

Unveiling Pneumonia in Chest X-Rays: The Power of Deep Learning

➡ Basic Idea Of The Project -

➡ A large dataset of chest X-rays, labelled as normal or pneumonia, is fed into the CNN.

➡ The CNN learns to identify patterns in the X-rays that are associated with pneumonia. These patterns might include opacities (cloudy areas) in the lungs.

➡ Once trained, the CNN can then be used to analyze new X-rays and predict whether they are normal or show signs of pneumonia.

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn.metrics import auc
import keras
import tensorflow as tf
from keras import backend as K
from keras import metrics
from keras.regularizers import l2
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout, BatchNormal
from keras.preprocessing.image import ImageDataGenerator
from keras.metrics import AUC
from keras.optimizers import Adam
from keras.callbacks import EarlyStopping
from sklearn.metrics import classification_report, confusion_matrix
```

```
In [2]: # Set a seed value
seed_value = 45

# 1. Set `PYTHONHASHSEED` environment variable at a fixed value
import os
os.environ['PYTHONHASHSEED'] = str(seed_value)

# 2. Set `python` built-in pseudo-random generator at a fixed value
import random
random.seed(seed_value)

# 3. Set `numpy` pseudo-random generator at a fixed value
import numpy as np
np.random.seed(seed_value)

# 4. Set `tensorflow` pseudo-random generator at a fixed value
```



```
shuffle=False,  
seed = 42)
```

```
test_generator.reset()
```

Found 5216 images belonging to 2 classes.

Found 16 images belonging to 2 classes.

Found 624 images belonging to 2 classes.

```
In [4]: #train the model  
cnn = Sequential()  
cnn.add(InputLayer(input_shape=(hyper_dimension, hyper_dimension, hyper_channels)))  
  
cnn.add(Conv2D(filters=32, kernel_size=3, activation='relu'))  
cnn.add(MaxPooling2D(pool_size=(2,2)))  
  
cnn.add(Conv2D(filters=32, kernel_size=3, activation='relu'))  
cnn.add(MaxPooling2D(pool_size=(2,2)))  
  
cnn.add(Conv2D(filters=32, kernel_size=3, activation='relu'))  
cnn.add(MaxPooling2D(pool_size=(2,2)))  
  
cnn.add(Flatten())  
  
cnn.add(Dense(activation='relu', units=128))  
cnn.add(Dense(activation='sigmoid', units=1))  
  
