

```
import tensorflow as tf
import numpy as np
import pandas as pd
from sklearn.metrics import classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
import os

INFO:tensorflow:Enabling eager execution
INFO:tensorflow:Enabling v2 tensorshape
INFO:tensorflow:Enabling resource variables
INFO:tensorflow:Enabling tensor equality
INFO:tensorflow:Enabling control flow v2

In [12]: epochs = 10 # quantidade de vezes a ser executado o algoritmo, uma epoch é quanto toda
batch = 32 # número de amostras que será carregado a cada execução

In [3]: #carrega o modelo da InceptionResNetV2 com os pesos aprendidos no treino da Inception
base_model = tf.keras.applications.Inception(weights='imagenet', include_top=False)

In [4]: # O restante do modelo e suas camadas são discutidos a seguir
# x recebe o final da InceptionResNetV2
x=base_model.output

In [5]: #Nova configuração para o modelo

#adiciona apos x uma camada GlobalMaxPooling2D e atribui este no x novamente (logo:
x=tf.keras.layers.GlobalMaxPooling2D()(x)

#adiciona apos x uma camada densa com 128 neurônios com funcao de ativacao relu. Atrib
x=tf.keras.layers.Dense(128,activation='relu')(x)

#adiciona apos x uma camada densa com 64 neurônios com funcao de ativacao relu. Atrib
x=tf.keras.layers.Dense(64,activation='relu')(x)

#adiciona apos x uma camda densa com 32 neurônios com funcao de ativacao relu. Atrib
x=tf.keras.layers.Dense(32,activation='relu')(x)

#adiciona apos x os neurônios que devem ser utilizados, nesse caso foram desligados 2
x=tf.keras.layers.Dropout(0.5)(x)

#adiciona apos x uma camada densa com 7 neurônios (sete classes) com funcao de ativa
#preds=tf.keras.layers.Dense(3,activation='softmax')(x)
preds=tf.keras.layers.Dense(3,activation='sigmoid')(x)

#definindo modelo final
model=tf.keras.models.Model(inputs=base_model.input,outputs=preds)

#mostrando modelo final e sua estrutura
model.summary()

Model: "model"

Layer (type) Output Shape Param # Connected to
-----
input_1 (InputLayer) (None, None, None, 0
block1_conv1 (Conv2D) (None, None, None, 3 864 input_1[0][0]
block1_conv1_bn (BatchNormaliz (None, None, None, 3 128 block1_conv1[0][0]
block1_conv1_act (Activation) (None, None, None, 3 0 block1_conv1_bn[0][0]
block1_conv2 (Conv2D) (None, None, None, 6 18432 block1_conv1_act[0]
[0]
block1_conv2_bn (BatchNormaliz (None, None, None, 6 256 block1_conv2[0][0]
block1_conv2_act (Activation) (None, None, None, 6 0 block1_conv2_bn[0][0]
block2_sepconv1 (SeparableConv2 (None, None, None, 1 8768 block1_conv2_act[0]
[0]
block2_sepconv1_bn (BatchNormal (None, None, None, 1 512 block2_sepconv1[0][0]
block2_sepconv2_act (Activation) (None, None, None, 1 0 block2_sepconv1_bn[0]
[0]
block2_sepconv2 (SeparableConv2 (None, None, None, 1 17536 block2_sepconv2_act
[0][0]
block2_sepconv2_bn (BatchNormal (None, None, None, 1 512 block2_sepconv2[0][0]
conv2d (Conv2D) (None, None, None, 1 8192 block1_conv2_act[0]
[0]
block2_pool (MaxPooling2D) (None, None, None, 1 0 block2_sepconv2_bn[0]
[0]
batch_normalization (BatchNorma (None, None, None, 1 512 conv2d[0][0]
add (Add) (None, None, None, 1 0 block2_pool[0][0]
[0][0] batch_normalization
block3_sepconv1_act (Activation) (None, None, None, 1 0 add[0][0]
block3_sepconv1 (SeparableConv2 (None, None, None, 2 33920 block3_sepconv1_act
[0][0]
block3_sepconv1_bn (BatchNormal (None, None, None, 2 1024 block3_sepconv1[0][0]
