

# Classify\_imgs

October 8, 2020

```
[1]: from tensorflow import keras
from imutils import paths
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import SGD
import numpy as np
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D
from tensorflow.keras.applications.densenet import DenseNet121
from tensorflow.keras.models import Model
```

```
[2]: width, height = 224, 224
```

```
[3]: Densebase = DenseNet121(include_top=False, input_shape=(
    width, height, 3), weights='imagenet', classes=2)

x = Densebase.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
predictions = Dense(2, activation='sigmoid')(x)
model = Model(inputs=Densebase.input,
              outputs=predictions)

model.summary()
```

\_relu[0][0]

```
-----
conv5_block4_1_bn (BatchNormali (None, 7, 7, 128)    512
conv5_block4_1_conv[0][0]
```

```
-----
conv5_block4_1_relu (Activation (None, 7, 7, 128)    0
conv5_block4_1_bn[0][0]
```

```
-----
conv5_block4_2_conv (Conv2D)      (None, 7, 7, 32)    36864
conv5_block4_1_relu[0][0]
```

```

conv5_block4_concat (Concatenat (None, 7, 7, 640)    0
conv5_block3_concat[0][0]
conv5_block4_2_conv[0][0]
-----

conv5_block5_0_bn (BatchNormali (None, 7, 7, 640)    2560
conv5_block4_concat[0][0]
-----

conv5_block5_0_relu (Activation (None, 7, 7, 640)    0
conv5_block5_0_bn[0][0]
-----

conv5_block5_1_conv (Conv2D)      (None, 7, 7, 128)    81920
conv5_block5_0_relu[0][0]
-----

conv5_block5_1_bn (BatchNormali (None, 7, 7, 128)    512
conv5_block5_1_conv[0][0]
-----

conv5_block5_1_relu (Activation (None, 7, 7, 128)    0
conv5_block5_1_bn[0][0]
-----

conv5_block5_2_conv (Conv2D)      (None, 7, 7, 32)     36864
conv5_block5_1_relu[0][0]
-----

conv5_block5_concat (Concatenat (None, 7, 7, 672)    0
conv5_block4_concat[0][0]
conv5_block5_2_conv[0][0]
-----

conv5_block6_0_bn (BatchNormali (None, 7, 7, 672)    2688
conv5_block5_concat[0][0]
-----

conv5_block6_0_relu (Activation (None, 7, 7, 672)    0
conv5_block6_0_bn[0][0]
-----

conv5_block6_1_conv (Conv2D)      (None, 7, 7, 128)    86016
conv5_block6_0_relu[0][0]
-----

conv5_block6_1_bn (BatchNormali (None, 7, 7, 128)    512
conv5_block6_1_conv[0][0]

```

```

-----
conv5_block6_1_relu (Activation (None, 7, 7, 128)    0
conv5_block6_1_bn[0][0]
-----
conv5_block6_2_conv (Conv2D)      (None, 7, 7, 32)    36864
conv5_block6_1_relu[0][0]
-----
conv5_block6_concat (Concatenat (None, 7, 7, 704)    0
conv5_block5_concat[0][0]
conv5_block6_2_conv[0][0]
-----
conv5_block7_0_bn (BatchNormali (None, 7, 7, 704)    2816
conv5_block6_concat[0][0]
-----
conv5_block7_0_relu (Activation (None, 7, 7, 704)    0
conv5_block7_0_bn[0][0]
-----
conv5_block7_1_conv (Conv2D)      (None, 7, 7, 128)    90112
conv5_block7_0_relu[0][0]
-----
conv5_block7_1_bn (BatchNormali (None, 7, 7, 128)    512
conv5_block7_1_conv[0][0]
-----
conv5_block7_1_relu (Activation (None, 7, 7, 128)    0
conv5_block7_1_bn[0][0]
-----
conv5_block7_2_conv (Conv2D)      (None, 7, 7, 32)    36864
conv5_block7_1_relu[0][0]
-----
conv5_block7_concat (Concatenat (None, 7, 7, 736)    0
conv5_block6_concat[0][0]
conv5_block7_2_conv[0][0]
-----
conv5_block8_0_bn (BatchNormali (None, 7, 7, 736)    2944
conv5_block7_concat[0][0]
-----

```

```

conv5_block8_0_relu (Activation (None, 7, 7, 736)    0
conv5_block8_0_bn[0][0]
-----

conv5_block8_1_conv (Conv2D)      (None, 7, 7, 128)    94208
conv5_block8_0_relu[0][0]
-----

conv5_block8_1_bn (BatchNormaliz (None, 7, 7, 128)    512
conv5_block8_1_conv[0][0]
-----

conv5_block8_1_relu (Activation (None, 7, 7, 128)    0
conv5_block8_1_bn[0][0]
-----

conv5_block8_2_conv (Conv2D)      (None, 7, 7, 32)     36864
conv5_block8_1_relu[0][0]
-----

conv5_block8_concat (Concatenat (None, 7, 7, 768)    0
conv5_block7_concat[0][0]
conv5_block8_2_conv[0][0]
-----

conv5_block9_0_bn (BatchNormaliz (None, 7, 7, 768)    3072
conv5_block8_concat[0][0]
-----

conv5_block9_0_relu (Activation (None, 7, 7, 768)    0
conv5_block9_0_bn[0][0]
-----

conv5_block9_1_conv (Conv2D)      (None, 7, 7, 128)    98304
conv5_block9_0_relu[0][0]
-----

conv5_block9_1_bn (BatchNormaliz (None, 7, 7, 128)    512
conv5_block9_1_conv[0][0]
-----

conv5_block9_1_relu (Activation (None, 7, 7, 128)    0
conv5_block9_1_bn[0][0]
-----

conv5_block9_2_conv (Conv2D)      (None, 7, 7, 32)     36864
conv5_block9_1_relu[0][0]
-----

