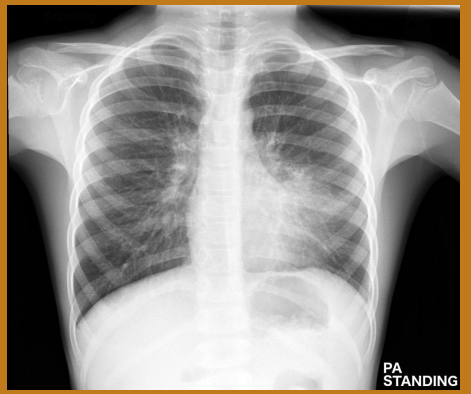
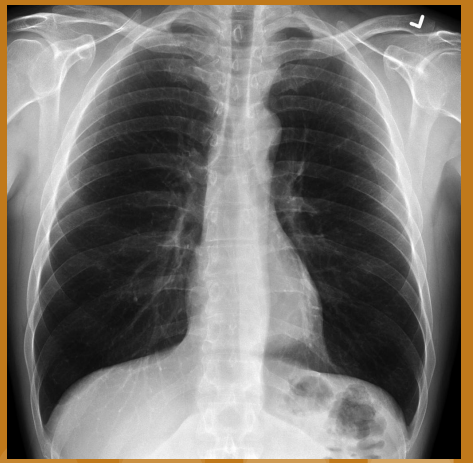
Pneumonia Image Classifier

Alisa Radjabova

Why are we doing this?



- Be able to classify x-ray as having pneumonia or not
- Be able to eliminate the human error of predicting based on x-rays

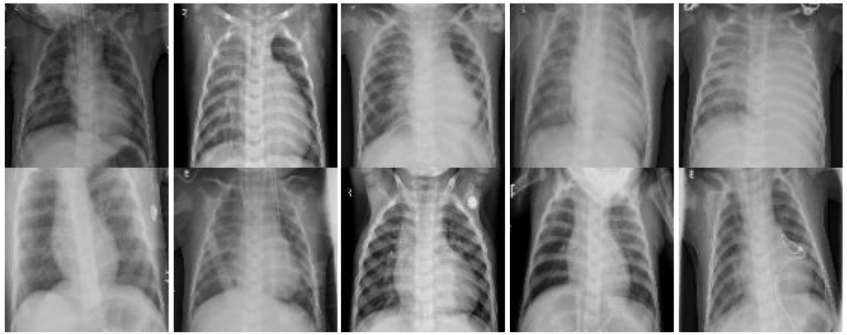


Problems with the Data

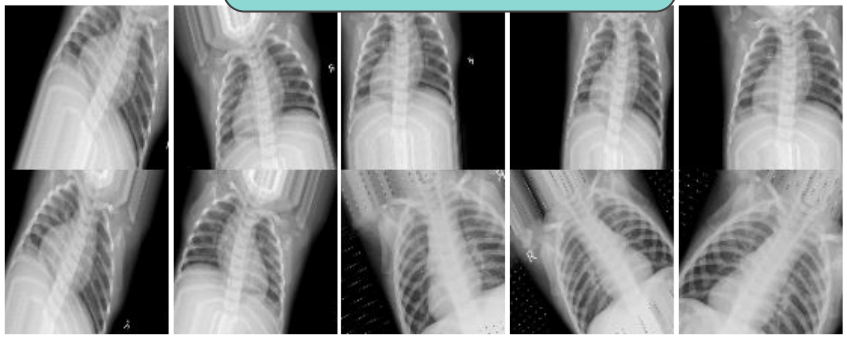
- For people it would be hard to identify very accurately if a patient has pneumonia or not