block3_sepconv2_act (Activation) (None, None, None, 2 0 block3_sepconv1_bn[0]
[0]
block3_sepconv2 (SeparableConv2 (None, None, None, 2 67840 block3_sepconv2_act
[0][0]
block3_sepconv2_bn (BatchNormal (None, None, None, 2 1024 block3_sepconv2[0][0]
conv2d_1 (Conv2D) (None, None, None, 2 32768 add[0][0]
block3_pool (MaxPooling2D) (None, None, None, 2 0 block3_sepconv2_bn[0]
[0]
batch_normalization_1 (BatchNor (None, None, None, 2 1024 conv2d_1[0][0]
add_1 (Add) (None, None, None, 2 0 block3_pool[0][0]
[0][0] batch_normalization_1
block4_sepconv1_act (Activation) (None, None, None, 2 0 add_1[0][0]
block4_sepconv1 (SeparableConv2 (None, None, None, 7 188672 block4_sepconv1_act
[0][0]
block4_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block4_sepconv1[0][0]
block4_sepconv2_act (Activation) (None, None, None, 7 0 block4_sepconv1_bn[0]
[0]
block4_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block4_sepconv2_act
[0][0]
block4_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block4_sepconv2[0][0]
conv2d_2 (Conv2D) (None, None, None, 7 186368 add_1[0][0]
block4_pool (MaxPooling2D) (None, None, None, 7 0 block4_sepconv2_bn[0]
[0]
batch_normalization_2 (BatchNor (None, None, None, 7 2912 conv2d_2[0][0]
add_2 (Add) (None, None, None, 7 0 block4_pool[0][0]
[0][0] batch_normalization_2
block5_sepconv1_act (Activation) (None, None, None, 7 0 add_2[0][0]
block5_sepconv1 (SeparableConv2 (None, None, None, 7 536536 block5_sepconv1_act
[0][0]
block5_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block5_sepconv1[0][0]
block5_sepconv2_act (Activation) (None, None, None, 7 0 block5_sepconv1_bn[0]
[0]
block5_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block5_sepconv2_act
[0][0]
block5_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block5_sepconv2[0][0]
block5_sepconv3_act (Activation) (None, None, None, 7 0 block5_sepconv2_bn[0]
[0]
block5_sepconv3 (SeparableConv2 (None, None, None, 7 536536 block5_sepconv3_act
[0][0]
block5_sepconv3_bn (BatchNormal (None, None, None, 7 2912 block5_sepconv3[0][0]
add_3 (Add) (None, None, None, 7 0 block5_sepconv3_bn[0]
[0] add_2[0][0]
block6_sepconv1_act (Activation) (None, None, None, 7 0 add_3[0][0]
block6_sepconv1 (SeparableConv2 (None, None, None, 7 536536 block6_sepconv1_act
[0][0]
block6_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block6_sepconv1[0][0]
block6_sepconv2_act (Activation) (None, None, None, 7 0 block6_sepconv1_bn[0]
[0]
block6_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block6_sepconv2_act
[0][0]
block6_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block6_sepconv2[0][0]
block6_sepconv3_act (Activation) (None, None, None, 7 0 block6_sepconv2_bn[0]
[0]
block6_sepconv3 (SeparableConv2 (None, None, None, 7 536536 block6_sepconv3_act
[0][0]
block6_sepconv3_bn (BatchNormal (None, None, None, 7 2912 block6_sepconv3[0][0]
add_4 (Add) (None, None, None, 7 0 block6_sepconv3_bn[0]
[0] add_3[0][0]
block7_sepconv1_act (Activation) (None, None, None, 7 0 add_4[0][0]
block7_sepconv1 (SeparableConv2 (None, None, None, 7 536536 block7_sepconv1_act
[0][0]
block7_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block7_sepconv1[0][0]
block7_sepconv2_act (Activation) (None, None, None, 7 0 block7_sepconv1_bn[0]
[0]
block7_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block7_sepconv2_act
[0][0]