```

```

-----
conv5_block9_concat (Concatenat (None, 7, 7, 800)    0
conv5_block8_concat[0][0]
conv5_block9_2_conv[0][0]
-----

-----
conv5_block10_0_bn (BatchNormal (None, 7, 7, 800)    3200
conv5_block9_concat[0][0]
-----

-----
conv5_block10_0_relu (Activatio (None, 7, 7, 800)    0
conv5_block10_0_bn[0][0]
-----

-----
conv5_block10_1_conv (Conv2D)    (None, 7, 7, 128)    102400
conv5_block10_0_relu[0][0]
-----

-----
conv5_block10_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block10_1_conv[0][0]
-----

-----
conv5_block10_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block10_1_bn[0][0]
-----

-----
conv5_block10_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block10_1_relu[0][0]
-----

-----
conv5_block10_concat (Concatena (None, 7, 7, 832)    0
conv5_block9_concat[0][0]
conv5_block10_2_conv[0][0]
-----

-----
conv5_block11_0_bn (BatchNormal (None, 7, 7, 832)    3328
conv5_block10_concat[0][0]
-----

-----
conv5_block11_0_relu (Activatio (None, 7, 7, 832)    0
conv5_block11_0_bn[0][0]
-----

-----
conv5_block11_1_conv (Conv2D)    (None, 7, 7, 128)    106496
conv5_block11_0_relu[0][0]
-----

-----
conv5_block11_1_bn (BatchNormal (None, 7, 7, 128)    512

```

```

conv5_block11_1_conv[0][0]
-----
-----
conv5_block11_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block11_1_bn[0][0]
-----
-----
conv5_block11_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block11_1_relu[0][0]
-----
-----
conv5_block11_concat (Concatena (None, 7, 7, 864)    0
conv5_block10_concat[0][0]
conv5_block11_2_conv[0][0]
-----
-----
conv5_block12_0_bn (BatchNormal (None, 7, 7, 864)    3456
conv5_block11_concat[0][0]
-----
-----
conv5_block12_0_relu (Activatio (None, 7, 7, 864)    0
conv5_block12_0_bn[0][0]
-----
-----
conv5_block12_1_conv (Conv2D)    (None, 7, 7, 128)    110592
conv5_block12_0_relu[0][0]
-----
-----
conv5_block12_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block12_1_conv[0][0]
-----
-----
conv5_block12_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block12_1_bn[0][0]
-----
-----
conv5_block12_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block12_1_relu[0][0]
-----
-----
conv5_block12_concat (Concatena (None, 7, 7, 896)    0
conv5_block11_concat[0][0]
conv5_block12_2_conv[0][0]
-----
-----
conv5_block13_0_bn (BatchNormal (None, 7, 7, 896)    3584
conv5_block12_concat[0][0]
-----

```

```

-----
conv5_block13_0_relu (Activatio (None, 7, 7, 896)    0
conv5_block13_0_bn[0][0]

-----

conv5_block13_1_conv (Conv2D)    (None, 7, 7, 128)    114688
conv5_block13_0_relu[0][0]

-----

conv5_block13_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block13_1_conv[0][0]

-----

conv5_block13_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block13_1_bn[0][0]

-----

conv5_block13_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block13_1_relu[0][0]

-----

conv5_block13_concat (Concatena (None, 7, 7, 928)    0
conv5_block12_concat[0][0]
conv5_block13_2_conv[0][0]

-----

conv5_block14_0_bn (BatchNormal (None, 7, 7, 928)    3712
conv5_block13_concat[0][0]

-----

conv5_block14_0_relu (Activatio (None, 7, 7, 928)    0
conv5_block14_0_bn[0][0]

-----

conv5_block14_1_conv (Conv2D)    (None, 7, 7, 128)    118784
conv5_block14_0_relu[0][0]

-----

conv5_block14_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block14_1_conv[0][0]

-----

conv5_block14_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block14_1_bn[0][0]

-----

conv5_block14_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block14_1_relu[0][0]

```

```

-----
-----
conv5_block14_concat (Concatena (None, 7, 7, 960)    0
conv5_block13_concat[0][0]
conv5_block14_2_conv[0][0]
-----
-----
conv5_block15_0_bn (BatchNormal (None, 7, 7, 960)    3840
conv5_block14_concat[0][0]
-----
-----
conv5_block15_0_relu (Activatio (None, 7, 7, 960)    0
conv5_block15_0_bn[0][0]
-----
-----
conv5_block15_1_conv (Conv2D)    (None, 7, 7, 128)    122880
conv5_block15_0_relu[0][0]
-----
-----
conv5_block15_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block15_1_conv[0][0]
-----
-----
conv5_block15_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block15_1_bn[0][0]
-----
-----
conv5_block15_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block15_1_relu[0][0]
-----
-----
conv5_block15_concat (Concatena (None, 7, 7, 992)    0
conv5_block14_concat[0][0]
conv5_block15_2_conv[0][0]
-----
-----
conv5_block16_0_bn (BatchNormal (None, 7, 7, 992)    3968
conv5_block15_concat[0][0]
-----
-----
conv5_block16_0_relu (Activatio (None, 7, 7, 992)    0
conv5_block16_0_bn[0][0]
-----
-----
conv5_block16_1_conv (Conv2D)    (None, 7, 7, 128)    126976
conv5_block16_0_relu[0][0]
-----
-----