- The number of normal x-rays is a lot less than the number of pneumonia

Pneumonia

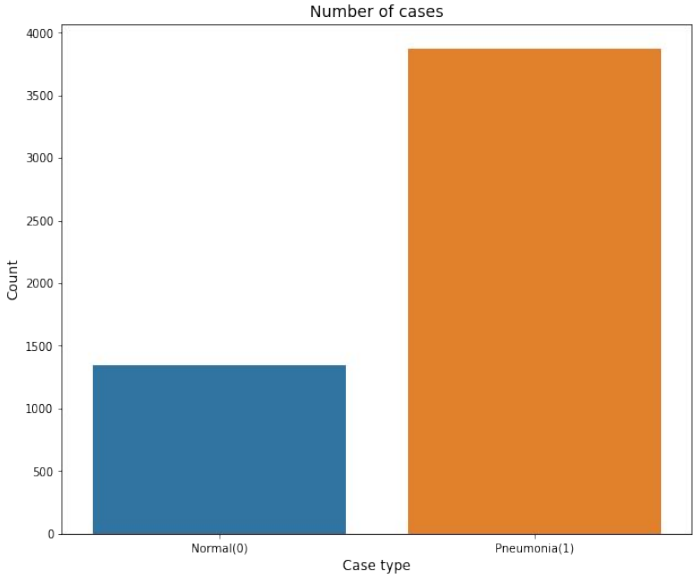


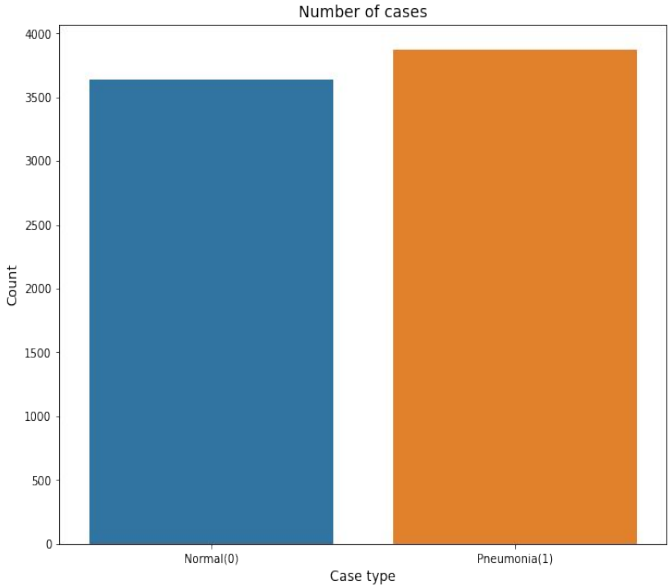
Normal



Balancing Targets
Image Augmentation

- In order to train our model, we need to have equal amounts of our target.
- We created these images to have different variations of the normal x-rays





Summary of Model

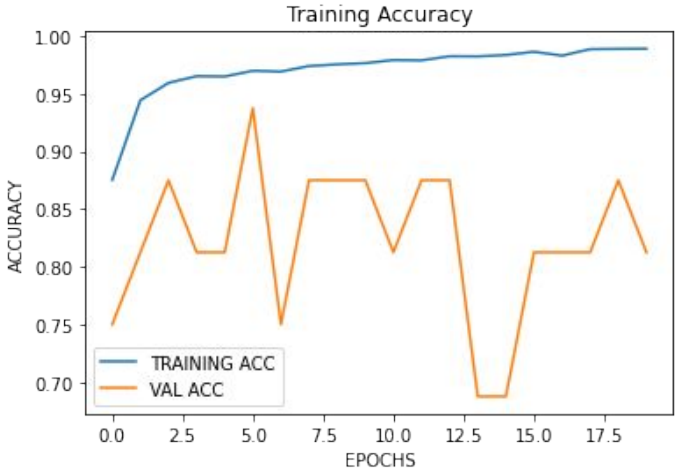
- There are a 3 different filters being applied to the images (different sizes of them)
- There are 3 hidden layers with shortening nodes/parameters
- Total of 823,193 (not enough power for bigger

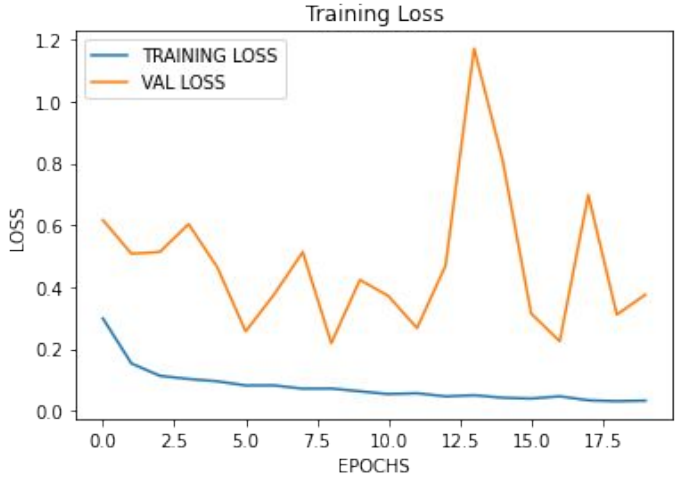
```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 253, 253, 8)       392
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 84, 84, 8)         0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 81, 81, 8)         1032
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 27, 27, 8)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 24, 24, 8)         1032
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 8, 8, 8)           0
_________________________________________________________________
flatten_1 (Flatten)          (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1024)              525312
_________________________________________________________________
dropout_1 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               262400
_________________________________________________________________
dropout_2 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_3 (Dense)              (None, 128)               32896
_________________________________________________________________
dropout_3 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_4 (Dense)              (None, 1)                 129
=================================================================
Total params: 823,193
Trainable params: 823,193
Non-trainable params: 0
```