cnn.compile(optimizer= 'adam', loss='binary_crossentropy', metrics=[AUC()])  
cnn_model = cnn.fit(train_generator,  
                    steps_per_epoch = len(train_generator),  
                    epochs = 200,  
                    validation_data = val_generator,  
                    validation_steps = len(val_generator),  
                    verbose=2)
```

Epoch 1/200
41/41 - 35s - loss: 0.5537 - auc: 0.6175 - val_loss: 0.8473 - val_auc: 0.6953 - 35s/
epoch - 844ms/step

Epoch 2/200
41/41 - 32s - loss: 0.3634 - auc: 0.8939 - val_loss: 0.5466 - val_auc: 0.8438 - 32s/
epoch - 784ms/step

Epoch 3/200
41/41 - 33s - loss: 0.2612 - auc: 0.9446 - val_loss: 0.5695 - val_auc: 0.8438 - 33s/
epoch - 812ms/step

Epoch 4/200
41/41 - 34s - loss: 0.2269 - auc: 0.9587 - val_loss: 0.5723 - val_auc: 0.8672 - 34s/
epoch - 820ms/step

Epoch 5/200
41/41 - 33s - loss: 0.2088 - auc: 0.9648 - val_loss: 0.6007 - val_auc: 0.8672 - 33s/
epoch - 802ms/step

Epoch 6/200
41/41 - 33s - loss: 0.1834 - auc: 0.9721 - val_loss: 0.7619 - val_auc: 0.8359 - 33s/
epoch - 795ms/step

Epoch 7/200
41/41 - 33s - loss: 0.1687 - auc: 0.9776 - val_loss: 0.9801 - val_auc: 0.8281 - 33s/
epoch - 794ms/step

Epoch 8/200
41/41 - 33s - loss: 0.1810 - auc: 0.9730 - val_loss: 0.7109 - val_auc: 0.8438 - 33s/
epoch - 797ms/step

Epoch 9/200
41/41 - 32s - loss: 0.1583 - auc: 0.9794 - val_loss: 1.1539 - val_auc: 0.7656 - 32s/
epoch - 786ms/step

Epoch 10/200
41/41 - 32s - loss: 0.1654 - auc: 0.9776 - val_loss: 1.2458 - val_auc: 0.7969 - 32s/
epoch - 789ms/step

Epoch 11/200
41/41 - 34s - loss: 0.1477 - auc: 0.9820 - val_loss: 0.6348 - val_auc: 0.7969 - 34s/
epoch - 832ms/step

Epoch 12/200
41/41 - 42s - loss: 0.1487 - auc: 0.9808 - val_loss: 0.6172 - val_auc: 0.8750 - 42s/
epoch - 1s/step

Epoch 13/200
41/41 - 40s - loss: 0.1467 - auc: 0.9822 - val_loss: 0.7123 - val_auc: 0.8125 - 40s/
epoch - 982ms/step

Epoch 14/200
41/41 - 36s - loss: 0.1388 - auc: 0.9844 - val_loss: 0.5306 - val_auc: 0.7969 - 36s/
epoch - 881ms/step

Epoch 15/200
41/41 - 37s - loss: 0.1390 - auc: 0.9841 - val_loss: 0.7222 - val_auc: 0.8359 - 37s/
epoch - 914ms/step

Epoch 16/200
41/41 - 34s - loss: 0.1360 - auc: 0.9845 - val_loss: 0.7900 - val_auc: 0.8438 - 34s/
epoch - 826ms/step

Epoch 17/200
41/41 - 35s - loss: 0.1573 - auc: 0.9792 - val_loss: 0.8279 - val_auc: 0.8281 - 35s/
epoch - 864ms/step

Epoch 18/200
41/41 - 36s - loss: 0.1419 - auc: 0.9834 - val_loss: 0.4654 - val_auc: 0.9219 - 36s/
epoch - 886ms/step

Epoch 19/200
41/41 - 35s - loss: 0.1378 - auc: 0.9841 - val_loss: 0.6148 - val_auc: 0.8438 - 35s/

epoch - 845ms/step
Epoch 20/200
41/41 - 35s - loss: 0.1367 - auc: 0.9846 - val_loss: 0.8095 - val_auc: 0.9062 - 35s/
epoch - 845ms/step
Epoch 21/200
41/41 - 36s - loss: 0.1332 - auc: 0.9850 - val_loss: 0.4230 - val_auc: 0.9219 - 36s/
epoch - 870ms/step
Epoch 22/200
41/41 - 35s - loss: 0.1332 - auc: 0.9848 - val_loss: 0.6207 - val_auc: 0.8672 - 35s/
epoch - 844ms/step
Epoch 23/200
41/41 - 34s - loss: 0.1211 - auc: 0.9874 - val_loss: 0.5140 - val_auc: 0.8438 - 34s/
epoch - 819ms/step
Epoch 24/200
41/41 - 34s - loss: 0.1163 - auc: 0.9888 - val_loss: 1.0480 - val_auc: 0.8281 - 34s/
epoch - 818ms/step
Epoch 25/200
41/41 - 35s - loss: 0.1263 - auc: 0.9861 - val_loss: 0.6067 - val_auc: 0.8750 - 35s/
epoch - 854ms/step
Epoch 26/200
41/41 - 33s - loss: 0.1164 - auc: 0.9880 - val_loss: 0.5773 - val_auc: 0.9062 - 33s/
epoch - 809ms/step
Epoch 27/200
41/41 - 33s - loss: 0.1259 - auc: 0.9860 - val_loss: 0.5061 - val_auc: 0.8281 - 33s/
epoch - 806ms/step
Epoch 28/200
41/41 - 33s - loss: 0.1292 - auc: 0.9858 - val_loss: 0.3949 - val_auc: 0.8984 - 33s/
epoch - 807ms/step
Epoch 29/200
41/41 - 34s - loss: 0.1267 - auc: 0.9859 - val_loss: 0.9182 - val_auc: 0.7734 - 34s/
epoch - 821ms/step
Epoch 30/200
41/41 - 33s - loss: 0.1128 - auc: 0.9892 - val_loss: 0.5417 - val_auc: 0.8594 - 33s/
epoch - 808ms/step
Epoch 31/200
41/41 - 33s - loss: 0.1180 - auc: 0.9880 - val_loss: 0.6424 - val_auc: 0.8125 - 33s/
epoch - 811ms/step
Epoch 32/200
41/41 - 33s - loss: 0.1185 - auc: 0.9883 - val_loss: 0.5581 - val_auc: 0.8906 - 33s/
epoch - 804ms/step
Epoch 33/200
41/41 - 34s - loss: 0.1163 - auc: 0.9885 - val_loss: 0.6163 - val_auc: 0.9141 - 34s/
epoch - 822ms/step
Epoch 34/200
41/41 - 33s - loss: 0.1198 - auc: 0.9876 - val_loss: 0.7615 - val_auc: 0.8594 - 33s/
epoch - 809ms/step
Epoch 35/200
41/41 - 33s - loss: 0.1196 - auc: 0.9877 - val_loss: 0.4391 - val_auc: 0.9531 - 33s/