```



```

conv5_block16_1_bn (BatchNormal (None, 7, 7, 128)    512
conv5_block16_1_conv[0][0]
-----
conv5_block16_1_relu (Activatio (None, 7, 7, 128)    0
conv5_block16_1_bn[0][0]
-----
conv5_block16_2_conv (Conv2D)    (None, 7, 7, 32)    36864
conv5_block16_1_relu[0][0]
-----
conv5_block16_concat (Concatena (None, 7, 7, 1024)    0
conv5_block15_concat[0][0]
conv5_block16_2_conv[0][0]
-----
bn (BatchNormalization)          (None, 7, 7, 1024)    4096
conv5_block16_concat[0][0]
-----
relu (Activation)                 (None, 7, 7, 1024)    0          bn[0][0]
-----
global_average_pooling2d (Globa (None, 1024)          0          relu[0][0]
-----
dense (Dense)                     (None, 1024)          1049600
global_average_pooling2d[0][0]
-----
dense_1 (Dense)                   (None, 2)             2050          dense[0][0]
=====
=====
Total params: 8,089,154
Trainable params: 8,005,506
Non-trainable params: 83,648
-----
-----

```

```

[4]: # initialize the number of training epochs and batch size
NUM_EPOCHS = 50
BS = 16
TRAIN_PATH = '../dados/'
# determine the total number of image paths in training, validation,
# and testing directories
totalTrain = len(list(paths.list_images(TRAIN_PATH)))

```

```
[5]: # initialize the training data augmentation object
trainAug = ImageDataGenerator(
    rescale=1 / 255.0,
    rotation_range=20,
    zoom_range=0.05,
    width_shift_range=0.05,
    height_shift_range=0.05,
    shear_range=0.05,
    horizontal_flip=True,
    validation_split=0.1)
```

```
[6]: # initialize the testing data augmentation object
testAug = ImageDataGenerator(rescale=1 / 255.0, validation_split=0.1)
```

```
[7]: # initialize the training generator
trainGen = trainAug.flow_from_directory(
    TRAIN_PATH,
    class_mode="categorical",
    target_size=(height, width),
    color_mode="rgb",
    shuffle=True,
    seed=123,
    batch_size=BS,
    subset='training')
```

Found 3200 images belonging to 2 classes.

```
[8]: # initialize the testing generator
testGen = testAug.flow_from_directory(
    TRAIN_PATH,
    class_mode="categorical",
    target_size=(height, width),
    color_mode="rgb",
    shuffle=False,
    batch_size=BS,
    subset='validation')
```

Found 355 images belonging to 2 classes.

```
[9]: opt = SGD(lr=1e-1, momentum=0.9, decay=1e-1 / NUM_EPOCHS)
model.compile(loss="binary_crossentropy",
              optimizer=opt,
              metrics=["accuracy",
                      keras.metrics.AUC(),
                      keras.metrics.Precision(),
                      keras.metrics.Recall()])
```

```
[10]: from PIL import Image, ImageFile
ImageFile.LOAD_TRUNCATED_IMAGES = True

# train our Keras model
H = model.fit(
    trainGen,
    validation_data=testGen,
    epochs=NUM_EPOCHS)
```

Epoch 1/50

200/200 [=====] - 312s 2s/step - loss: 0.3264 - accuracy: 0.9003 - auc: 0.9526 - precision: 0.9005 - recall: 0.9050 - val\_loss: 193.1002 - val\_accuracy: 0.5380 - val\_auc: 0.5380 - val\_precision: 0.5380 - val\_recall: 0.5380

Epoch 2/50

200/200 [=====] - 301s 2s/step - loss: 0.2081 - accuracy: 0.9388 - auc: 0.9767 - precision: 0.9384 - recall: 0.9384 - val\_loss: 0.1827 - val\_accuracy: 0.9324 - val\_auc: 0.9812 - val\_precision: 0.9324 - val\_recall: 0.9324

Epoch 3/50

200/200 [=====] - 300s 2s/step - loss: 0.1699 - accuracy: 0.9459 - auc: 0.9823 - precision: 0.9453 - recall: 0.9453 - val\_loss: 0.2016 - val\_accuracy: 0.9352 - val\_auc: 0.9754 - val\_precision: 0.9352 - val\_recall: 0.9352

Epoch 4/50

200/200 [=====] - 300s 2s/step - loss: 0.1458 - accuracy: 0.9506 - auc: 0.9864 - precision: 0.9509 - recall: 0.9506 - val\_loss: 0.2126 - val\_accuracy: 0.9099 - val\_auc: 0.9755 - val\_precision: 0.9099 - val\_recall: 0.9099

Epoch 5/50

200/200 [=====] - 299s 1s/step - loss: 0.1438 - accuracy: 0.9513 - auc: 0.9866 - precision: 0.9513 - recall: 0.9513 - val\_loss: 0.1866 - val\_accuracy: 0.9183 - val\_auc: 0.9786 - val\_precision: 0.9183 - val\_recall: 0.9183

