Classification & detection of COVID-19 cases using deep neural networks with X-ray images

Ali EL ALAOUI

Ibtissam RHAYOU

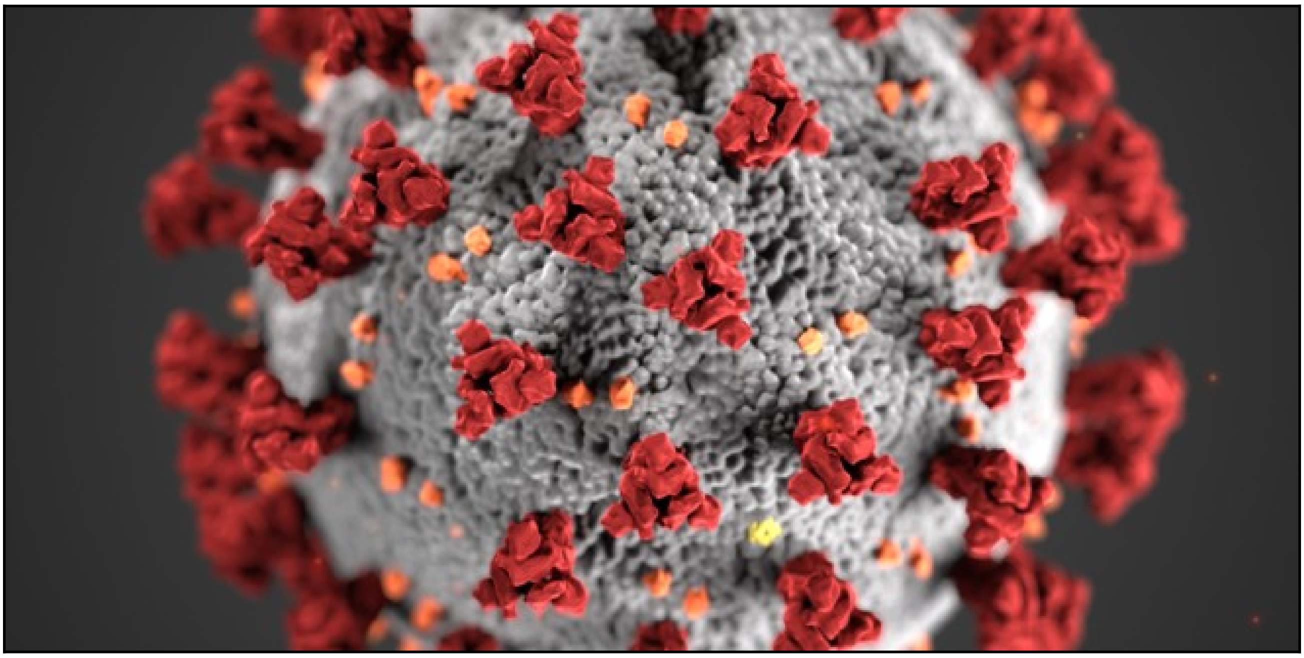
Imane BOUAYAD

Amal CHATIR

Table of content

Introduction:

Coronavirus disease 2019 (COVID-19) is defined as an illness caused by a novel coronavirus now called severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2; formerly called 2019-nCoV), which was first identified amid an outbreak of respiratory illness cases in Wuhan City, Hubei Province, China, and since **X-Ray** scans of the lungs can detect features of viral pneumonia and assess severity of lung damage. The method can be used to diagnose COVID-19. hence the idea of using what we've learned in our machine learning class to detect COVID-19 using X-Ray scans.



Problem Statement:

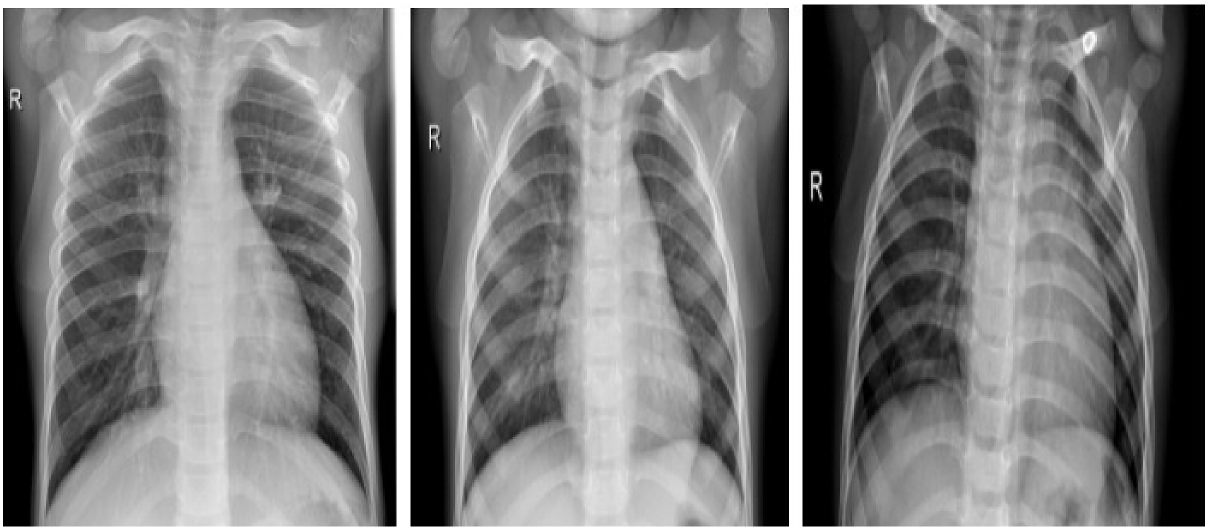
Our project is focused on predicting if a person is affected with COVID-19 Virus, using chest X-ray and CT images of patients which are positive or suspected of COVID-19 or other viral and bacterial pneumonia (MERS, SARS, and ARDS.). And use these images to develop AI-based approaches to predict and understand the infection.

Material and methods:

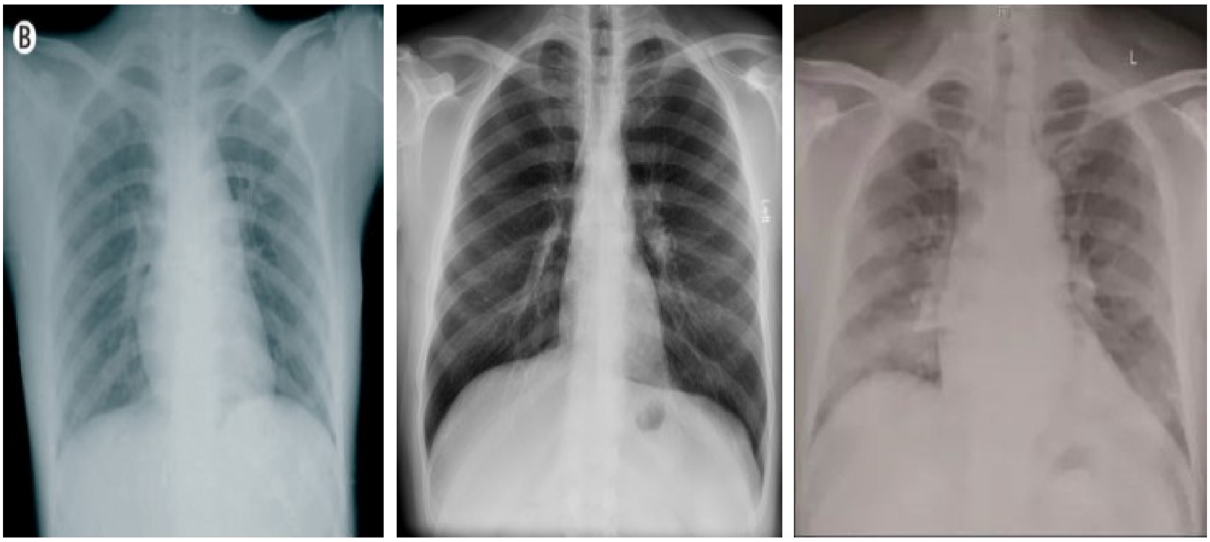
X-ray image DataSet:

The data set used in this work is obtained from two different sources used for the diagnosis of COVID-19, which currently consists of around 444 X-Ray images of COVID-19 positive patients and total 555 images. The repository of images is open for contributions and new images are added frequently. All the images are verified and annotated including the findings of the X-ray. Websites such as Radiopedia.org and Figure1.com were used to collect the images. Here, we have selected only the PA views in the training and testing for the chest X-Rays. After selecting PA view, we have 142 X-ray images of COVID-19 positive patients. For normal healthy patients, we have used the Kaggle's Chest X-Ray Images (Pneumonia) consisting of 5863 images. These images are divided into two different class values– 'Normal' and 'Covid'. From this data set, we have selected around 111 images of normal X-rays. This data set is further converted into training and testing with the split of 80% data for training and 20% data for testing purpose.