epoch - 805ms/step
Epoch 36/200
41/41 - 34s - loss: 0.1068 - auc: 0.9892 - val_loss: 0.4613 - val_auc: 0.8594 - 34s/
epoch - 840ms/step
Epoch 37/200
41/41 - 33s - loss: 0.1091 - auc: 0.9893 - val_loss: 0.3981 - val_auc: 0.9688 - 33s/
epoch - 804ms/step
Epoch 38/200

41/41 - 34s - loss: 0.1123 - auc: 0.9895 - val_loss: 0.5505 - val_auc: 0.9375 - 34s/
epoch - 834ms/step
Epoch 39/200
41/41 - 35s - loss: 0.1071 - auc: 0.9901 - val_loss: 0.3771 - val_auc: 0.9375 - 35s/
epoch - 861ms/step
Epoch 40/200
41/41 - 34s - loss: 0.1075 - auc: 0.9901 - val_loss: 0.8111 - val_auc: 0.8906 - 34s/
epoch - 825ms/step
Epoch 41/200
41/41 - 34s - loss: 0.1087 - auc: 0.9891 - val_loss: 0.6567 - val_auc: 0.8125 - 34s/
epoch - 820ms/step
Epoch 42/200
41/41 - 33s - loss: 0.1040 - auc: 0.9908 - val_loss: 0.8594 - val_auc: 0.8203 - 33s/
epoch - 805ms/step
Epoch 43/200
41/41 - 33s - loss: 0.1080 - auc: 0.9899 - val_loss: 0.6514 - val_auc: 0.8984 - 33s/
epoch - 801ms/step
Epoch 44/200
41/41 - 33s - loss: 0.1022 - auc: 0.9905 - val_loss: 0.6955 - val_auc: 0.8125 - 33s/
epoch - 804ms/step
Epoch 45/200
41/41 - 33s - loss: 0.1078 - auc: 0.9901 - val_loss: 0.6022 - val_auc: 0.8750 - 33s/
epoch - 808ms/step
Epoch 46/200
41/41 - 33s - loss: 0.0959 - auc: 0.9921 - val_loss: 0.4018 - val_auc: 0.9141 - 33s/
epoch - 795ms/step
Epoch 47/200
41/41 - 34s - loss: 0.1018 - auc: 0.9905 - val_loss: 1.2249 - val_auc: 0.7656 - 34s/
epoch - 832ms/step
Epoch 48/200
41/41 - 33s - loss: 0.0970 - auc: 0.9917 - val_loss: 0.4175 - val_auc: 0.9062 - 33s/
epoch - 799ms/step
Epoch 49/200
41/41 - 33s - loss: 0.1008 - auc: 0.9910 - val_loss: 0.6849 - val_auc: 0.8594 - 33s/
epoch - 800ms/step
Epoch 50/200
41/41 - 33s - loss: 0.0966 - auc: 0.9920 - val_loss: 0.9250 - val_auc: 0.7578 - 33s/
epoch - 801ms/step
Epoch 51/200
41/41 - 34s - loss: 0.0934 - auc: 0.9923 - val_loss: 0.3396 - val_auc: 1.0000 - 34s/
epoch - 821ms/step
Epoch 52/200
41/41 - 34s - loss: 0.0914 - auc: 0.9924 - val_loss: 0.6979 - val_auc: 0.7891 - 34s/
epoch - 832ms/step
Epoch 53/200
41/41 - 34s - loss: 0.0967 - auc: 0.9918 - val_loss: 0.6117 - val_auc: 0.9141 - 34s/
epoch - 833ms/step
Epoch 54/200
41/41 - 33s - loss: 0.0919 - auc: 0.9918 - val_loss: 0.8236 - val_auc: 0.8906 - 33s/
epoch - 803ms/step
Epoch 55/200
41/41 - 33s - loss: 0.1002 - auc: 0.9909 - val_loss: 0.3257 - val_auc: 1.0000 - 33s/
epoch - 808ms/step
Epoch 56/200
41/41 - 35s - loss: 0.0992 - auc: 0.9916 - val_loss: 0.7599 - val_auc: 0.9062 - 35s/
epoch - 849ms/step

Epoch 57/200
41/41 - 35s - loss: 0.0855 - auc: 0.9940 - val_loss: 0.5465 - val_auc: 0.8438 - 35s/
epoch - 857ms/step
Epoch 58/200
41/41 - 33s - loss: 0.0850 - auc: 0.9938 - val_loss: 0.4373 - val_auc: 0.9062 - 33s/
epoch - 806ms/step
Epoch 59/200
41/41 - 34s - loss: 0.0876 - auc: 0.9934 - val_loss: 0.4414 - val_auc: 0.8906 - 34s/
epoch - 819ms/step
Epoch 60/200
41/41 - 33s - loss: 0.0898 - auc: 0.9927 - val_loss: 0.5918 - val_auc: 0.8906 - 33s/
epoch - 799ms/step
Epoch 61/200
41/41 - 35s - loss: 0.0962 - auc: 0.9922 - val_loss: 0.4532 - val_auc: 0.9375 - 35s/
epoch - 865ms/step
Epoch 62/200
41/41 - 35s - loss: 0.0896 - auc: 0.9931 - val_loss: 0.3411 - val_auc: 0.9062 - 35s/
epoch - 849ms/step
Epoch 63/200
41/41 - 42s - loss: 0.0916 - auc: 0.9929 - val_loss: 0.5045 - val_auc: 0.8594 - 42s/
epoch - 1s/step
Epoch 64/200
41/41 - 43s - loss: 0.0899 - auc: 0.9934 - val_loss: 0.4829 - val_auc: 0.9219 - 43s/
epoch - 1s/step
Epoch 65/200
41/41 - 38s - loss: 0.0846 - auc: 0.9936 - val_loss: 0.4347 - val_auc: 0.9219 - 38s/
epoch - 917ms/step
Epoch 66/200
41/41 - 39s - loss: 0.0847 - auc: 0.9933 - val_loss: 0.5669 - val_auc: 0.8672 - 39s/
epoch - 940ms/step
Epoch 67/200
41/41 - 38s - loss: 0.0903 - auc: 0.9933 - val_loss: 0.3970 - val_auc: 0.9219 - 38s/
epoch - 930ms/step
Epoch 68/200
41/41 - 37s - loss: 0.0798 - auc: 0.9946 - val_loss: 0.2934 - val_auc: 0.9688 - 37s/
epoch - 913ms/step
Epoch 69/200
41/41 - 41s - loss: 0.0885 - auc: 0.9926 - val_loss: 0.4062 - val_auc: 0.8672 - 41s/
epoch - 998ms/step
Epoch 70/200
41/41 - 38s - loss: 0.0869 - auc: 0.9932 - val_loss: 0.4195 - val_auc: 0.9141 - 38s/
epoch - 919ms/step
Epoch 71/200
41/41 - 35s - loss: 0.0839 - auc: 0.9943 - val_loss: 0.4381 - val_auc: 0.8594 - 35s/
epoch - 854ms/step
Epoch 72/200
41/41 - 39s - loss: 0.0840 - auc: 0.9932 - val_loss: 0.4875 - val_auc: 0.8281 - 39s/
epoch - 947ms/step
Epoch 73/200
41/41 - 39s - loss: 0.0870 - auc: 0.9933 - val_loss: 0.5496 - val_auc: 0.8594 - 39s/
epoch - 957ms/step