Accuracy and Loss

- Because of the the Dropout layer we are not overfitting out model. (explain at end)

- After each run (epoch) of the model, the loss is calculated (how far we are from the correct target)

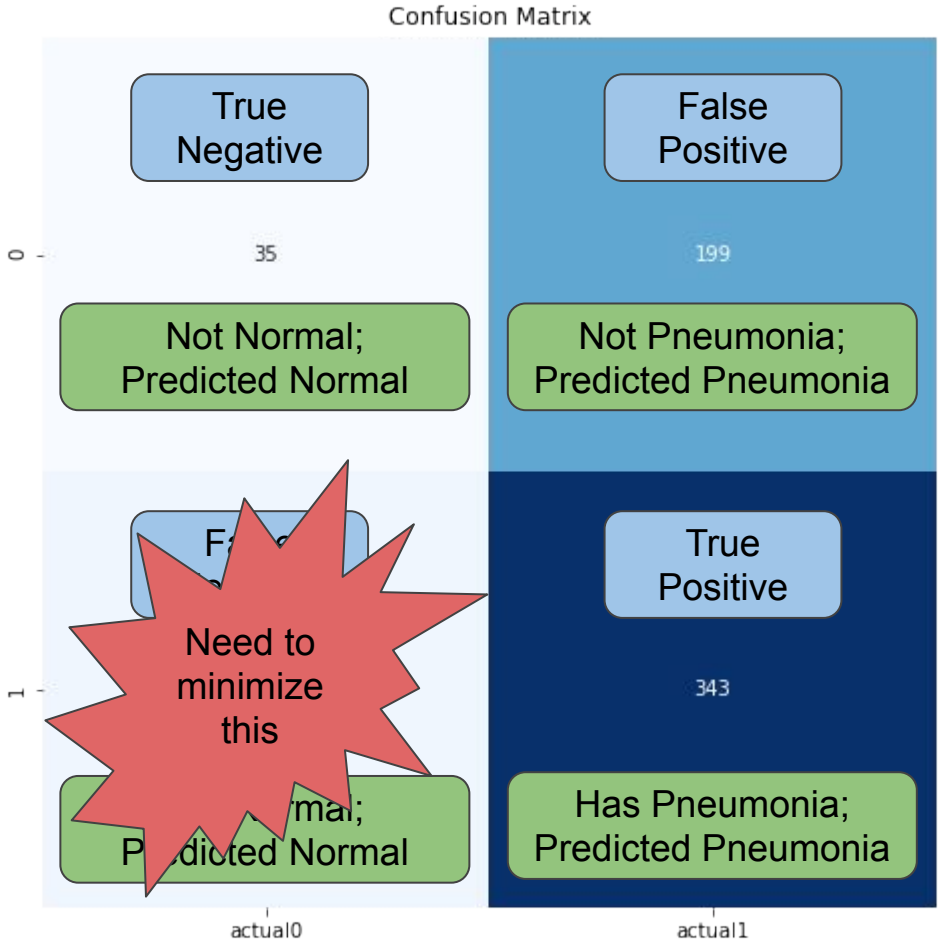




Confusion Matrix

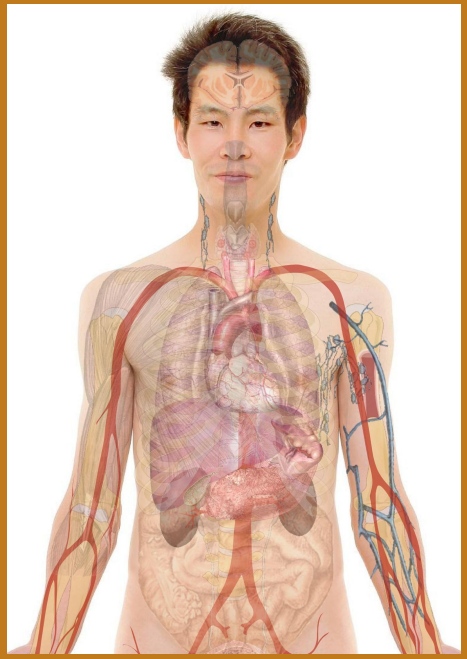
- Recall -> 88%
- Accuracy -> 73%

- Recall is more valuable here because the cost analysis of the model shows it is better to have false positives because of the value of patient health/life



Future Work:

- More images of normal x-rays, lets models make variations equally
- Create larger Neural Networks, use autoencoders, or look into anomalies
- Work on stronger computers/servers to fit bigger models



Thank You