Normal(patient not affected):



Covid:



Data pre-processing

Since the data set is not uniform and the X-ray images are of different sizes therefore, we have converted all the images into the same size of 64 x 64 pixels. For this, RGB reordering has been applied and the final input to the proposed model is provided as 64 × 64 × 3 pixels, we reshape and standardize the images before feeding them to the network.

Model: 4-Layer Neural Network

Neural Networks are built of simple elements called neurons, which take in a real value, multiply it by a weight, and run it through a non-linear activation function. By constructing multiple layers of neurons, each of which receives part of the input variables, and then passes on its results to the next layers, the network can learn very complex functions. Theoretically, a neural network is capable of learning the shape of just any function, given enough computational power.

In the context of our project we opted for a **4-Layer Neural Networks**, because its Strength lies in the fact that it's very effective for high dimensionality problems, able to deal with complex relations between variables, non-exhaustive category sets and complex functions relating input to output variables. Powerful tuning options to prevent over- and under-fitting.

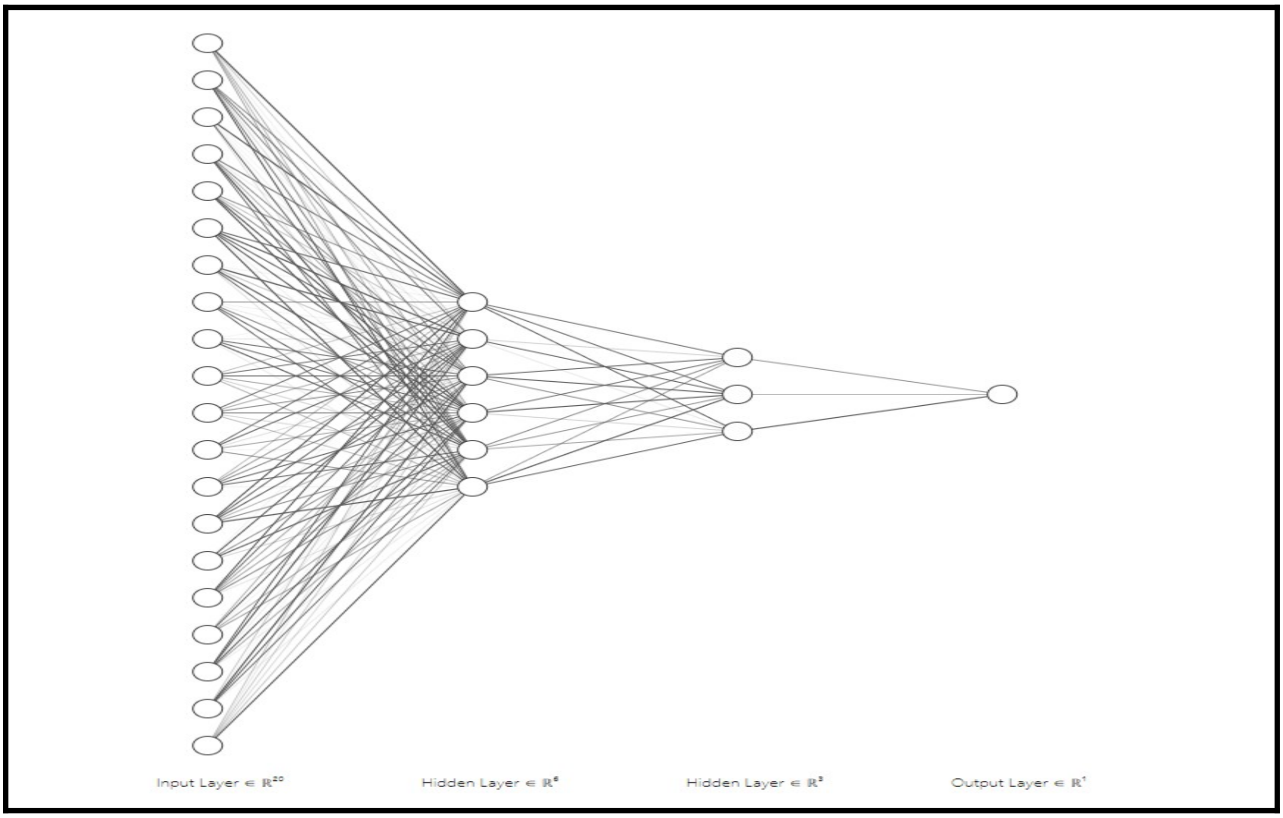
In the case of target detection which in our case is detecting whether the patient has COVID-19 or not, the **output layer** only needs a single node. The output of this node is thresholded to provide a positive or negative indication of the target's presence or absence in the input data.

