task2_template

April 17, 2020

1 Class Challenge: Image Classification of COVID-19 X-rays

2 Task 2 [Total points: 30]

2.1 Setup

- This assignment involves the following packages: 'matplotlib', 'numpy', and 'sklearn'.
- If you are using conda, use the following commands to install the above packages:

```
conda install matplotlib
conda install numpy
conda install -c anaconda scikit-learn
```

- If you are using pip, use use the following commands to install the above packages:

```
pip install matplotlib
pip install numpy
pip install sklearn
```

2.2 Data

Please download the data using the following link: COVID-19.

- After downloading 'Covid_Data_GradientCrescent.zip', unzip the file and you should see the following data structure:

|–all |———train |———test |–two |———train |———test

- Put the 'all' folder, the 'two' folder and this python notebook in the **same directory** so that the following code can correctly locate the data.

2.3 [20 points] Multi-class Classification

[2]:
```
import os

import tensorflow as tf
```

```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.preprocessing.image import ImageDataGenerator

os.environ['OMP_NUM_THREADS'] = '1'
os.environ['CUDA_VISIBLE_DEVICES'] = '-1'
tf.__version__
```

[2]:
```
'2.1.0'
```

Load Image Data

[3]:
```
DATA_LIST = os.listdir('all/train')
DATASET_PATH  = 'all/train'
TEST_DIR =  'all/test'
IMAGE_SIZE    = (224, 224)
NUM_CLASSES   = len(DATA_LIST)
BATCH_SIZE    = 10  # try reducing batch size or freeze more layers if your GPU␣
 ↪runs out of memory
NUM_EPOCHS    = 100
LEARNING_RATE = 0.0001 # start off with high rate first 0.001 and experiment␣
 ↪with reducing it gradually
```

Generate Training and Validation Batches

[4]:
```
train_datagen = ImageDataGenerator(rescale=1./
 ↪255,rotation_range=50,featurewise_center = True,
                                    featurewise_std_normalization =␣
 ↪True,width_shift_range=0.2,
                                    height_shift_range=0.2,shear_range=0.
 ↪25,zoom_range=0.1,
                                    zca_whitening = True,channel_shift_range = 20,
                                    horizontal_flip = True,vertical_flip = True,
                                    validation_split = 0.2,fill_mode='constant')


train_batches = train_datagen.
 ↪flow_from_directory(DATASET_PATH,target_size=IMAGE_SIZE,
                                                    ␣
 ↪shuffle=True,batch_size=BATCH_SIZE,
                                                     subset = "training",seed=42,
                                                     class_mode="categorical")

valid_batches = train_datagen.
 ↪flow_from_directory(DATASET_PATH,target_size=IMAGE_SIZE,
```

```
                                                    ␣
↪shuffle=True,batch_size=BATCH_SIZE,
                                                     subset = "validation",
                                                    ␣
↪seed=42,class_mode="categorical")
```

```
H:\Anaconda3\envs\tf\lib\site-
packages\keras_preprocessing\image\image_data_generator.py:341: UserWarning:
This ImageDataGenerator specifies `zca_whitening` which overrides setting
of`featurewise_std_normalization`.
  warnings.warn('This ImageDataGenerator specifies '
```

```
Found 216 images belonging to 4 classes.
Found 54 images belonging to 4 classes.
```

[10 points] Build Model Hint: Starting from a pre-trained model typically helps performance on a new task, e.g. starting with weights obtained by training on ImageNet.

[5]:
```
raise NotImplementedError("Build your model based on an architecture of your␣
↪choice "
                          "A sample model summary is shown below")
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
vgg16 (Model)                (None, 7, 7, 512)         14714688
_________________________________________________________________
flatten (Flatten)            (None, 25088)             0
_________________________________________________________________
feature_dense (Dense)        (None, 256)               6422784
_________________________________________________________________
dense (Dense)                (None, 4)                 1028
=================================================================
Total params: 21,138,500
Trainable params: 6,423,812
Non-trainable params: 14,714,688
_________________________________________________________________
None
```

