

# COVID-19 CASES FROM CHEST X-RAY IMAGES

Avădănei Antonia și Predescu Eduard; Mentor: Emanuela Haller

Universitatea din Bucuresti, Facultatea de Matematică și Informatică



## Introducere

Pandemia de Coronavirus(COVID-19) continuă să aibă un efect devastator asupra sănătății și bunăstării populației globale. Un pas critic în lupta împotriva COVID-19 este screening-ul eficient al pacienților infectați, una dintre abordările cheie de screening fiind examinarea radiologică folosind radiografia toracică. În studiile timpurii s-a constatat că pacienții infectați cu COVID-19 prezintă anumite anomalii în radiografiile lor toracice. Așadar, scopul proiectului nostru este să antrenăm un model de clasificare care să identifice cazuri de Covid-19 din radiografii pulmonare.

## Dataset

Pentru acest proiect am folosit setul de date COVID-19 CHEST X-RAY DATABASE ce conține 21.165 imagini grayscale de dimensiune 299x299, împărțite în 4 clase: Normal, COVID, Viral Pneumonia, Lung\_Opacity.

Distribuția datelor este urmatoarea:

	Normal	COVID	Viral Pneumonia	Lung Opacity
Nr. samples	10.192	3.616	1.345	6.012

Fig. 1: Distribuția datelor înainte de split.

Ulterior, împărțim datele in 3 split-uri: train, test și validation in proporție de 70%, 15%, 15%, și aducem imaginile la formatul 128x128. Astfel, avem următoarea distribuție uniformă:

	Normal	COVID	Viral Pneumonia	Lung Opacity	Total
Train	7.134	2.532	941	4210	14.815
Test	1.529	542	202	901	3.175
Validation	1.529	542	202	901	3.175
Total	10.192	3.616	1.345	6.012	21.165

Fig. 2: Distribuția datelor dupa split.

Pentru a crește artificial datasetul, am augmentat datele prin rotații, flip orizontal și am folosit pentru normalizare mean si standard deviation din ImageNet.

## Metodă

În acest proiect am folosit 2 arhitecturi:

- O rețea simplă creată de catre noi de la 0 pe care am antrenat-o atât cu date augmentate cât și cu date neaugmentate.
- O retea ResNet18 pe care am folosit-o antrenată from scratch si preantrenată pe 1000 de clase din datasetul Imagenet pe care am antrenat-o doar cu datele augmentate.

Am folosit transfer learning pentru că are mai multe beneficii, printre care se numără un timp de antrenare mult mai mic, o performanță mult mai ridicată a rețelei neurale, cât și faptul că nu avem nevoie de foarte multe date de antrenare pentru o performanță bună.

## Arhitectură

Arhitecturile celor două modele folosite sunt următoarele:

Pentru rețeaua creată de către noi:

- Conv2d(3, 6, kernel\_size=(5, 5), stride=(1, 1))
- MaxPool2d(kernel\_size=8, stride=8, padding=0, dilation=1)
- Conv2d(6, 16, kernel\_size=(5, 5), stride=(1, 1))
- Linear(in\_features=16, out\_features=100, bias=True)
- Linear(in\_features=100, out\_features=50, bias=True)
- Linear(in\_features=50, out\_features=4, bias=True)

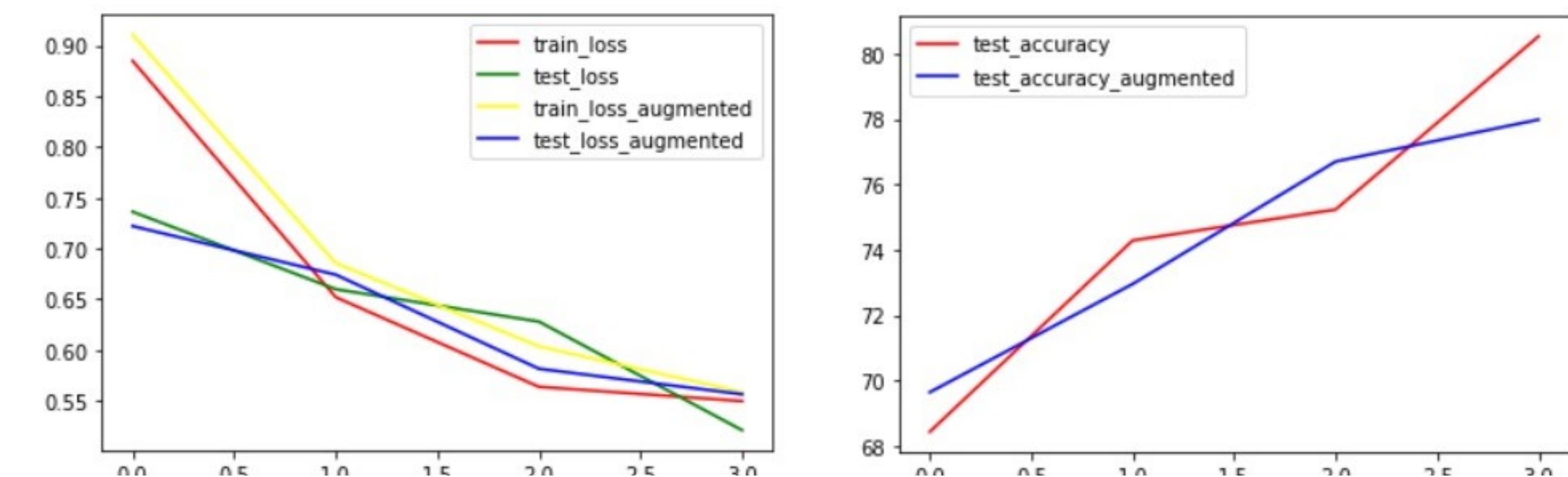
Pentru rețeaua ResNet18(am modificat output\_size=4 corespunzător celor 4 clase):