Defining the neural network structure:

The proposed model depends on the working of a 4-layer Neural Network.
The applied parameters in this model consist of [12288, 20, 6, 3, 1] :

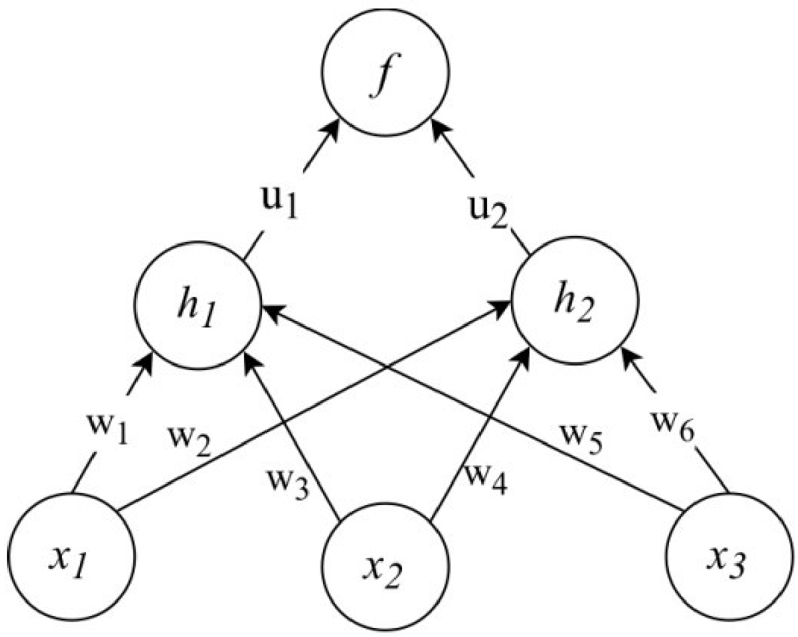
- Input Layer : 12288 nodes.

- Hidden Layer 1: 20 nodes.
- Hidden Layer 2: 6 nodes.
- Hidden Layer 3: 3 nodes.
- Output Layer 4: 1 nodes.



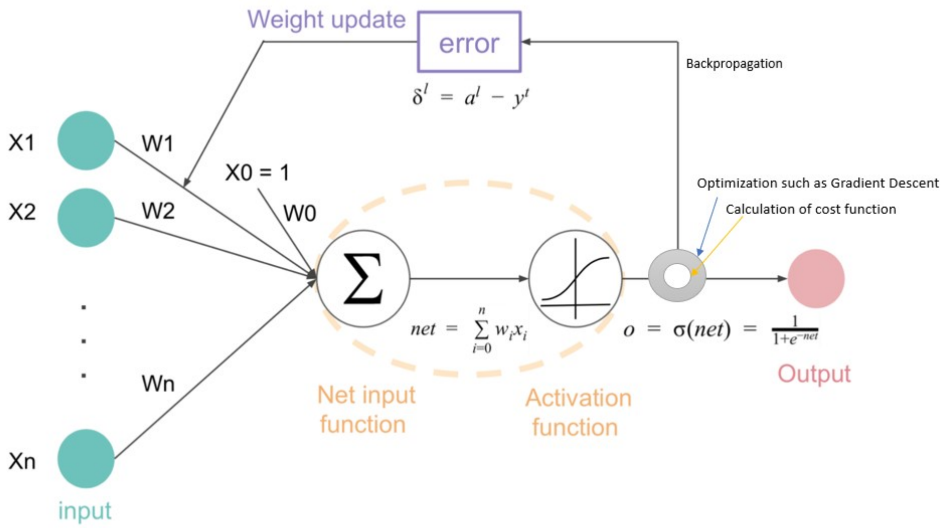
Forward Propagation :

Forward propagation (or forward pass) refers to the calculation and storage of intermediate variables (including outputs) for a neural network in order from the input layer to the output layer.