[5 points] Train Model

[6]:
```
#FIT MODEL
print(len(train_batches))
print(len(valid_batches))

STEP_SIZE_TRAIN=train_batches.n//train_batches.batch_size
```

```
STEP_SIZE_VALID=valid_batches.n//valid_batches.batch_size

raise NotImplementedError("Use the model.fit function to train your network")
```

```
22
6
```

```
H:\Anaconda3\envs\tf\lib\site-
packages\keras_preprocessing\image\image_data_generator.py:716: UserWarning:
This ImageDataGenerator specifies `featurewise_center`, but it hasn't been fit
on any training data. Fit it first by calling `.fit(numpy_data)`.
  warnings.warn('This ImageDataGenerator specifies '
H:\Anaconda3\envs\tf\lib\site-
packages\keras_preprocessing\image\image_data_generator.py:735: UserWarning:
This ImageDataGenerator specifies `zca_whitening`, but it hasn't been fit on any
training data. Fit it first by calling `.fit(numpy_data)`.
  warnings.warn('This ImageDataGenerator specifies '
```

```
WARNING:tensorflow:sample_weight modes were coerced from
  ...
    to
  ['...']
WARNING:tensorflow:sample_weight modes were coerced from
  ...
    to
  ['...']
Train for 21 steps, validate for 5 steps
Epoch 1/100
```

```
H:\Anaconda3\envs\tf\lib\site-
packages\keras_preprocessing\image\image_data_generator.py:716: UserWarning:
This ImageDataGenerator specifies `featurewise_center`, but it hasn't been fit
on any training data. Fit it first by calling `.fit(numpy_data)`.
  warnings.warn('This ImageDataGenerator specifies '
H:\Anaconda3\envs\tf\lib\site-
packages\keras_preprocessing\image\image_data_generator.py:735: UserWarning:
This ImageDataGenerator specifies `zca_whitening`, but it hasn't been fit on any
training data. Fit it first by calling `.fit(numpy_data)`.
  warnings.warn('This ImageDataGenerator specifies '
```

```
21/21 [==============================] - 66s 3s/step - loss: 1.6074 - acc:
0.2427 - val_loss: 1.3532 - val_acc: 0.3600
Epoch 2/100
21/21 [==============================] - 64s 3s/step - loss: 1.2078 - acc:
0.4272 - val_loss: 1.1174 - val_acc: 0.4800
Epoch 3/100
21/21 [==============================] - 64s 3s/step - loss: 1.0575 - acc:
0.5048 - val_loss: 0.9547 - val_acc: 0.6600
Epoch 4/100
```

```
21/21 [==============================] - 63s 3s/step - loss: 1.0413 - acc:
0.5680 - val_loss: 0.9135 - val_acc: 0.5000
Epoch 5/100
21/21 [==============================] - 63s 3s/step - loss: 0.9270 - acc:
0.5971 - val_loss: 0.9027 - val_acc: 0.6800
Epoch 6/100
21/21 [==============================] - 63s 3s/step - loss: 0.9233 - acc:
0.5777 - val_loss: 0.8939 - val_acc: 0.5800
Epoch 7/100
21/21 [==============================] - 65s 3s/step - loss: 0.9431 - acc:
0.5728 - val_loss: 0.8314 - val_acc: 0.6000
Epoch 8/100
21/21 [==============================] - 64s 3s/step - loss: 0.9077 - acc:
0.6262 - val_loss: 0.8964 - val_acc: 0.6000
Epoch 9/100
21/21 [==============================] - 63s 3s/step - loss: 0.8537 - acc:
0.6456 - val_loss: 0.8214 - val_acc: 0.6800
Epoch 10/100
21/21 [==============================] - 65s 3s/step - loss: 0.8151 - acc:
0.6699 - val_loss: 0.7294 - val_acc: 0.6600
Epoch 11/100
21/21 [==============================] - 67s 3s/step - loss: 0.7999 - acc:
0.6650 - val_loss: 0.8830 - val_acc: 0.6600
Epoch 12/100
21/21 [==============================] - 68s 3s/step - loss: 0.7547 - acc:
0.6650 - val_loss: 0.7593 - val_acc: 0.6600
Epoch 13/100
21/21 [==============================] - 63s 3s/step - loss: 0.7219 - acc:
0.6505 - val_loss: 0.7540 - val_acc: 0.6200
Epoch 14/100
21/21 [==============================] - 72s 3s/step - loss: 0.7014 - acc:
0.6845 - val_loss: 0.8700 - val_acc: 0.5600
Epoch 15/100
21/21 [==============================] - 70s 3s/step - loss: 0.7170 - acc:
0.7039 - val_loss: 0.7459 - val_acc: 0.6800
Epoch 16/100
21/21 [==============================] - 67s 3s/step - loss: 0.7106 - acc:
0.6990 - val_loss: 0.7080 - val_acc: 0.6800
Epoch 17/100
21/21 [==============================] - 65s 3s/step - loss: 0.6886 - acc:
0.6942 - val_loss: 0.7756 - val_acc: 0.6800
Epoch 18/100
21/21 [==============================] - 67s 3s/step - loss: 0.7079 - acc:
0.6990 - val_loss: 0.6881 - val_acc: 0.6200
Epoch 19/100
21/21 [==============================] - 68s 3s/step - loss: 0.7108 - acc:
0.6845 - val_loss: 0.8930 - val_acc: 0.6000
Epoch 20/100
```

```
21/21 [==============================] - 72s 3s/step - loss: 0.6758 - acc:
0.7330 - val_loss: 0.8015 - val_acc: 0.6000
Epoch 21/100
21/21 [==============================] - 67s 3s/step - loss: 0.7053 - acc:
0.6748 - val_loss: 0.7972 - val_acc: 0.7200
Epoch 22/100
21/21 [==============================] - 64s 3s/step - loss: 0.7212 - acc:
0.6845 - val_loss: 0.6422 - val_acc: 0.6800
Epoch 23/100
21/21 [==============================] - 64s 3s/step - loss: 0.7102 - acc:
0.7184 - val_loss: 0.7059 - val_acc: 0.6600
Epoch 24/100
21/21 [==============================] - 64s 3s/step - loss: 0.6430 - acc:
0.7136 - val_loss: 0.6389 - val_acc: 0.7200
Epoch 25/100
21/21 [==============================] - 64s 3s/step - loss: 0.6289 - acc:
0.7233 - val_loss: 0.7491 - val_acc: 0.6000
Epoch 26/100
21/21 [==============================] - 63s 3s/step - loss: 0.6824 - acc:
0.7136 - val_loss: 0.7847 - val_acc: 0.6200
Epoch 27/100
21/21 [==============================] - 64s 3s/step - loss: 0.6452 - acc:
0.7136 - val_loss: 0.7972 - val_acc: 0.5800
Epoch 28/100
21/21 [==============================] - 63s 3s/step - loss: 0.6968 - acc:
0.7330 - val_loss: 0.8064 - val_acc: 0.6800
Epoch 29/100
21/21 [==============================] - 63s 3s/step - loss: 0.6546 - acc:
0.7282 - val_loss: 0.8066 - val_acc: 0.6400
Epoch 30/100
21/21 [==============================] - 63s 3s/step - loss: 0.6388 - acc:
0.7330 - val_loss: 0.6779 - val_acc: 0.6000
Epoch 31/100
21/21 [==============================] - 63s 3s/step - loss: 0.6296 - acc:
0.7330 - val_loss: 0.6667 - val_acc: 0.7000
Epoch 32/100
21/21 [==============================] - 63s 3s/step - loss: 0.6013 - acc:
0.7427 - val_loss: 0.5956 - val_acc: 0.6600
Epoch 33/100
21/21 [==============================] - 63s 3s/step - loss: 0.6152 - acc:
0.7476 - val_loss: 0.7639 - val_acc: 0.6200
Epoch 34/100
21/21 [==============================] - 63s 3s/step - loss: 0.6560 - acc:
0.7136 - val_loss: 0.9225 - val_acc: 0.5800
Epoch 35/100
21/21 [==============================] - 63s 3s/step - loss: 0.5997 - acc:
0.7767 - val_loss: 0.8162 - val_acc: 0.6800