block7_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block7_sepconv2[0][0]
block7_sepconv3_act (Activation) (None, None, None, 7 0 block7_sepconv2_bn[0]
[0]
block7_sepconv3 (SeparableConv2 (None, None, None, 7 536536 block7_sepconv3_act
[0][0]
block7_sepconv3_bn (BatchNormal (None, None, None, 7 2912 block7_sepconv3[0][0]
add_5 (Add) (None, None, None, 7 0 block7_sepconv3_bn[0]
[0] add_4[0][0]
block8_sepconv1_act (Activation) (None, None, None, 7 0 add_5[0][0]
block8_sepconv1 (SeparableConv2 (None, None, None, 7 536536 block8_sepconv1_act
[0][0]
block8_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block8_sepconv1[0][0]
block8_sepconv2_act (Activation) (None, None, None, 7 0 block8_sepconv1_bn[0]
[0]
block8_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block8_sepconv2_act
[0][0]
block8_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block8_sepconv2[0][0]
block8_sepconv3_act (Activation) (None, None, None, 7 0 block8_sepconv2_bn[0]
[0]
block8_sepconv3 (SeparableConv2 (None, None, None, 7 536536 block8_sepconv3_act
[0][0]
block8_sepconv3_bn (BatchNormal (None, None, None, 7 2912 block8_sepconv3[0][0]
add_6 (Add) (None, None, None, 7 0 block8_sepconv3_bn[0]
[0] add_5[0][0]
block9_sepconv1_act (Activation) (None, None, None, 7 0 add_6[0][0]
block9_sepconv1 (SeparableConv2 (None, None, None, 7 536536 block9_sepconv1_act
[0][0]
block9_sepconv1_bn (BatchNormal (None, None, None, 7 2912 block9_sepconv1[0][0]
block9_sepconv2_act (Activation) (None, None, None, 7 0 block9_sepconv1_bn[0]
[0]
block9_sepconv2 (SeparableConv2 (None, None, None, 7 536536 block9_sepconv2_act
[0][0]
block9_sepconv2_bn (BatchNormal (None, None, None, 7 2912 block9_sepconv2[0][0]
block9_sepconv3_act (Activation) (None, None, None, 7 0 block9_sepconv2_bn[0]
[0]
block9_sepconv3 (SeparableConv2 (None, None, None, 7 536536 block9_sepconv3_act
[0][0]
block9_sepconv3_bn (BatchNormal (None, None, None, 7 2912 block9_sepconv3[0][0]
add_7 (Add) (None, None, None, 7 0 block9_sepconv3_bn[0]
[0] add_6[0][0]
block10_sepconv1_act (Activatio (None, None, None, 7 0 add_7[0][0]
block10_sepconv1 (SeparableConv (None, None, None, 7 536536 block10_sepconv1_act
[0][0]
block10_sepconv1_bn (BatchNorma (None, None, None, 7 2912 block10_sepconv1[0]
[0]
block10_sepconv2_act (Activatio (None, None, None, 7 0 block10_sepconv1_bn
[0][0]
block10_sepconv2 (SeparableConv (None, None, None, 7 536536 block10_sepconv2_act
[0][0]
block10_sepconv2_bn (BatchNorma (None, None, None, 7 2912 block10_sepconv2[0]
[0]
block10_sepconv3_act (Activatio (None, None, None, 7 0 block10_sepconv2_bn
[0][0]
block10_sepconv3 (SeparableConv (None, None, None, 7 536536 block10_sepconv3_act
[0][0]
block10_sepconv3_bn (BatchNorma (None, None, None, 7 2912 block10_sepconv3[0]
[0]
add_8 (Add) (None, None, None, 7 0 block10_sepconv3_bn[0]
[0] add_7[0][0]
block11_sepconv1_act (Activatio (None, None, None, 7 0 add_8[0][0]
block11_sepconv1 (SeparableConv (None, None, None, 7 536536 block11_sepconv1_act
[0][0]
block11_sepconv1_bn (BatchNorma (None, None, None, 7 2912 block11_sepconv1[0]
[0]
block11_sepconv2_act (Activatio (None, None, None, 7 0 block11_sepconv1_bn
[0][0]
block11_sepconv2 (SeparableConv (None, None, None, 7 536536 block11_sepconv2_act
[0][0]
block11_sepconv2_bn (BatchNorma (None, None, None, 7 2912 block11_sepconv2[0]
[0]