Epoch 6/50

200/200 [=====] - 301s 2s/step - loss: 0.1406 - accuracy: 0.9516 - auc: 0.9870 - precision: 0.9519 - recall: 0.9516 - val\_loss: 0.1518 - val\_accuracy: 0.9408 - val\_auc: 0.9851 - val\_precision: 0.9435 - val\_recall: 0.9408

Epoch 7/50

200/200 [=====] - 304s 2s/step - loss: 0.1229 - accuracy: 0.9563 - auc: 0.9896 - precision: 0.9559 - recall: 0.9553 - val\_loss: 0.1347 - val\_accuracy: 0.9465 - val\_auc: 0.9889 - val\_precision: 0.9465 - val\_recall: 0.9465

Epoch 8/50

200/200 [=====] - 325s 2s/step - loss: 0.1149 - accuracy: 0.9581 - auc: 0.9911 - precision: 0.9581 - recall: 0.9581 - val\_loss:

0.1446 - val\_accuracy: 0.9380 - val\_auc: 0.9871 - val\_precision: 0.9356 -  
val\_recall: 0.9408  
Epoch 9/50  
200/200 [=====] - 326s 2s/step - loss: 0.1081 -  
accuracy: 0.9584 - auc: 0.9926 - precision: 0.9584 - recall: 0.9584 - val\_loss:  
0.1539 - val\_accuracy: 0.9437 - val\_auc: 0.9872 - val\_precision: 0.9463 -  
val\_recall: 0.9437  
Epoch 10/50  
200/200 [=====] - 329s 2s/step - loss: 0.1064 -  
accuracy: 0.9631 - auc: 0.9926 - precision: 0.9631 - recall: 0.9631 - val\_loss:  
0.1721 - val\_accuracy: 0.9380 - val\_auc: 0.9832 - val\_precision: 0.9379 -  
val\_recall: 0.9352  
Epoch 11/50  
200/200 [=====] - 321s 2s/step - loss: 0.1086 -  
accuracy: 0.9606 - auc: 0.9919 - precision: 0.9612 - recall: 0.9606 - val\_loss:  
0.1070 - val\_accuracy: 0.9465 - val\_auc: 0.9942 - val\_precision: 0.9412 -  
val\_recall: 0.9465  
Epoch 12/50  
200/200 [=====] - 317s 2s/step - loss: 0.0995 -  
accuracy: 0.9647 - auc: 0.9934 - precision: 0.9647 - recall: 0.9644 - val\_loss:  
0.1383 - val\_accuracy: 0.9493 - val\_auc: 0.9893 - val\_precision: 0.9493 -  
val\_recall: 0.9493  
Epoch 13/50  
200/200 [=====] - 302s 2s/step - loss: 0.0928 -  
accuracy: 0.9666 - auc: 0.9941 - precision: 0.9669 - recall: 0.9666 - val\_loss:  
0.1426 - val\_accuracy: 0.9408 - val\_auc: 0.9883 - val\_precision: 0.9435 -  
val\_recall: 0.9408  
Epoch 14/50  
200/200 [=====] - 298s 1s/step - loss: 0.0935 -  
accuracy: 0.9666 - auc: 0.9942 - precision: 0.9669 - recall: 0.9663 - val\_loss:  
0.0987 - val\_accuracy: 0.9634 - val\_auc: 0.9953 - val\_precision: 0.9634 -  
val\_recall: 0.9634  
Epoch 15/50  
200/200 [=====] - 294s 1s/step - loss: 0.0853 -  
accuracy: 0.9703 - auc: 0.9951 - precision: 0.9703 - recall: 0.9703 - val\_loss:  
0.1085 - val\_accuracy: 0.9549 - val\_auc: 0.9913 - val\_precision: 0.9549 -  
val\_recall: 0.9549  
Epoch 16/50  
200/200 [=====] - 298s 1s/step - loss: 0.0837 -  
accuracy: 0.9712 - auc: 0.9948 - precision: 0.9719 - recall: 0.9712 - val\_loss:  
0.1044 - val\_accuracy: 0.9606 - val\_auc: 0.9939 - val\_precision: 0.9606 -  
val\_recall: 0.9606  
Epoch 17/50  
200/200 [=====] - 331s 2s/step - loss: 0.0871 -  
accuracy: 0.9684 - auc: 0.9948 - precision: 0.9681 - recall: 0.9681 - val\_loss:  
0.1357 - val\_accuracy: 0.9549 - val\_auc: 0.9886 - val\_precision: 0.9549 -  
val\_recall: 0.9549  
Epoch 18/50