Epoch 36/100
```

```
21/21 [==============================] - 63s 3s/step - loss: 0.5723 - acc:
0.7524 - val_loss: 0.7213 - val_acc: 0.6800
Epoch 37/100
21/21 [==============================] - 63s 3s/step - loss: 0.6523 - acc:
0.7039 - val_loss: 0.7317 - val_acc: 0.7400
Epoch 38/100
21/21 [==============================] - 64s 3s/step - loss: 0.6422 - acc:
0.7282 - val_loss: 0.6942 - val_acc: 0.6800
Epoch 39/100
21/21 [==============================] - 68s 3s/step - loss: 0.6045 - acc:
0.7621 - val_loss: 0.7990 - val_acc: 0.6000
Epoch 40/100
21/21 [==============================] - 69s 3s/step - loss: 0.6027 - acc:
0.7330 - val_loss: 0.7935 - val_acc: 0.6400
Epoch 41/100
21/21 [==============================] - 74s 4s/step - loss: 0.5507 - acc:
0.7913 - val_loss: 0.7630 - val_acc: 0.6600
Epoch 42/100
21/21 [==============================] - 67s 3s/step - loss: 0.6141 - acc:
0.7233 - val_loss: 0.7112 - val_acc: 0.7000
Epoch 43/100
21/21 [==============================] - 77s 4s/step - loss: 0.5431 - acc:
0.7718 - val_loss: 0.7325 - val_acc: 0.7000
Epoch 44/100
21/21 [==============================] - 69s 3s/step - loss: 0.5976 - acc:
0.7476 - val_loss: 0.7057 - val_acc: 0.6400
Epoch 45/100
21/21 [==============================] - 77s 4s/step - loss: 0.5342 - acc:
0.7621 - val_loss: 0.7234 - val_acc: 0.7000
Epoch 46/100
21/21 [==============================] - 74s 4s/step - loss: 0.5404 - acc:
0.7864 - val_loss: 0.5719 - val_acc: 0.7000
Epoch 47/100
21/21 [==============================] - 66s 3s/step - loss: 0.5525 - acc:
0.7670 - val_loss: 0.8004 - val_acc: 0.6400
Epoch 48/100
21/21 [==============================] - 66s 3s/step - loss: 0.6155 - acc:
0.7573 - val_loss: 0.6358 - val_acc: 0.7200
Epoch 49/100
21/21 [==============================] - 70s 3s/step - loss: 0.6035 - acc:
0.7233 - val_loss: 0.9068 - val_acc: 0.5400
Epoch 50/100
21/21 [==============================] - 72s 3s/step - loss: 0.5507 - acc:
0.7670 - val_loss: 0.7026 - val_acc: 0.7000
Epoch 51/100
21/21 [==============================] - 68s 3s/step - loss: 0.5346 - acc:
0.7718 - val_loss: 0.7747 - val_acc: 0.6200
Epoch 52/100
```

```
21/21 [==============================] - 67s 3s/step - loss: 0.5631 - acc:
0.7667 - val_loss: 0.6148 - val_acc: 0.6800
Epoch 53/100
21/21 [==============================] - 73s 3s/step - loss: 0.5081 - acc:
0.7816 - val_loss: 0.8564 - val_acc: 0.6600
Epoch 54/100
21/21 [==============================] - 76s 4s/step - loss: 0.5436 - acc:
0.7913 - val_loss: 0.7360 - val_acc: 0.6800
Epoch 55/100
21/21 [==============================] - 68s 3s/step - loss: 0.5652 - acc:
0.7476 - val_loss: 0.8324 - val_acc: 0.6800
Epoch 56/100
21/21 [==============================] - 73s 3s/step - loss: 0.5786 - acc:
0.7427 - val_loss: 0.8425 - val_acc: 0.6400
Epoch 57/100
21/21 [==============================] - 68s 3s/step - loss: 0.5638 - acc:
0.7282 - val_loss: 0.5901 - val_acc: 0.7600
Epoch 58/100
21/21 [==============================] - 67s 3s/step - loss: 0.5494 - acc:
0.7476 - val_loss: 0.5829 - val_acc: 0.7600
Epoch 59/100
21/21 [==============================] - 67s 3s/step - loss: 0.5598 - acc:
0.7573 - val_loss: 0.6610 - val_acc: 0.6000