Epoch 74/200
41/41 - 36s - loss: 0.0860 - auc: 0.9936 - val_loss: 0.5844 - val_auc: 0.8438 - 36s/
epoch - 885ms/step
Epoch 75/200
41/41 - 35s - loss: 0.0799 - auc: 0.9946 - val_loss: 0.5747 - val_auc: 0.8281 - 35s/

epoch - 846ms/step
Epoch 76/200
41/41 - 34s - loss: 0.0835 - auc: 0.9944 - val_loss: 0.7030 - val_auc: 0.7500 - 34s/
epoch - 821ms/step
Epoch 77/200
41/41 - 34s - loss: 0.0861 - auc: 0.9938 - val_loss: 0.7061 - val_auc: 0.8438 - 34s/
epoch - 833ms/step
Epoch 78/200
41/41 - 34s - loss: 0.0755 - auc: 0.9945 - val_loss: 0.6064 - val_auc: 0.7812 - 34s/
epoch - 825ms/step
Epoch 79/200
41/41 - 33s - loss: 0.0761 - auc: 0.9948 - val_loss: 0.4687 - val_auc: 0.9375 - 33s/
epoch - 807ms/step
Epoch 80/200
41/41 - 33s - loss: 0.0743 - auc: 0.9950 - val_loss: 0.7859 - val_auc: 0.9062 - 33s/
epoch - 812ms/step
Epoch 81/200
41/41 - 33s - loss: 0.0807 - auc: 0.9941 - val_loss: 0.5824 - val_auc: 0.8906 - 33s/
epoch - 815ms/step
Epoch 82/200
41/41 - 33s - loss: 0.0761 - auc: 0.9951 - val_loss: 0.6870 - val_auc: 0.8750 - 33s/
epoch - 811ms/step
Epoch 83/200
41/41 - 34s - loss: 0.0735 - auc: 0.9957 - val_loss: 0.4051 - val_auc: 0.9688 - 34s/
epoch - 821ms/step
Epoch 84/200
41/41 - 33s - loss: 0.0767 - auc: 0.9944 - val_loss: 0.6610 - val_auc: 0.8438 - 33s/
epoch - 813ms/step
Epoch 85/200
41/41 - 33s - loss: 0.0781 - auc: 0.9948 - val_loss: 0.4563 - val_auc: 0.8750 - 33s/
epoch - 807ms/step
Epoch 86/200
41/41 - 34s - loss: 0.0695 - auc: 0.9957 - val_loss: 0.8514 - val_auc: 0.8750 - 34s/
epoch - 822ms/step
Epoch 87/200
41/41 - 33s - loss: 0.0742 - auc: 0.9942 - val_loss: 0.5297 - val_auc: 0.8594 - 33s/
epoch - 807ms/step
Epoch 88/200
41/41 - 34s - loss: 0.0693 - auc: 0.9953 - val_loss: 0.4272 - val_auc: 0.8750 - 34s/
epoch - 830ms/step
Epoch 89/200
41/41 - 33s - loss: 0.0712 - auc: 0.9951 - val_loss: 0.4780 - val_auc: 0.8750 - 33s/
epoch - 814ms/step
Epoch 90/200
41/41 - 34s - loss: 0.0828 - auc: 0.9941 - val_loss: 0.5999 - val_auc: 0.8438 - 34s/
epoch - 817ms/step
Epoch 91/200
41/41 - 34s - loss: 0.0730 - auc: 0.9955 - val_loss: 0.7769 - val_auc: 0.8594 - 34s/
epoch - 824ms/step
Epoch 92/200
41/41 - 33s - loss: 0.0690 - auc: 0.9959 - val_loss: 0.6844 - val_auc: 0.7969 - 33s/
epoch - 810ms/step
Epoch 93/200
41/41 - 34s - loss: 0.0722 - auc: 0.9950 - val_loss: 0.8943 - val_auc: 0.8438 - 34s/
epoch - 819ms/step
Epoch 94/200

41/41 - 33s - loss: 0.0692 - auc: 0.9953 - val_loss: 0.8562 - val_auc: 0.8047 - 33s/
epoch - 807ms/step
Epoch 95/200
41/41 - 33s - loss: 0.0686 - auc: 0.9955 - val_loss: 0.6654 - val_auc: 0.9062 - 33s/
epoch - 814ms/step
Epoch 96/200
41/41 - 33s - loss: 0.0678 - auc: 0.9956 - val_loss: 0.5583 - val_auc: 0.8438 - 33s/
epoch - 810ms/step
Epoch 97/200
41/41 - 33s - loss: 0.0729 - auc: 0.9947 - val_loss: 0.3928 - val_auc: 0.8984 - 33s/
epoch - 812ms/step
Epoch 98/200
41/41 - 34s - loss: 0.0719 - auc: 0.9952 - val_loss: 0.4386 - val_auc: 0.8906 - 34s/
epoch - 819ms/step
Epoch 99/200
41/41 - 33s - loss: 0.0640 - auc: 0.9963 - val_loss: 0.3450 - val_auc: 0.9219 - 33s/
epoch - 810ms/step
Epoch 100/200
41/41 - 33s - loss: 0.0692 - auc: 0.9956 - val_loss: 0.4635 - val_auc: 0.8594 - 33s/
epoch - 808ms/step
Epoch 101/200
41/41 - 33s - loss: 0.0637 - auc: 0.9967 - val_loss: 0.8826 - val_auc: 0.8594 - 33s/
epoch - 815ms/step
Epoch 102/200
41/41 - 39s - loss: 0.0736 - auc: 0.9950 - val_loss: 0.3660 - val_auc: 0.9062 - 39s/
epoch - 959ms/step
Epoch 103/200
41/41 - 33s - loss: 0.0679 - auc: 0.9959 - val_loss: 0.7723 - val_auc: 0.8047 - 33s/
epoch - 815ms/step
Epoch 104/200
41/41 - 33s - loss: 0.0659 - auc: 0.9959 - val_loss: 0.5924 - val_auc: 0.8359 - 33s/
epoch - 814ms/step
Epoch 105/200
41/41 - 35s - loss: 0.0612 - auc: 0.9969 - val_loss: 0.4098 - val_auc: 0.9219 - 35s/
epoch - 855ms/step
Epoch 106/200
41/41 - 36s - loss: 0.0582 - auc: 0.9969 - val_loss: 0.3741 - val_auc: 0.9297 - 36s/
epoch - 875ms/step
Epoch 107/200
41/41 - 52s - loss: 0.0615 - auc: 0.9962 - val_loss: 0.4260 - val_auc: 0.9219 - 52s/
epoch - 1s/step
Epoch 108/200
41/41 - 34s - loss: 0.0646 - auc: 0.9957 - val_loss: 0.2491 - val_auc: 1.0000 - 34s/
epoch - 819ms/step
Epoch 109/200
41/41 - 33s - loss: 0.0693 - auc: 0.9962 - val_loss: 0.4231 - val_auc: 0.9141 - 33s/
epoch - 803ms/step
Epoch 110/200
41/41 - 34s - loss: 0.0705 - auc: 0.9954 - val_loss: 0.4539 - val_auc: 0.8906 - 34s/
epoch - 819ms/step
Epoch 111/200
41/41 - 34s - loss: 0.0596 - auc: 0.9964 - val_loss: 0.5307 - val_auc: 0.9375 - 34s/
epoch - 831ms/step