200/200 [=====] - 300s 1s/step - loss: 0.0834 -  
accuracy: 0.9703 - auc: 0.9953 - precision: 0.9703 - recall: 0.9697 - val\_loss:  
0.1805 - val\_accuracy: 0.9155 - val\_auc: 0.9834 - val\_precision: 0.9155 -  
val\_recall: 0.9155  
Epoch 19/50  
200/200 [=====] - 294s 1s/step - loss: 0.0869 -  
accuracy: 0.9691 - auc: 0.9949 - precision: 0.9691 - recall: 0.9691 - val\_loss:  
0.1037 - val\_accuracy: 0.9549 - val\_auc: 0.9938 - val\_precision: 0.9549 -  
val\_recall: 0.9549  
Epoch 20/50  
200/200 [=====] - 298s 1s/step - loss: 0.0822 -  
accuracy: 0.9681 - auc: 0.9954 - precision: 0.9678 - recall: 0.9681 - val\_loss:  
0.1258 - val\_accuracy: 0.9352 - val\_auc: 0.9908 - val\_precision: 0.9352 -  
val\_recall: 0.9352  
Epoch 21/50  
200/200 [=====] - 296s 1s/step - loss: 0.0820 -  
accuracy: 0.9669 - auc: 0.9955 - precision: 0.9666 - recall: 0.9669 - val\_loss:  
0.1000 - val\_accuracy: 0.9549 - val\_auc: 0.9947 - val\_precision: 0.9549 -  
val\_recall: 0.9549  
Epoch 22/50  
200/200 [=====] - 286s 1s/step - loss: 0.0782 -  
accuracy: 0.9728 - auc: 0.9959 - precision: 0.9731 - recall: 0.9725 - val\_loss:  
0.1165 - val\_accuracy: 0.9493 - val\_auc: 0.9924 - val\_precision: 0.9493 -  
val\_recall: 0.9493  
Epoch 23/50  
200/200 [=====] - 285s 1s/step - loss: 0.0763 -  
accuracy: 0.9728 - auc: 0.9961 - precision: 0.9728 - recall: 0.9725 - val\_loss:  
0.1077 - val\_accuracy: 0.9549 - val\_auc: 0.9931 - val\_precision: 0.9576 -  
val\_recall: 0.9549  
Epoch 24/50  
200/200 [=====] - 284s 1s/step - loss: 0.0669 -  
accuracy: 0.9747 - auc: 0.9973 - precision: 0.9747 - recall: 0.9747 - val\_loss:  
0.1289 - val\_accuracy: 0.9408 - val\_auc: 0.9906 - val\_precision: 0.9408 -  
val\_recall: 0.9408  
Epoch 25/50  
200/200 [=====] - 285s 1s/step - loss: 0.0679 -  
accuracy: 0.9747 - auc: 0.9962 - precision: 0.9747 - recall: 0.9747 - val\_loss:  
0.1372 - val\_accuracy: 0.9408 - val\_auc: 0.9899 - val\_precision: 0.9407 -  
val\_recall: 0.9380  
Epoch 26/50  
200/200 [=====] - 288s 1s/step - loss: 0.0727 -  
accuracy: 0.9716 - auc: 0.9964 - precision: 0.9716 - recall: 0.9716 - val\_loss:  
0.1060 - val\_accuracy: 0.9577 - val\_auc: 0.9937 - val\_precision: 0.9577 -  
val\_recall: 0.9577  
Epoch 27/50  
200/200 [=====] - 299s 1s/step - loss: 0.0675 -  
accuracy: 0.9744 - auc: 0.9967 - precision: 0.9747 - recall: 0.9741 - val\_loss:  
0.1171 - val\_accuracy: 0.9606 - val\_auc: 0.9921 - val\_precision: 0.9606 -

val\_recall: 0.9606  
Epoch 28/50  
200/200 [=====] - 294s 1s/step - loss: 0.0639 -  
accuracy: 0.9766 - auc: 0.9970 - precision: 0.9766 - recall: 0.9766 - val\_loss:  
0.0949 - val\_accuracy: 0.9606 - val\_auc: 0.9947 - val\_precision: 0.9606 -  
val\_recall: 0.9606  
Epoch 29/50  
200/200 [=====] - 296s 1s/step - loss: 0.0598 -  
accuracy: 0.9766 - auc: 0.9975 - precision: 0.9763 - recall: 0.9766 - val\_loss:  
0.1348 - val\_accuracy: 0.9437 - val\_auc: 0.9896 - val\_precision: 0.9437 -  
val\_recall: 0.9437  
Epoch 30/50  
200/200 [=====] - 295s 1s/step - loss: 0.0676 -  
accuracy: 0.9734 - auc: 0.9965 - precision: 0.9737 - recall: 0.9731 - val\_loss:  
0.1259 - val\_accuracy: 0.9549 - val\_auc: 0.9916 - val\_precision: 0.9549 -  
val\_recall: 0.9549  
Epoch 31/50  
200/200 [=====] - 297s 1s/step - loss: 0.0623 -  
accuracy: 0.9744 - auc: 0.9972 - precision: 0.9744 - recall: 0.9741 - val\_loss:  
0.1228 - val\_accuracy: 0.9521 - val\_auc: 0.9885 - val\_precision: 0.9521 -  
val\_recall: 0.9521  
Epoch 32/50  
200/200 [=====] - 298s 1s/step - loss: 0.0609 -  
accuracy: 0.9775 - auc: 0.9976 - precision: 0.9769 - recall: 0.9775 - val\_loss:  
0.1211 - val\_accuracy: 0.9493 - val\_auc: 0.9912 - val\_precision: 0.9493 -  
val\_recall: 0.9493  
Epoch 33/50  
200/200 [=====] - 299s 1s/step - loss: 0.0584 -  
accuracy: 0.9744 - auc: 0.9978 - precision: 0.9741 - recall: 0.9744 - val\_loss:  
0.1028 - val\_accuracy: 0.9577 - val\_auc: 0.9949 - val\_precision: 0.9577 -  
val\_recall: 0.9577  
Epoch 34/50  
200/200 [=====] - 302s 2s/step - loss: 0.0596 -  
accuracy: 0.9791 - auc: 0.9978 - precision: 0.9791 - recall: 0.9787 - val\_loss:  
0.1174 - val\_accuracy: 0.9493 - val\_auc: 0.9922 - val\_precision: 0.9492 -  
val\_recall: 0.9465  
Epoch 35/50  
200/200 [=====] - 298s 1s/step - loss: 0.0583 -  
accuracy: 0.9800 - auc: 0.9975 - precision: 0.9800 - recall: 0.9803 - val\_loss:  
0.1267 - val\_accuracy: 0.9465 - val\_auc: 0.9914 - val\_precision: 0.9465 -  
val\_recall: 0.9465  
Epoch 36/50  
200/200 [=====] - 302s 2s/step - loss: 0.0568 -  
accuracy: 0.9772 - auc: 0.9976 - precision: 0.9775 - recall: 0.9769 - val\_loss:  
0.1158 - val\_accuracy: 0.9549 - val\_auc: 0.9905 - val\_precision: 0.9551 -  
val\_recall: 0.9577  
Epoch 37/50  
200/200 [=====] - 305s 2s/step - loss: 0.0568 -