Epoch 60/100
21/21 [==============================] - 67s 3s/step - loss: 0.5162 - acc:
0.7573 - val_loss: 0.6419 - val_acc: 0.6400
Epoch 61/100
21/21 [==============================] - 65s 3s/step - loss: 0.5650 - acc:
0.7282 - val_loss: 0.5991 - val_acc: 0.7600
Epoch 62/100
21/21 [==============================] - 64s 3s/step - loss: 0.5455 - acc:
0.7670 - val_loss: 0.5701 - val_acc: 0.7400
Epoch 63/100
21/21 [==============================] - 65s 3s/step - loss: 0.5706 - acc:
0.7427 - val_loss: 0.6488 - val_acc: 0.7000
Epoch 64/100
21/21 [==============================] - 72s 3s/step - loss: 0.5495 - acc:
0.7767 - val_loss: 0.6517 - val_acc: 0.7200
Epoch 65/100
21/21 [==============================] - 66s 3s/step - loss: 0.5267 - acc:
0.7864 - val_loss: 0.6769 - val_acc: 0.7000
Epoch 66/100
21/21 [==============================] - 76s 4s/step - loss: 0.5111 - acc:
0.7857 - val_loss: 0.7348 - val_acc: 0.6600
Epoch 67/100
21/21 [==============================] - 68s 3s/step - loss: 0.4734 - acc:
0.8252 - val_loss: 0.5517 - val_acc: 0.7400
Epoch 68/100
```

```
21/21 [==============================] - 65s 3s/step - loss: 0.5352 - acc:
0.7961 - val_loss: 0.8462 - val_acc: 0.6800
Epoch 69/100
21/21 [==============================] - 66s 3s/step - loss: 0.5727 - acc:
0.7670 - val_loss: 0.6732 - val_acc: 0.7600
Epoch 70/100
21/21 [==============================] - 64s 3s/step - loss: 0.5697 - acc:
0.7670 - val_loss: 0.6238 - val_acc: 0.6600
Epoch 71/100
21/21 [==============================] - 70s 3s/step - loss: 0.5606 - acc:
0.7670 - val_loss: 0.8438 - val_acc: 0.6000
Epoch 72/100
21/21 [==============================] - 68s 3s/step - loss: 0.5461 - acc:
0.7718 - val_loss: 0.6238 - val_acc: 0.7400
Epoch 73/100
21/21 [==============================] - 74s 4s/step - loss: 0.5965 - acc:
0.7476 - val_loss: 0.4831 - val_acc: 0.7600
Epoch 74/100
21/21 [==============================] - 65s 3s/step - loss: 0.4972 - acc:
0.7816 - val_loss: 0.6426 - val_acc: 0.7200
Epoch 75/100
21/21 [==============================] - 64s 3s/step - loss: 0.5378 - acc:
0.8107 - val_loss: 0.7613 - val_acc: 0.6800
Epoch 76/100
21/21 [==============================] - 65s 3s/step - loss: 0.5068 - acc:
0.8058 - val_loss: 0.8575 - val_acc: 0.6600
Epoch 77/100
21/21 [==============================] - 65s 3s/step - loss: 0.5610 - acc:
0.7476 - val_loss: 0.7318 - val_acc: 0.7200
Epoch 78/100
21/21 [==============================] - 64s 3s/step - loss: 0.5256 - acc:
0.7670 - val_loss: 0.6099 - val_acc: 0.7200
Epoch 79/100
21/21 [==============================] - 67s 3s/step - loss: 0.4928 - acc:
0.8204 - val_loss: 0.5245 - val_acc: 0.7400
Epoch 80/100
21/21 [==============================] - 68s 3s/step - loss: 0.5503 - acc:
0.7524 - val_loss: 0.6653 - val_acc: 0.6200
Epoch 81/100
21/21 [==============================] - 66s 3s/step - loss: 0.5619 - acc:
0.7573 - val_loss: 0.6939 - val_acc: 0.6400
Epoch 82/100
21/21 [==============================] - 67s 3s/step - loss: 0.5407 - acc:
0.7379 - val_loss: 0.6574 - val_acc: 0.6600
Epoch 83/100
21/21 [==============================] - 65s 3s/step - loss: 0.4767 - acc:
0.7913 - val_loss: 0.7607 - val_acc: 0.5400
Epoch 84/100
```