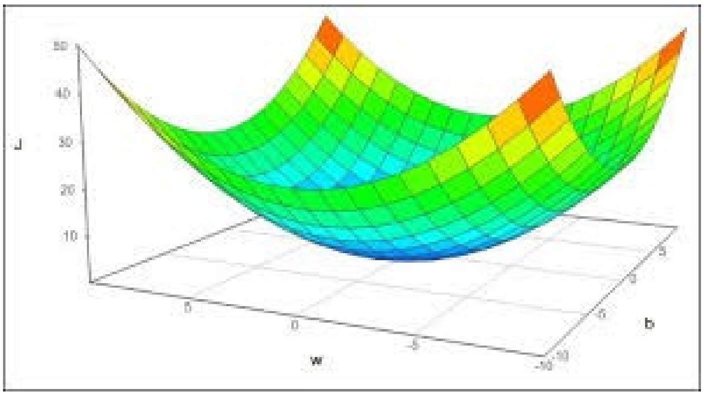
Backpropagation:

Backpropagation refers to the method of calculating the gradient of neural network parameters. In short, the method traverses the network in reverse order, from the output to the input layer, according to the chain rule from calculus. The algorithm stores any intermediate variables (partial derivatives) required while calculating the gradient with respect to some parameters.



Cost function:

One of the reasons we use cost function for logistic regression is that it's a convex function with a single global optimum. You can imagine rolling a ball down the bowl-shaped function (image bellow)—it would settle at the bottom.

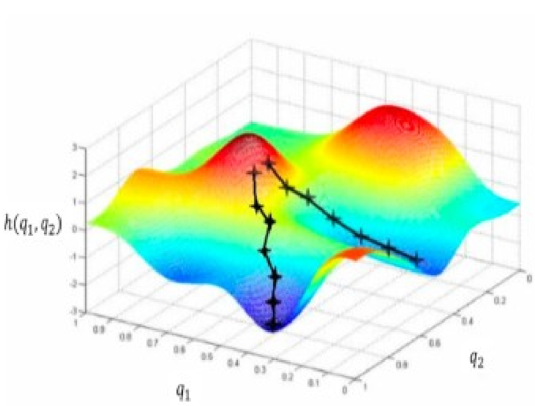


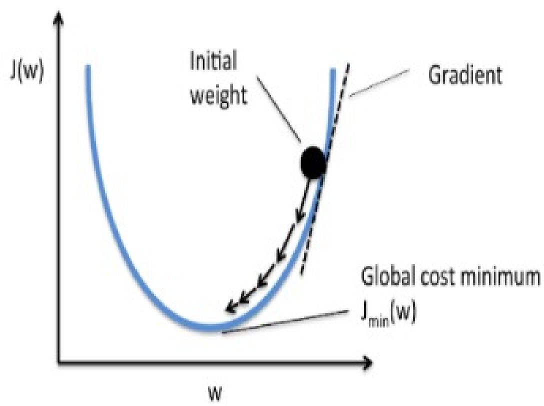
Similarly is to find the minimum of the cost function, we need to get to the lowest point. To do that, we can start from anywhere on the function and iteratively move down in the direction of the steepest slope, adjusting the values of w and b that lead us to the minimum.

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))]$$

Gradient Descent:

Gradient descent is an optimization algorithm used to find the values of parameters (coefficients) of a function (f) that minimizes a cost function (cost).