Epoch 112/200
41/41 - 34s - loss: 0.0589 - auc: 0.9968 - val_loss: 0.5629 - val_auc: 0.8984 - 34s/
epoch - 818ms/step

Epoch 113/200
41/41 - 33s - loss: 0.0620 - auc: 0.9950 - val_loss: 0.6351 - val_auc: 0.8594 - 33s/
epoch - 814ms/step
Epoch 114/200
41/41 - 68s - loss: 0.0616 - auc: 0.9965 - val_loss: 0.4877 - val_auc: 0.9219 - 68s/
epoch - 2s/step
Epoch 115/200
41/41 - 103s - loss: 0.0627 - auc: 0.9962 - val_loss: 0.5769 - val_auc: 0.8984 - 103
s/epoch - 3s/step
Epoch 116/200
41/41 - 101s - loss: 0.0564 - auc: 0.9973 - val_loss: 0.3599 - val_auc: 0.9141 - 101
s/epoch - 2s/step
Epoch 117/200
41/41 - 101s - loss: 0.0612 - auc: 0.9963 - val_loss: 0.5376 - val_auc: 0.9297 - 101
s/epoch - 2s/step
Epoch 118/200
41/41 - 101s - loss: 0.0583 - auc: 0.9971 - val_loss: 0.3617 - val_auc: 0.9531 - 101
s/epoch - 2s/step
Epoch 119/200
41/41 - 93s - loss: 0.0519 - auc: 0.9975 - val_loss: 0.2587 - val_auc: 0.9688 - 93s/
epoch - 2s/step
Epoch 120/200
41/41 - 81s - loss: 0.0560 - auc: 0.9971 - val_loss: 0.5473 - val_auc: 0.9062 - 81s/
epoch - 2s/step
Epoch 121/200
41/41 - 33s - loss: 0.0636 - auc: 0.9961 - val_loss: 0.4899 - val_auc: 0.9062 - 33s/
epoch - 795ms/step
Epoch 122/200
41/41 - 32s - loss: 0.0489 - auc: 0.9979 - val_loss: 0.2696 - val_auc: 0.9688 - 32s/
epoch - 790ms/step
Epoch 123/200
41/41 - 37s - loss: 0.0620 - auc: 0.9962 - val_loss: 0.7305 - val_auc: 0.8828 - 37s/
epoch - 892ms/step
Epoch 124/200
41/41 - 33s - loss: 0.0602 - auc: 0.9964 - val_loss: 0.3002 - val_auc: 0.9609 - 33s/
epoch - 804ms/step
Epoch 125/200
41/41 - 38s - loss: 0.0630 - auc: 0.9960 - val_loss: 0.6395 - val_auc: 0.8672 - 38s/
epoch - 919ms/step
Epoch 126/200
41/41 - 34s - loss: 0.0562 - auc: 0.9974 - val_loss: 0.5889 - val_auc: 0.9219 - 34s/
epoch - 819ms/step
Epoch 127/200
41/41 - 33s - loss: 0.0589 - auc: 0.9962 - val_loss: 0.7328 - val_auc: 0.8516 - 33s/
epoch - 799ms/step
Epoch 128/200
41/41 - 35s - loss: 0.0538 - auc: 0.9970 - val_loss: 0.3773 - val_auc: 0.8906 - 35s/
epoch - 850ms/step
Epoch 129/200
41/41 - 37s - loss: 0.0540 - auc: 0.9977 - val_loss: 0.6533 - val_auc: 0.8125 - 37s/
epoch - 904ms/step
Epoch 130/200
41/41 - 36s - loss: 0.0586 - auc: 0.9968 - val_loss: 0.3977 - val_auc: 0.9688 - 36s/
epoch - 875ms/step
Epoch 131/200
41/41 - 37s - loss: 0.0522 - auc: 0.9978 - val_loss: 0.4057 - val_auc: 0.9062 - 37s/

epoch - 896ms/step
Epoch 132/200
41/41 - 54s - loss: 0.0581 - auc: 0.9963 - val_loss: 0.3944 - val_auc: 0.8906 - 54s/
epoch - 1s/step
Epoch 133/200
41/41 - 81s - loss: 0.0524 - auc: 0.9974 - val_loss: 0.5207 - val_auc: 0.8516 - 81s/
epoch - 2s/step
Epoch 134/200
41/41 - 37s - loss: 0.0535 - auc: 0.9977 - val_loss: 0.2934 - val_auc: 0.9375 - 37s/
epoch - 891ms/step
Epoch 135/200
41/41 - 34s - loss: 0.0642 - auc: 0.9966 - val_loss: 0.2421 - val_auc: 0.9844 - 34s/
epoch - 833ms/step
Epoch 136/200
41/41 - 34s - loss: 0.0598 - auc: 0.9970 - val_loss: 0.3055 - val_auc: 0.9375 - 34s/
epoch - 826ms/step
Epoch 137/200
41/41 - 35s - loss: 0.0527 - auc: 0.9970 - val_loss: 0.6470 - val_auc: 0.8750 - 35s/
epoch - 852ms/step
Epoch 138/200
41/41 - 34s - loss: 0.0584 - auc: 0.9966 - val_loss: 0.3024 - val_auc: 0.9688 - 34s/
epoch - 834ms/step
Epoch 139/200
41/41 - 34s - loss: 0.0482 - auc: 0.9977 - val_loss: 0.4997 - val_auc: 0.9297 - 34s/
epoch - 836ms/step
Epoch 140/200
41/41 - 35s - loss: 0.0497 - auc: 0.9977 - val_loss: 0.4460 - val_auc: 0.8906 - 35s/
epoch - 843ms/step
Epoch 141/200
41/41 - 37s - loss: 0.0497 - auc: 0.9977 - val_loss: 0.2691 - val_auc: 0.9531 - 37s/
epoch - 909ms/step
Epoch 142/200
41/41 - 40s - loss: 0.0527 - auc: 0.9972 - val_loss: 0.2943 - val_auc: 1.0000 - 40s/
epoch - 966ms/step
Epoch 143/200
41/41 - 106s - loss: 0.0446 - auc: 0.9981 - val_loss: 0.1772 - val_auc: 1.0000 - 106
s/epoch - 3s/step
Epoch 144/200
41/41 - 39s - loss: 0.0528 - auc: 0.9962 - val_loss: 0.2900 - val_auc: 0.9844 - 39s/
epoch - 946ms/step
Epoch 145/200
41/41 - 38s - loss: 0.0582 - auc: 0.9969 - val_loss: 0.3715 - val_auc: 0.9297 - 38s/
epoch - 930ms/step
Epoch 146/200
41/41 - 42s - loss: 0.0431 - auc: 0.9985 - val_loss: 0.3124 - val_auc: 0.9219 - 42s/
epoch - 1s/step
Epoch 147/200
41/41 - 38s - loss: 0.0458 - auc: 0.9983 - val_loss: 0.3182 - val_auc: 0.9375 - 38s/
epoch - 923ms/step
Epoch 148/200
41/41 - 37s - loss: 0.0459 - auc: 0.9983 - val_loss: 0.3264 - val_auc: 0.9219 - 37s/
epoch - 901ms/step
Epoch 149/200
41/41 - 37s - loss: 0.0446 - auc: 0.9983 - val_loss: 0.3263 - val_auc: 0.9375 - 37s/
epoch - 901ms/step
