Sanjay Mahto

Final Report:

Pneumonia detection on chest X-ray

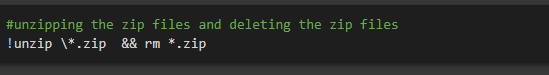
# Problem Statement

In this pandemic pneumonia is one of the most common symptom of covid-19. This project aims to analyze the chest X-rays of person and identify if person has pneumonia or not. It would be highly beneficial in the current situation with drastic increase in patients.

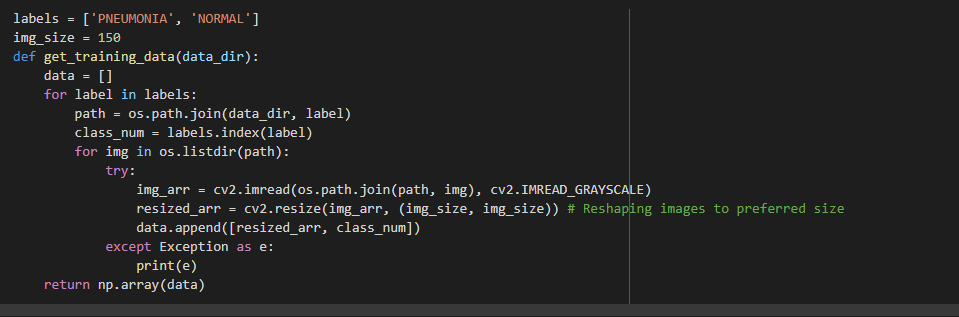
The [data](https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia) was taken from Kaggles.com which is openly available for the general public for the analysis. This would be really helpful for the medical practitioner who are dealing with the chest x-ray on day today basis and also to the general public to identify the pneumonia using this model.

# Data Gathering and preprocessing

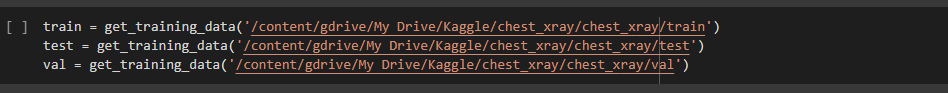
The acquired dataset consists around 6 thousands chest x-rays in 2 different category “Pneumonia” and “Normal”. In the raw dataset we had around 1300 Normal chest x-rays and around 3700 Pneumonia chest x-ray. Dataset was divided into 3 separate directories “Train”, ”test” and “Validate”



Two labels were created for the two categories of data and image size foe each image was fixed to 150. A function was created to read and resize all the image to the desired image size.



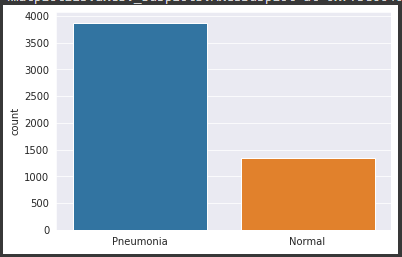
After the function has been defined it was called for 3 different sets of data and stored in the train, test and validate variable for future processing.



Once we had the data in the respective variables labels were appended in the training data which will help the model train and predict.

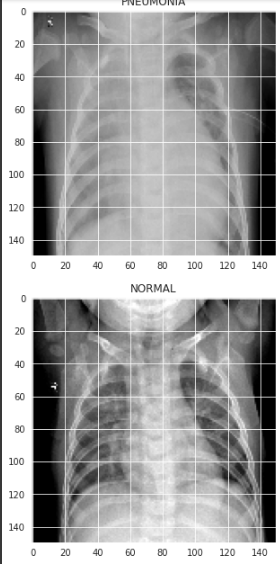


After appending the labels to the respective images it was visualize which presented the value of Normal and pneumonia x-ray.

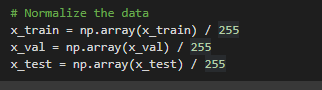


# Data Exploration

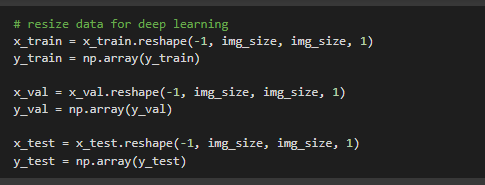
To visualize the data further 1 sample from each category was rendered. Below is the resulting image from both of the category.