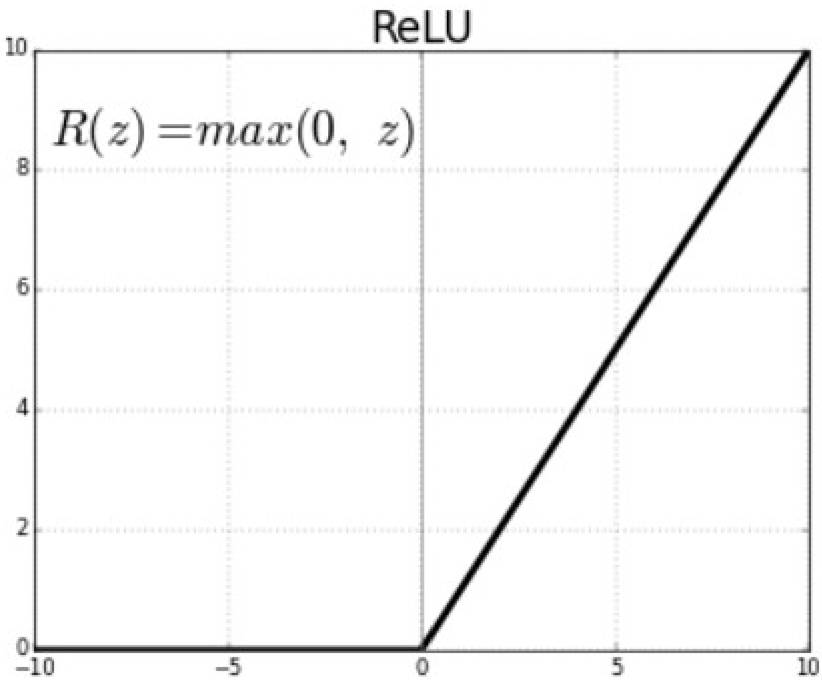


Activation function:

We used two different activation functions for the hidden layer and the output layer:

<u>hidden layer: ReLU</u>

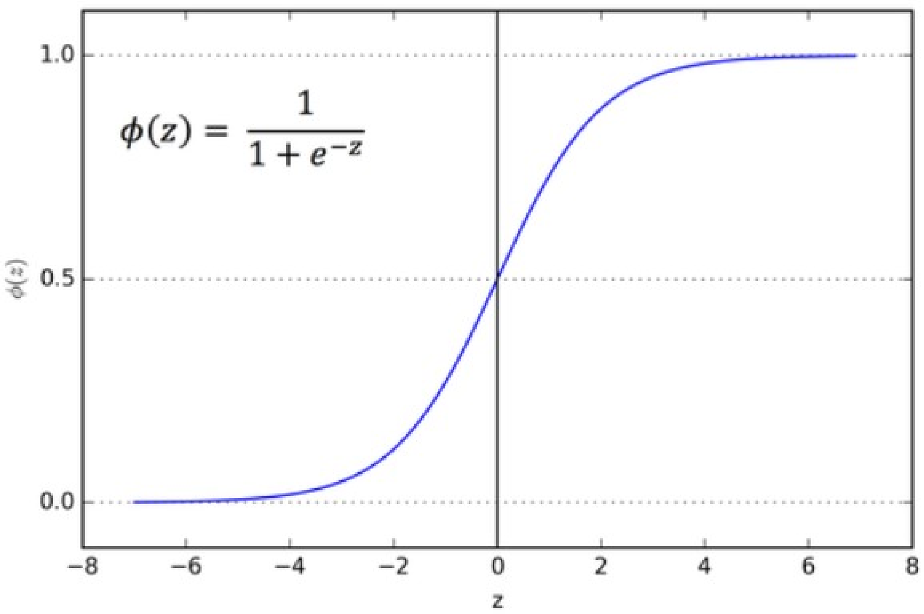
The **ReLU** is the most used activation function in the world right now.Since, it is used in almost all the convolutional neural networks or deep learning.



As you can see, the ReLU is half rectified (from bottom). f(z) is zero when z is less than zero and f(z) is equal to z when z is above or equal to zero.

- **Computationally efficient**: allows the network to converge very quickly
- **Non-linear**: although it looks like a linear function, ReLU has a derivative function and allows for backpropagation

<u>output layer: Sigmoid</u>



The main reason why we use sigmoid function is because it exists between (0 to 1). Therefore, it is especially used for models where we have to predict the probability as an output.Since probability of anything exists only between the range of 0 and 1, sigmoid is the right choice.

Discussion and conclusions

This is a proven fact that rigorous testing and social distancing are some of the most important measures to be taken by the governments in different parts of the world to control the COVID-19 pandemic. But there are a lot of limitations that can be overcomed with AI. Proposed model can detect a COVID-19 positive patient. With the limited amount of data we had, we were able to achieve a 90% true positive rate. By adding more samples of chest X-ray to the training data set and improving this model architecture, we can get higher accuracy. With the fast detection of COVID-19, we will be able to contact and isolate the COVID-19 patients and reduce community transmission.

References:

https://www.coursera.org/learn/machine-learning/home/welcome

https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia#IM-0129-0001.jpeg

https://github.com/ieee8023/covid-chestxray-dataset/issues/37