Layer Name	Output Size	ResNet-18
conv1	$112 \times 112 \times 64$	$7 \times 7, 64, \text{stride } 2$
conv2_x	$56 \times 56 \times 64$	$3 \times 3 \text{ max pool, stride } 2$ $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	$7 \times 7 \text{ average pool}$
fully connected	1000	$512 \times 1000 \text{ fully connections}$
softmax	1000	

Fig. 3: Arhitectura Resnet18

## Rezultate

Pentru rețeaua creată de către noi am antrenat 2 modele, unul folosind setul de date nemodificat, iar pentru celalalt model am folosit augmentările prezentate anterior. Ca funcție de loss am folosit negative log likelihood, iar ca optimizer SGD cu learning rate = 0.01 si momentum = 0.9. Am antrenat modelul pentru 7-10 epoci, însă am observat că atinge rezultatele maxime la epoca 4. Rezultatele sunt următoarele:



## Rezultate

Matricele de confuzie pentru rețeaua creată de către noi, pe datele augmentate și cele neaugmentate.

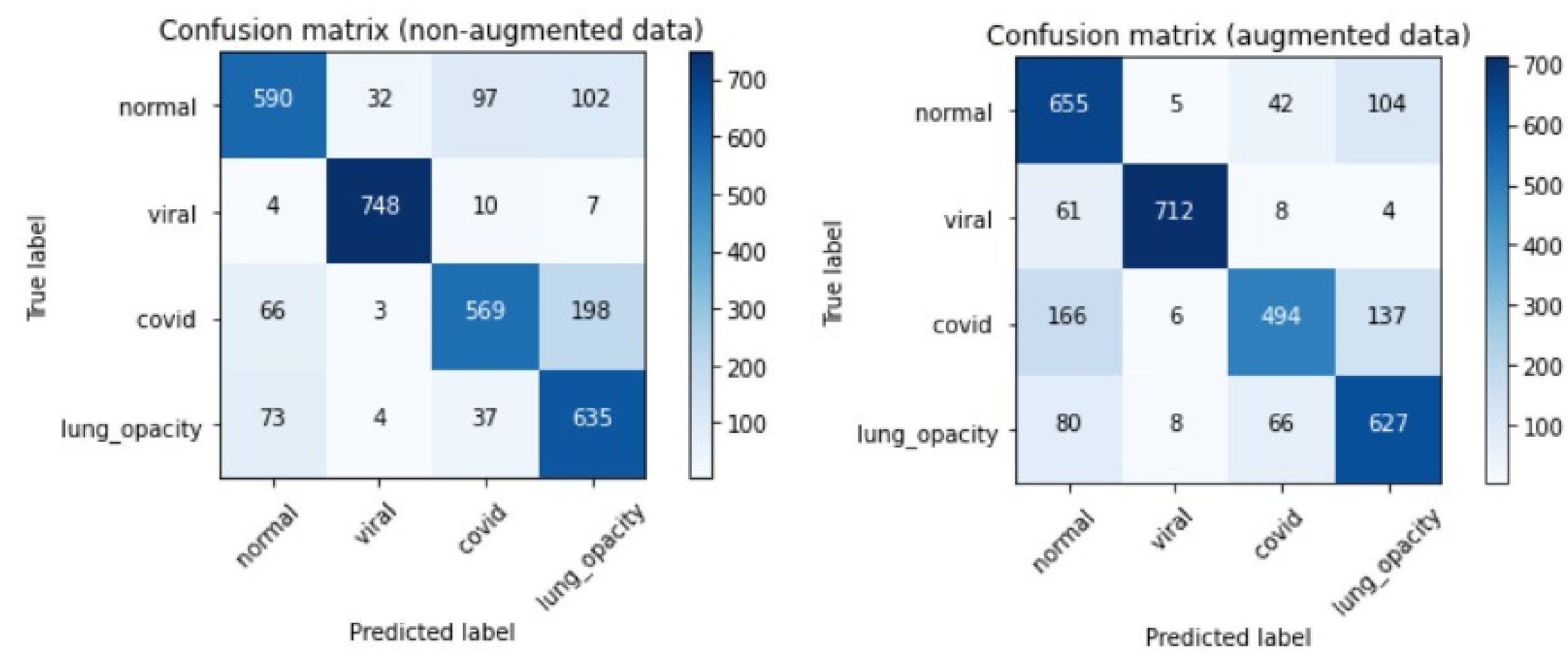


Fig. 5: Matricele de confuzie

Pentru Resnet18 am folosit CrossEntropyLoss și Adam optimizer cu learning rate = 3e-5, si rezultatele au fost vizibil mai bune, acuratetea ajungand la 95% pe pretrained si 84% from scratch, spre deosebire de 80% la modelele anterioare create de catre noi.