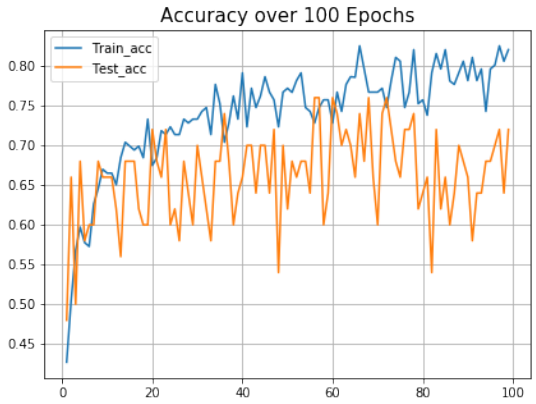
```
21/21 [==============================] - 63s 3s/step - loss: 0.4603 - acc:
0.8155 - val_loss: 0.6997 - val_acc: 0.7200
Epoch 85/100
21/21 [==============================] - 63s 3s/step - loss: 0.4556 - acc:
0.7961 - val_loss: 0.7430 - val_acc: 0.6200
Epoch 86/100
21/21 [==============================] - 63s 3s/step - loss: 0.5102 - acc:
0.8204 - val_loss: 0.7158 - val_acc: 0.6600
Epoch 87/100
21/21 [==============================] - 65s 3s/step - loss: 0.5002 - acc:
0.7810 - val_loss: 0.6548 - val_acc: 0.6000
Epoch 88/100
21/21 [==============================] - 63s 3s/step - loss: 0.4876 - acc:
0.7767 - val_loss: 0.7578 - val_acc: 0.6400
Epoch 89/100
21/21 [==============================] - 63s 3s/step - loss: 0.5028 - acc:
0.7913 - val_loss: 0.7843 - val_acc: 0.7000
Epoch 90/100
21/21 [==============================] - 63s 3s/step - loss: 0.4556 - acc:
0.8058 - val_loss: 0.8094 - val_acc: 0.6800
Epoch 91/100
21/21 [==============================] - 63s 3s/step - loss: 0.5060 - acc:
0.7816 - val_loss: 0.7303 - val_acc: 0.6600
Epoch 92/100
21/21 [==============================] - 63s 3s/step - loss: 0.4337 - acc:
0.8107 - val_loss: 0.8861 - val_acc: 0.5800
Epoch 93/100
21/21 [==============================] - 62s 3s/step - loss: 0.4943 - acc:
0.7816 - val_loss: 0.8048 - val_acc: 0.6400
Epoch 94/100
21/21 [==============================] - 63s 3s/step - loss: 0.4923 - acc:
0.7961 - val_loss: 1.0071 - val_acc: 0.6400
Epoch 95/100
21/21 [==============================] - 63s 3s/step - loss: 0.5634 - acc:
0.7427 - val_loss: 0.7142 - val_acc: 0.6800
Epoch 96/100
21/21 [==============================] - 63s 3s/step - loss: 0.5167 - acc:
0.7961 - val_loss: 0.6059 - val_acc: 0.6800
Epoch 97/100
21/21 [==============================] - 63s 3s/step - loss: 0.4929 - acc:
0.8010 - val_loss: 0.5782 - val_acc: 0.7000
Epoch 98/100
21/21 [==============================] - 64s 3s/step - loss: 0.4356 - acc:
0.8252 - val_loss: 0.5561 - val_acc: 0.7200
Epoch 99/100
21/21 [==============================] - 63s 3s/step - loss: 0.4297 - acc:
0.8058 - val_loss: 0.6671 - val_acc: 0.6400
Epoch 100/100
```