Epoch 150/200

41/41 - 36s - loss: 0.0414 - auc: 0.9983 - val_loss: 0.2896 - val_auc: 0.9531 - 36s/
epoch - 879ms/step
Epoch 151/200
41/41 - 38s - loss: 0.0454 - auc: 0.9979 - val_loss: 0.2856 - val_auc: 0.9375 - 38s/
epoch - 935ms/step
Epoch 152/200
41/41 - 37s - loss: 0.0448 - auc: 0.9976 - val_loss: 0.3275 - val_auc: 0.9531 - 37s/
epoch - 892ms/step
Epoch 153/200
41/41 - 42s - loss: 0.0453 - auc: 0.9978 - val_loss: 0.2125 - val_auc: 0.9844 - 42s/
epoch - 1s/step
Epoch 154/200
41/41 - 38s - loss: 0.0430 - auc: 0.9977 - val_loss: 0.4090 - val_auc: 0.9375 - 38s/
epoch - 938ms/step
Epoch 155/200
41/41 - 39s - loss: 0.0399 - auc: 0.9983 - val_loss: 0.1998 - val_auc: 0.9688 - 39s/
epoch - 939ms/step
Epoch 156/200
41/41 - 39s - loss: 0.0462 - auc: 0.9970 - val_loss: 0.3413 - val_auc: 0.9531 - 39s/
epoch - 943ms/step
Epoch 157/200
41/41 - 36s - loss: 0.0375 - auc: 0.9988 - val_loss: 0.3211 - val_auc: 0.9531 - 36s/
epoch - 873ms/step
Epoch 158/200
41/41 - 39s - loss: 0.0434 - auc: 0.9983 - val_loss: 0.4036 - val_auc: 0.8750 - 39s/
epoch - 954ms/step
Epoch 159/200
41/41 - 39s - loss: 0.0423 - auc: 0.9982 - val_loss: 0.2812 - val_auc: 0.9531 - 39s/
epoch - 950ms/step
Epoch 160/200
41/41 - 38s - loss: 0.0390 - auc: 0.9987 - val_loss: 0.2601 - val_auc: 0.9531 - 38s/
epoch - 919ms/step
Epoch 161/200
41/41 - 38s - loss: 0.0414 - auc: 0.9983 - val_loss: 0.5686 - val_auc: 0.8438 - 38s/
epoch - 923ms/step
Epoch 162/200
41/41 - 35s - loss: 0.0436 - auc: 0.9977 - val_loss: 0.2328 - val_auc: 0.9688 - 35s/
epoch - 853ms/step
Epoch 163/200
41/41 - 38s - loss: 0.0463 - auc: 0.9983 - val_loss: 0.3617 - val_auc: 0.9375 - 38s/
epoch - 925ms/step
Epoch 164/200
41/41 - 42s - loss: 0.0404 - auc: 0.9986 - val_loss: 0.2855 - val_auc: 0.9375 - 42s/
epoch - 1s/step
Epoch 165/200
41/41 - 37s - loss: 0.0380 - auc: 0.9985 - val_loss: 0.7532 - val_auc: 0.8750 - 37s/
epoch - 911ms/step
Epoch 166/200
41/41 - 35s - loss: 0.0423 - auc: 0.9977 - val_loss: 0.4679 - val_auc: 0.9062 - 35s/
epoch - 859ms/step
Epoch 167/200
41/41 - 35s - loss: 0.0438 - auc: 0.9980 - val_loss: 0.6256 - val_auc: 0.7969 - 35s/
epoch - 844ms/step
Epoch 168/200
41/41 - 35s - loss: 0.0378 - auc: 0.9984 - val_loss: 0.4269 - val_auc: 0.9453 - 35s/
epoch - 863ms/step

Epoch 169/200
41/41 - 42s - loss: 0.0397 - auc: 0.9980 - val_loss: 0.3133 - val_auc: 0.9375 - 42s/
epoch - 1s/step
Epoch 170/200
41/41 - 39s - loss: 0.0362 - auc: 0.9989 - val_loss: 0.3302 - val_auc: 0.9219 - 39s/
epoch - 940ms/step
Epoch 171/200
41/41 - 41s - loss: 0.0452 - auc: 0.9979 - val_loss: 0.3636 - val_auc: 0.9219 - 41s/
epoch - 994ms/step
Epoch 172/200
41/41 - 38s - loss: 0.0433 - auc: 0.9985 - val_loss: 0.5031 - val_auc: 0.8906 - 38s/
epoch - 932ms/step
Epoch 173/200
41/41 - 36s - loss: 0.0413 - auc: 0.9974 - val_loss: 0.3235 - val_auc: 0.9531 - 36s/
epoch - 882ms/step
Epoch 174/200
41/41 - 37s - loss: 0.0427 - auc: 0.9978 - val_loss: 0.5816 - val_auc: 0.8906 - 37s/
epoch - 895ms/step
Epoch 175/200
41/41 - 37s - loss: 0.0376 - auc: 0.9989 - val_loss: 0.5096 - val_auc: 0.8906 - 37s/
epoch - 898ms/step
Epoch 176/200
41/41 - 37s - loss: 0.0410 - auc: 0.9987 - val_loss: 0.4269 - val_auc: 0.9062 - 37s/
epoch - 897ms/step
Epoch 177/200
41/41 - 37s - loss: 0.0469 - auc: 0.9972 - val_loss: 0.3120 - val_auc: 0.9375 - 37s/
epoch - 899ms/step
Epoch 178/200
41/41 - 36s - loss: 0.0383 - auc: 0.9985 - val_loss: 0.5084 - val_auc: 0.8594 - 36s/
epoch - 883ms/step
Epoch 179/200
41/41 - 36s - loss: 0.0382 - auc: 0.9978 - val_loss: 0.6104 - val_auc: 0.8203 - 36s/
epoch - 867ms/step
Epoch 180/200
41/41 - 39s - loss: 0.0374 - auc: 0.9985 - val_loss: 0.4868 - val_auc: 0.9219 - 39s/
epoch - 962ms/step
Epoch 181/200
41/41 - 37s - loss: 0.0382 - auc: 0.9985 - val_loss: 0.5794 - val_auc: 0.8203 - 37s/
epoch - 893ms/step
Epoch 182/200
41/41 - 34s - loss: 0.0377 - auc: 0.9989 - val_loss: 0.3552 - val_auc: 0.9219 - 34s/
epoch - 839ms/step
Epoch 183/200
41/41 - 35s - loss: 0.0383 - auc: 0.9988 - val_loss: 0.4598 - val_auc: 0.9297 - 35s/
epoch - 844ms/step
Epoch 184/200
41/41 - 35s - loss: 0.0408 - auc: 0.9980 - val_loss: 0.4545 - val_auc: 0.8438 - 35s/
epoch - 850ms/step
Epoch 185/200
41/41 - 35s - loss: 0.0409 - auc: 0.9985 - val_loss: 0.1804 - val_auc: 1.0000 - 35s/
epoch - 847ms/step
Epoch 186/200
41/41 - 34s - loss: 0.0371 - auc: 0.9986 - val_loss: 0.3079 - val_auc: 0.9375 - 34s/
epoch - 841ms/step
Epoch 187/200
41/41 - 35s - loss: 0.0365 - auc: 0.9979 - val_loss: 0.3775 - val_auc: 0.9062 - 35s/