accuracy: 0.9809 - auc: 0.9977 - precision: 0.9809 - recall: 0.9809 - val\_loss:  
 0.1122 - val\_accuracy: 0.9577 - val\_auc: 0.9928 - val\_precision: 0.9577 -  
 val\_recall: 0.9577  
 Epoch 38/50  
 200/200 [=====] - 307s 2s/step - loss: 0.0560 -  
 accuracy: 0.9787 - auc: 0.9981 - precision: 0.9784 - recall: 0.9787 - val\_loss:  
 0.1131 - val\_accuracy: 0.9465 - val\_auc: 0.9935 - val\_precision: 0.9465 -  
 val\_recall: 0.9465  
 Epoch 39/50  
 200/200 [=====] - 296s 1s/step - loss: 0.0560 -  
 accuracy: 0.9778 - auc: 0.9981 - precision: 0.9778 - recall: 0.9778 - val\_loss:  
 0.1005 - val\_accuracy: 0.9493 - val\_auc: 0.9942 - val\_precision: 0.9493 -  
 val\_recall: 0.9493  
 Epoch 40/50  
 200/200 [=====] - 300s 1s/step - loss: 0.0531 -  
 accuracy: 0.9800 - auc: 0.9981 - precision: 0.9797 - recall: 0.9800 - val\_loss:  
 0.1399 - val\_accuracy: 0.9380 - val\_auc: 0.9896 - val\_precision: 0.9380 -  
 val\_recall: 0.9380  
 Epoch 41/50  
 200/200 [=====] - 305s 2s/step - loss: 0.0570 -  
 accuracy: 0.9762 - auc: 0.9978 - precision: 0.9763 - recall: 0.9766 - val\_loss:  
 0.1328 - val\_accuracy: 0.9408 - val\_auc: 0.9912 - val\_precision: 0.9408 -  
 val\_recall: 0.9408  
 Epoch 42/50  
 200/200 [=====] - 299s 1s/step - loss: 0.0536 -  
 accuracy: 0.9803 - auc: 0.9980 - precision: 0.9803 - recall: 0.9806 - val\_loss:  
 0.1192 - val\_accuracy: 0.9577 - val\_auc: 0.9943 - val\_precision: 0.9577 -  
 val\_recall: 0.9577  
 Epoch 43/50  
 200/200 [=====] - 303s 2s/step - loss: 0.0506 -  
 accuracy: 0.9822 - auc: 0.9982 - precision: 0.9822 - recall: 0.9825 - val\_loss:  
 0.1153 - val\_accuracy: 0.9493 - val\_auc: 0.9931 - val\_precision: 0.9493 -  
 val\_recall: 0.9493  
 Epoch 44/50  
 200/200 [=====] - 303s 2s/step - loss: 0.0486 -  
 accuracy: 0.9806 - auc: 0.9986 - precision: 0.9809 - recall: 0.9806 - val\_loss:  
 0.1160 - val\_accuracy: 0.9493 - val\_auc: 0.9935 - val\_precision: 0.9493 -  
 val\_recall: 0.9493  
 Epoch 45/50  
 200/200 [=====] - 299s 1s/step - loss: 0.0448 -  
 accuracy: 0.9834 - auc: 0.9988 - precision: 0.9834 - recall: 0.9834 - val\_loss:  
 0.1169 - val\_accuracy: 0.9521 - val\_auc: 0.9936 - val\_precision: 0.9521 -  
 val\_recall: 0.9521  
 Epoch 46/50  
 200/200 [=====] - 307s 2s/step - loss: 0.0519 -  
 accuracy: 0.9791 - auc: 0.9981 - precision: 0.9791 - recall: 0.9791 - val\_loss:  
 0.1065 - val\_accuracy: 0.9577 - val\_auc: 0.9942 - val\_precision: 0.9577 -  
 val\_recall: 0.9577

```

Epoch 47/50
200/200 [=====] - 302s 2s/step - loss: 0.0446 -
accuracy: 0.9822 - auc: 0.9987 - precision: 0.9825 - recall: 0.9825 - val_loss:
0.1198 - val_accuracy: 0.9521 - val_auc: 0.9930 - val_precision: 0.9521 -
val_recall: 0.9521
Epoch 48/50
200/200 [=====] - 289s 1s/step - loss: 0.0411 -
accuracy: 0.9841 - auc: 0.9990 - precision: 0.9841 - recall: 0.9841 - val_loss:
0.1297 - val_accuracy: 0.9465 - val_auc: 0.9928 - val_precision: 0.9465 -
val_recall: 0.9465
Epoch 49/50
200/200 [=====] - 286s 1s/step - loss: 0.0454 -
accuracy: 0.9822 - auc: 0.9985 - precision: 0.9822 - recall: 0.9825 - val_loss:
0.1202 - val_accuracy: 0.9437 - val_auc: 0.9930 - val_precision: 0.9437 -
val_recall: 0.9437
Epoch 50/50
200/200 [=====] - 285s 1s/step - loss: 0.0361 -
accuracy: 0.9866 - auc: 0.9992 - precision: 0.9869 - recall: 0.9866 - val_loss:
0.1256 - val_accuracy: 0.9493 - val_auc: 0.9929 - val_precision: 0.9493 -
val_recall: 0.9493