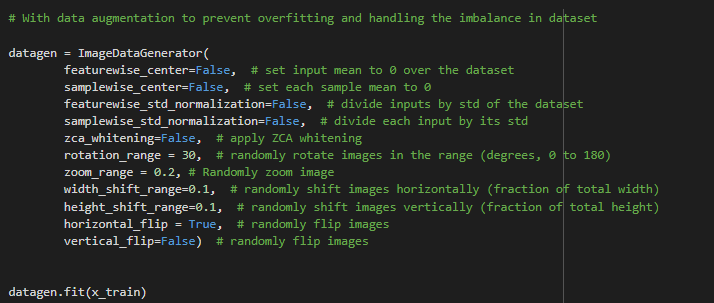
After the sample data visualization, Normalization of the data was done to highlight the image uniformly and to have a better view of the pixels



Next we have reshaped the data to have a consistency with the pixels across all the images in the dataset



Next part is to generate some random images based on the training images which will help the model not to over fit on the data provided for the training



# 

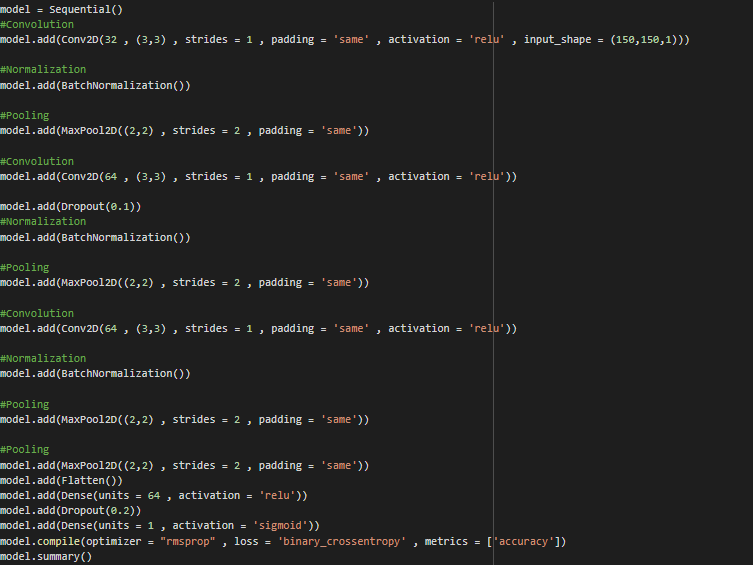
# Model building and optimizing

The last stage of the project is model building and optimizing of the model to get the best results out of the model. In this project we have used Convolutional Neural Network (CNN)

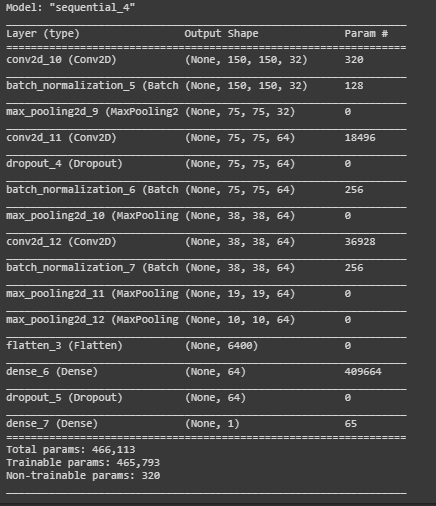
**Convolutional Neural Network**

For this project we have built the CNN model with 2 hidden layers with pooling, batch normalization and dropout layers. ‘relu’ activation function is used in all the layers.

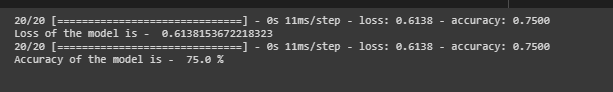
‘rmsprop’ optimizer was used to tune the model and acquire the best hyper parameter for the model.



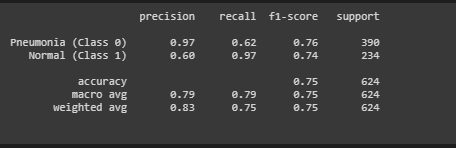
Model Summary:



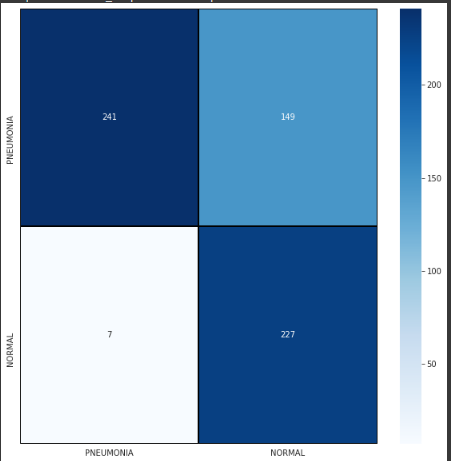
Accuracy: 75%



Classification Report:



# Confusion Matrix:



# Conclusion

So the final conclusion was the Model is working pretty good with 75% accuracy. We can develop the model further to increase the accuracy of the model.

# Future Research

Future research on this data could be adding more layer to increase the accuracy of the model and deploying the model on the cloud for general public to use.