```
epoch - 860ms/step
Epoch 188/200
41/41 - 35s - loss: 0.0376 - auc: 0.9983 - val_loss: 0.4252 - val_auc: 0.9375 - 35s/
epoch - 851ms/step
Epoch 189/200
41/41 - 34s - loss: 0.0466 - auc: 0.9979 - val_loss: 0.2014 - val_auc: 0.9688 - 34s/
epoch - 841ms/step
Epoch 190/200
41/41 - 36s - loss: 0.0329 - auc: 0.9990 - val_loss: 0.2240 - val_auc: 0.9688 - 36s/
epoch - 875ms/step
Epoch 191/200
41/41 - 35s - loss: 0.0379 - auc: 0.9986 - val_loss: 0.2682 - val_auc: 0.9531 - 35s/
epoch - 859ms/step
Epoch 192/200
41/41 - 61s - loss: 0.0308 - auc: 0.9986 - val_loss: 0.3298 - val_auc: 0.9375 - 61s/
epoch - 1s/step
Epoch 193/200
41/41 - 45s - loss: 0.0410 - auc: 0.9982 - val_loss: 0.3972 - val_auc: 0.8984 - 45s/
epoch - 1s/step
Epoch 194/200
41/41 - 34s - loss: 0.0406 - auc: 0.9986 - val_loss: 0.3069 - val_auc: 0.9375 - 34s/
epoch - 836ms/step
Epoch 195/200
41/41 - 34s - loss: 0.0321 - auc: 0.9992 - val_loss: 0.2398 - val_auc: 0.9688 - 34s/
epoch - 826ms/step
Epoch 196/200
41/41 - 38s - loss: 0.0344 - auc: 0.9989 - val_loss: 0.3773 - val_auc: 0.9219 - 38s/
epoch - 937ms/step
Epoch 197/200
41/41 - 38s - loss: 0.0314 - auc: 0.9993 - val_loss: 0.4968 - val_auc: 0.9062 - 38s/
epoch - 936ms/step
Epoch 198/200
41/41 - 44s - loss: 0.0362 - auc: 0.9985 - val_loss: 0.4066 - val_auc: 0.9219 - 44s/
epoch - 1s/step
Epoch 199/200
41/41 - 44s - loss: 0.0339 - auc: 0.9986 - val_loss: 0.3069 - val_auc: 0.9531 - 44s/
epoch - 1s/step
Epoch 200/200
41/41 - 43s - loss: 0.0451 - auc: 0.9980 - val_loss: 0.2269 - val_auc: 0.9844 - 43s/
epoch - 1s/step
```

Create a function that outputs metrics and charts

Metrics:

☐ Accuracy

☐ Precision

☐ Recall

☐ Specificity

➡ F1 Score

Charts:

➡ Train VS. Validation Loss

➡ Train VS. Validation AUC

➡ Confusion Matrix