```

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

```

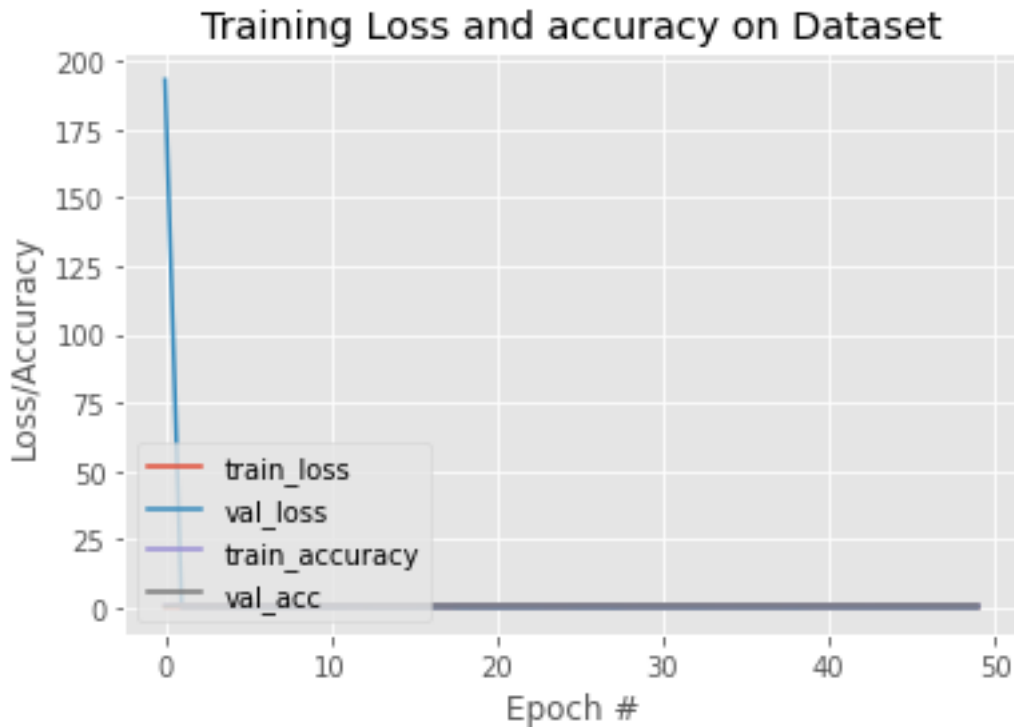
N = NUM_EPOCHS
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")

plt.plot(np.arange(0, N), H.history["accuracy"], label="train_accuracy")
plt.plot(np.arange(0, N), H.history["val_accuracy"], label="val_acc")
plt.title("Training Loss and accuracy on Dataset")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig('Training Loss and accuracy on Dataset')
H.history.keys()

