Assignment: Classifying Images with Convolutional Neural Networks (CNNs)

In this assignment you will work with a dataset, which consists of images of lung CT Scans available on Kaggle. You are required to develop a classifier that can take an image of a lung scan as input and predict whether the individual has COVID-19 or not. You need to train and evaluate your models using 5-fold cross validation of the data available at the aforementioned link (you can combine the positive and negative examples and use stratified cross validation).

Please refer to this paper for more details about the dataset and some methodological guidelines. You may have to use lung or lesion masks to achieve higher accuracy.

Complete the following steps, while optimizing your model in each case to get the best possible accuracy.

Q 1.

a) Build your own deep CNN using convolutional and pooling layers from Pytorch (highly recommended) or other tool of your preference and improve the accuracy (I.e., 5-fold cross validated) of the model by trying out different model architectures and hyper parameters. Try going as deep as training time allows you. Compute the average precision, recall and F1 score for the models you tried. Report your results[1] (see note 1 below).

b) Try using techniques such as normalization, drop out and early stopping to reduce overfitting and report the results[1].

Q2.

c) Use an already trained CNN such as VGG-19 or ResNet 50 and retrain its last few layers using the given data and evaluate your 5-fold cross validated accuracy. This is called Transfer Learning. Try different networks and parameters to improve the accuracy. Report your results[1].

d) Try using techniques such as normalization, drop out and early stopping to reduce overfitting and report the results[1].

Note 1:

For each question above, report in a table, the architecture of the network (the size and the type of each layer used) and the results (average precision, recall and F1) for the 3 best models you got according to the F1 score. See an example below.

Network Architecture	Results
For example, 5×5×10 conv (here, 5×5 is the convolutional kernel size and 10 is the number of channels/filters used) 2×2 max pool 4×4×20 conv 2×2 max pool 3×3×c conv 2×2 max pool 20x2 fully connected (You can list them comma separated if there are many layers) Details of techniques used for reducing overfitting (for parts b)	Precision = xx Recall = xx F1=xx
Details of the 2nd best performing model	
Details of the 3rd best performing model	

Deliverables:

- A PFD report with the results as required for Questions 1 and 2 parts a) and b) as described above.
- Source code submitted in a separate zip file.