```
In [10]: def create_charts(cnn, cnn_model):
    ## DEFINE ##
    ## Define 1: train & validation loss
    train_loss = cnn_model.history['loss']
    val_loss = cnn_model.history['val_loss']

    ## Define 2: train & validation AUC
    train_auc_name = list(cnn_model.history.keys())[3]
    val_auc_name = list(cnn_model.history.keys())[1]
    train_auc = cnn_model.history[train_auc_name]
    val_auc = cnn_model.history[val_auc_name]

    ## Define 3: y_pred & y_true
    y_true = test_generator.classes
    Y_pred = cnn.predict_generator(test_generator, steps = len(test_generator))
    y_pred = (Y_pred > 0.5).T[0]
    y_pred_prob = Y_pred.T[0]

    ## PLOT ##
    fig = plt.figure(figsize=(13, 10))

    ## PLOT 1: TRAIN VS. VALIDATION LOSS
    plt.subplot(2,2,1)
    plt.title("Training vs. Validation Loss")
    plt.plot(train_loss, label='training loss')
    plt.plot(val_loss, label='validation loss')
    plt.xlabel("Number of Epochs", size=14)
    plt.legend()

    ## PLOT 2: TRAIN VS. VALIDATION AUC
    plt.subplot(2,2,2)
    plt.title("Training vs. Validation AUC Score")
    plt.plot(train_auc, label='training auc')
    plt.plot(val_auc, label='validation auc')
    plt.xlabel("Number of Epochs", size=14)
    plt.legend()

    ## PLOT 3: CONFUSION MATRIX
    plt.subplot(2,2,3)
    # Set up the labels for in the confusion matrix
    cm = confusion_matrix(y_true, y_pred)
    names = ['True Negatives', 'False Positives', 'False Negatives', 'True Positive']
    counts = ['{0:0.0f}'.format(value) for value in cm.flatten()]
    percentages = ['{0:.2%}'.format(value) for value in cm.flatten()/np.sum(cm)]
    labels = [f'{v1}\n{v2}' for v1, v2 in zip(names, percentages)]
```

```

labels = np.asarray(labels).reshape(2,2)
ticklabels = ['Normal', 'Pneumonia']

    # Create confusion matrix as heatmap
sns.set(font_scale = 1.4)
ax = sns.heatmap(cm, annot=labels, fmt='', cmap='Oranges', xticklabels=ticklabels, yticklabels=ticklabels,
plt.xticks(size=12)
plt.yticks(size=12)
plt.title("Confusion Matrix") #plt.title("Confusion Matrix\n", fontsize=10)
plt.xlabel("Predicted", size=14)
plt.ylabel("Actual", size=14)
#plt.savefig('cm.png', transparent=True)

## PLOT 4: ROC CURVE
plt.subplot(2,2,4)
fpr, tpr, thresholds = roc_curve(y_true, y_pred_prob)
auc = roc_auc_score(y_true, y_pred_prob)
plt.title('ROC Curve')
plt.plot([0, 1], [0, 1], 'k--', label = "Random (AUC = 50%)")
plt.plot(fpr, tpr, label='CNN (AUC = {:.2f}%)' .format(auc*100))
plt.xlabel('False Positive Rate', size=14)
plt.ylabel('True Positive Rate', size=14)
plt.legend(loc='best')
#plt.savefig('roc.png', bbox_inches='tight', pad_inches=1)

## END PLOTS
plt.tight_layout()
;

## Summary Statistics
TN, FP, FN, TP = cm.ravel() # cm[0,0], cm[0, 1], cm[1, 0], cm[1, 1]
accuracy = (TP + TN) / np.sum(cm) # % positive out of all predicted positives
precision = TP / (TP+FP) # % positive out of all predicted positives
recall = TP / (TP+FN) # % positive out of all supposed to be positives
specificity = TN / (TN+FP) # % negative out of all supposed to be negatives
f1 = 2*precision*recall / (precision + recall)
stats_summary = '[Summary Statistics]\nAccuracy = {:.2%} | Precision = {:.2%} | Recall = {:.2%} | Specificity = {:.2%} | F1 Score = {:.2%}'
print(stats_summary)

```

In [12]: create_charts(cnn, cnn_model)

c:\users\91984\appdata\local\programs\python\python37\lib\site-packages\ipykernel_launcher.py:15: UserWarning: `Model.predict_generator` is deprecated and will be removed in a future version. Please use `Model.predict`, which supports generators.

```
from ipykernel import kernelapp as app
```

```
[Summary Statistics]
```

```
Accuracy = 91.83% | Precision = 89.33% | Recall = 98.72% | Specificity = 80.34% | F1 Score = 93.79%
```