```

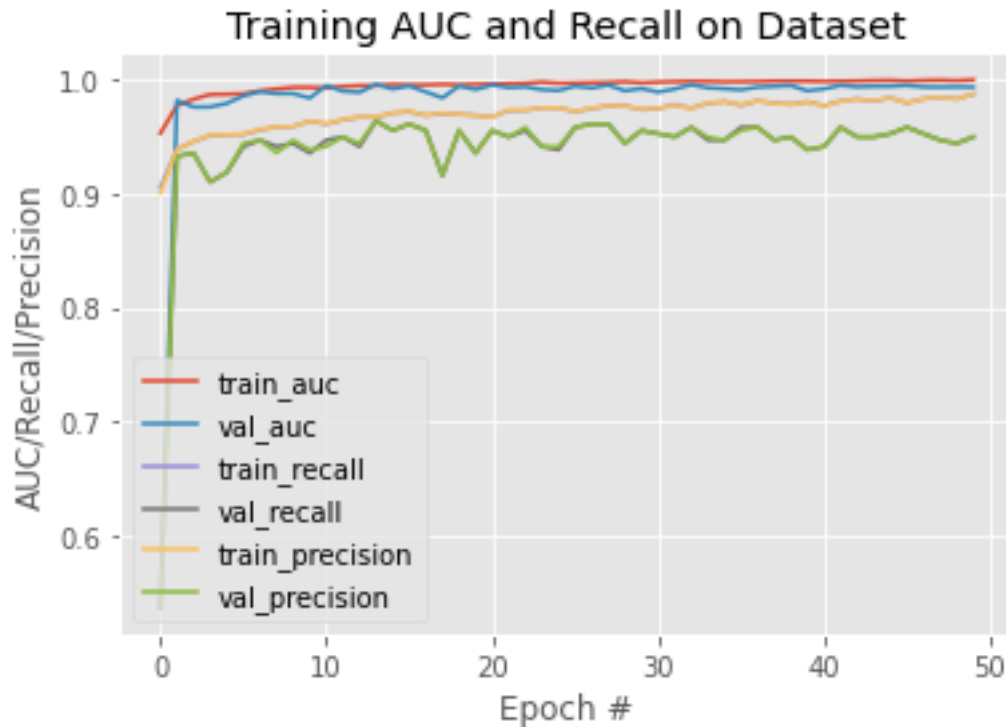
```
[11]: dict_keys(['loss', 'accuracy', 'auc', 'precision', 'recall', 'val_loss',
'val_accuracy', 'val_auc', 'val_precision', 'val_recall'])
```





```
[12]: plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, N), H.history["auc"], label="train_auc")
plt.plot(np.arange(0, N), H.history["val_auc"], label="val_auc")
plt.plot(np.arange(0, N), H.history["recall"], label="train_recall")
plt.plot(np.arange(0, N), H.history["val_recall"], label="val_recall")
plt.plot(np.arange(0, N), H.history["precision"], label="train_precision")
plt.plot(np.arange(0, N), H.history["val_precision"], label="val_precision")

plt.title("Training AUC and Recall on Dataset")
plt.xlabel("Epoch #")
plt.ylabel("AUC/Recall/Precision")
plt.legend(loc="lower left")
plt.savefig('Training AUC, Recall and Precision on Dataset')
```



```
[13]: from sklearn.metrics import classification_report
      from sklearn.metrics import confusion_matrix
      import pandas as pd
      import seaborn as sns

      testGen.reset()
      predIdxs = model.predict(testGen, batch_size=BS)

      # for each image in the testing set we need to find the index of the
      # label with corresponding largest predicted probability
      predIdxs = np.argmax(predIdxs, axis=1)

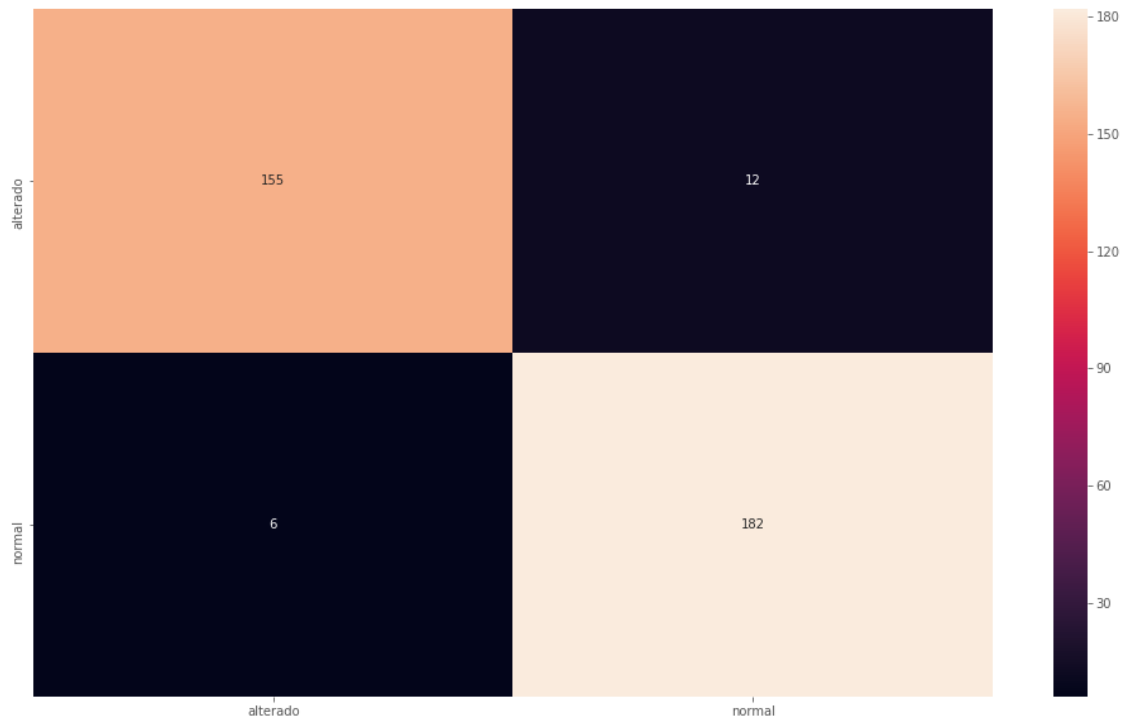
      conf_mat = confusion_matrix(testGen.classes, predIdxs)

      class_names = ['alterado', 'normal']
      fig = plt.figure(figsize=(17,10))
      df_cm = pd.DataFrame(conf_mat, index=class_names, columns=class_names)
      heatmap = sns.heatmap(df_cm, annot=True, fmt='d')
      heatmap

      # show a nicely formatted classification report
      print(classification_report(testGen.classes, predIdxs,
```

```
target_names=testGen.class_indices.keys()))
```

	precision	recall	f1-score	support
rx-alterado-anonim	0.96	0.93	0.95	167
rx-normal-anonim	0.94	0.97	0.95	188
accuracy			0.95	355
macro avg	0.95	0.95	0.95	355
weighted avg	0.95	0.95	0.95	355



```
[14]: model.save('Models/DenseNet121_H{}W{}.h5'.format(height, width))
```