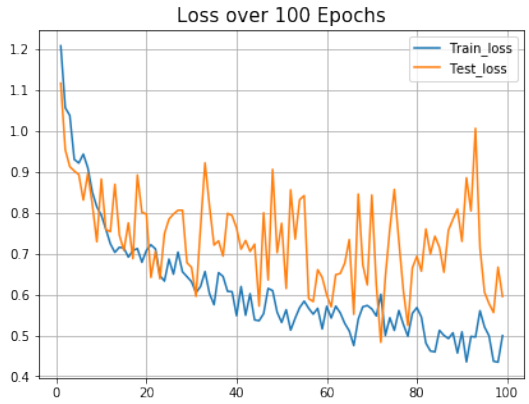
```
21/21 [==============================] - 64s 3s/step - loss: 0.4960 - acc:
0.8204 - val_loss: 0.5952 - val_acc: 0.7200
```

[5 points] Plot Accuracy and Loss During Training

[7]:
```
import matplotlib.pyplot as plt

raise NotImplementedError("Plot the accuracy and the loss during training")
```





Testing Model

[10]:
```
test_datagen = ImageDataGenerator(rescale=1. / 255)

eval_generator = test_datagen.
 ↪flow_from_directory(TEST_DIR,target_size=IMAGE_SIZE,
                                                    ␣
 ↪batch_size=1,shuffle=True,seed=42,class_mode="categorical")
eval_generator.reset()
print(len(eval_generator))
x = model.evaluate_generator(eval_generator,steps = np.ceil(len(eval_generator)),
                             use_multiprocessing = False,verbose = 1,workers=1)
print('Test loss:' , x[0])
print('Test accuracy:',x[1])
```

```
Found 36 images belonging to 4 classes.
36
WARNING:tensorflow:sample_weight modes were coerced from
  ...
    to
  ['...']
36/36 [==============================] - 10s 274ms/step - loss: 0.7570 - acc:
0.6944
```

```
Test loss: 0.757028494571235
Test accuracy: 0.6944444
```

2.4 [10 points] TSNE Plot

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a widely used technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets. After training is complete, extract features from a specific deep layer of your choice, use t-SNE to reduce the dimensionality of your extracted features to 2 dimensions and plot the resulting 2D features.

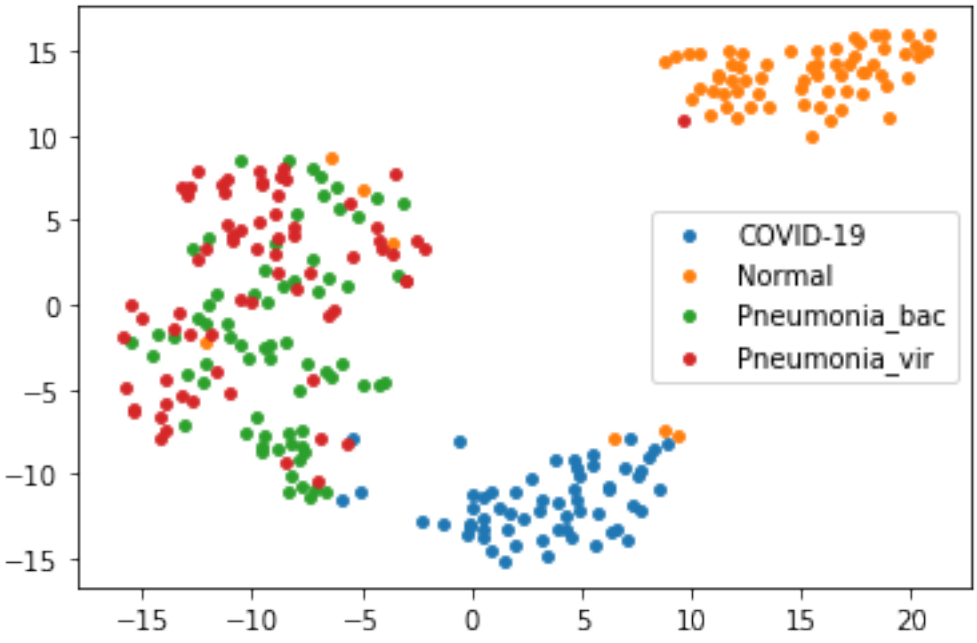
[15]:
```
from sklearn.manifold import TSNE

intermediate_layer_model = models.Model(inputs=model.input,
                                        outputs=model.get_layer('feature_dense').
 ↪output)

tsne_eval_generator = test_datagen.
 ↪flow_from_directory(DATASET_PATH,target_size=IMAGE_SIZE,
                                                    ␣
 ↪batch_size=1,shuffle=True,seed=42,class_mode="categorical")

raise NotImplementedError("Extract features from the tsne_data_generator and fit␣
 ↪a t-SNE model for the features,"
                          "and plot the resulting 2D features of the four␣
 ↪classes.")
```

```
Found 270 images belonging to 4 classes.
{'covid': 0, 'normal': 1, 'pneumonia_bac': 2, 'pneumonia_vir': 3}
Extracting features for 270 images.
270/270 [==============================] - 71s 265ms/step
Training TSNE model.
```


15
10
5
0
−5
−10
−15
−15
−10
−5
0
5
10
15
20
COVID-19
Normal
Pneumonia_bac
Pneumonia_vir