG-ul* si invers.

Din aces motiv, acuratatea acestui "metamodel" pe setul mic de date este de 57.58%, mai mica decat cea a *CNN-ului* din Sectiunea 4. Din acelasi motiv ce tine de proportia fiecarei clase in seturile de antrenare si testare, acuratetea metamodelului pe intregul set de date este infima: 4.63%, comparabila cu a *CNN-ului* din Sectiunea 3.

## Bibliografie

1. *Labeled Faces in the Wild*  
<http://vis-www.cs.umass.edu/lfw/>  
Data ultimei accesari: 01 June 2021