Validation Loss: 0.1553, Accuracy: 0.9531

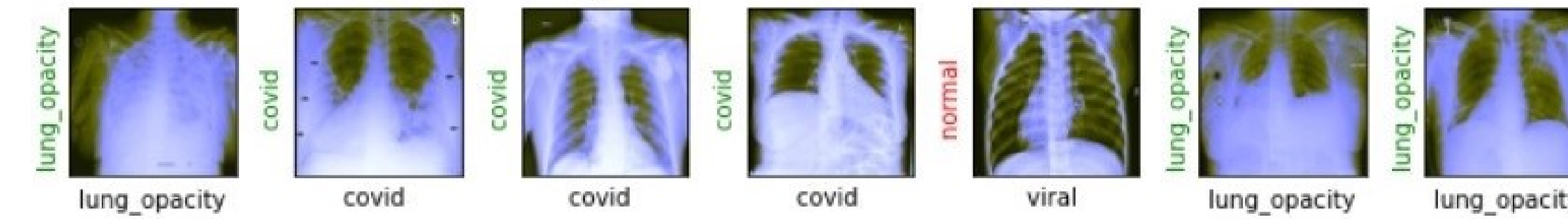


Fig. 6: Rezultate Resnet18 pretrained

Validation Loss: 0.3959, Accuracy: 0.8494

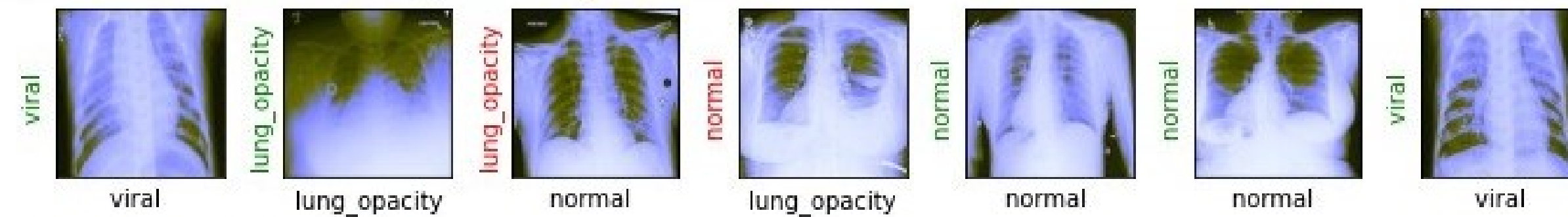


Fig. 7: Rezultate Resnet18 from scratch

## Concluzii

Având drept samples imagini cu radiografii, nu am putut realiza augmentări prea diverse, rezumandu-ne doar la rotatii folosind unghiuri mici și flip orizontal, ceea ce observăm că nu aduc deloc îmbunătățiri, ba chiar acuratețea scade cu 2%. Totuși, este de observat că rețeaua ResNet18, împreună cu o mai bună alegere a funcției de loss și a optimizerului au reușit să obțină rezultate mult mai bune.

## Referințe

- Kaiming He et al. Deep residual learning for image recognition. CVPR 2016
- M.E.H. Chowdhury et al. Can AI help in screening Viral and COVID-19 pneumonia?. IEEE Access, Vol. 8, 2020, pp. 132665 - 132676.
- Rahman, T. et al. Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images. arXiv preprint arXiv:2012.02238 2020