block11_sepconv3_act (Activatio (None, None, None, 7 0 block11_sepconv2_bn
[0][0]
block11_sepconv3 (SeparableConv (None, None, None, 7 536536 block11_sepconv3_act
[0][0]
block11_sepconv3_bn (BatchNorma (None, None, None, 7 2912 block11_sepconv3[0]
[0]
add_9 (Add) (None, None, None, 7 0 block11_sepconv3_bn
[0][0] add_8[0][0]
block12_sepconv1_act (Activatio (None, None, None, 7 0 add_9[0][0]
block12_sepconv1 (SeparableConv (None, None, None, 7 536536 block12_sepconv1_act
[0][0]
block12_sepconv1_bn (BatchNorma (None, None, None, 7 2912 block12_sepconv1[0]
[0]
block12_sepconv2_act (Activatio (None, None, None, 7 0 block12_sepconv1_bn
[0][0]
block12_sepconv2 (SeparableConv (None, None, None, 7 536536 block12_sepconv2_act
[0][0]
block12_sepconv2_bn (BatchNorma (None, None, None, 7 2912 block12_sepconv2[0]
[0]
block12_sepconv3_act (Activatio (None, None, None, 7 0 block12_sepconv2_bn
[0][0]
block12_sepconv3 (SeparableConv (None, None, None, 7 536536 block12_sepconv3_act
[0][0]
block12_sepconv3_bn (BatchNorma (None, None, None, 7 2912 block12_sepconv3[0]
[0]
add_10 (Add) (None, None, None, 7 0 block12_sepconv3_bn
[0][0] add_9[0][0]
block13_sepconv1_act (Activatio (None, None, None, 7 0 add_10[0][0]
block13_sepconv1 (SeparableConv (None, None, None, 7 536536 block13_sepconv1_act
[0][0]
block13_sepconv1_bn (BatchNorma (None, None, None, 7 2912 block13_sepconv1[0]
[0]
block13_sepconv2_act (Activatio (None, None, None, 7 0 block13_sepconv1_bn
[0][0]
block13_sepconv2 (SeparableConv (None, None, None, 1 752024 block13_sepconv2_act
[0][0]
block13_sepconv2_bn (BatchNorma (None, None, None, 1 4096 block13_sepconv2[0]
[0]
conv2d_3 (Conv2D) (None, None, None, 1 745472 add_10[0][0]
block13_pool (MaxPooling2D) (None, None, None, 1 0 block13_sepconv2_bn
[0][0]
batch_normalization_3 (BatchNor (None, None, None, 1 4096 conv2d_3[0][0]
add_11 (Add) (None, None, None, 1 0 block13_pool[0][0]
[0][0] batch_normalization_3
block14_sepconv1 (SeparableConv (None, None, None, 1 1582080 add_11[0][0]
block14_sepconv1_bn (BatchNorma (None, None, None, 1 6144 block14_sepconv1[0]
[0]
block14_sepconv1_act (Activatio (None, None, None, 1 0 block14_sepconv1_bn
[0][0]
block14_sepconv2 (SeparableConv (None, None, None, 2 3159552 block14_sepconv1_act
[0][0]
block14_sepconv2_bn (BatchNorma (None, None, None, 2 8192 block14_sepconv2[0]
[0]
block14_sepconv2_act (Activatio (None, None, None, 2 0 block14_sepconv2_bn
[0][0]
global_max_pooling2d (GlobalMax (None, 2048) 0 block14_sepconv2_act
[0][0]
dense (Dense) (None, 128) 262272 global_max_pooling2d
[0][0]
dense_1 (Dense) (None, 64) 8256 dense[0][0]
dense_2 (Dense) (None, 32) 2080 dense_1[0][0]
dropout (Dropout) (None, 32) 0 dense_2[0][0]
dense_3 (Dense) (None, 3) 99 dropout[0][0]
=====
Total params: 21,134,187
Trainable params: 21,079,659
Non-trainable params: 54,508

In [6]: #congelando os neurônios já treinados na ImageNet, queremos retrainar somente a últim
for i in range(1, len(model.layers)):
    if i.name.split('/')[-1] != 'dense':
        i.trainable=False
    else:
        i.trainable=True

In [7]: #iniciando objeto que apanhara todas as imagens de treino, processando as imagens com
train_data_gen = tf.keras.preprocessing.image.ImageDataGenerator(preprocessing_functi
#iniciando objeto que apanhara todas as imagens de teste, processando as imagens com
test_data_gen = tf.keras.preprocessing.image.ImageDataGenerator(preprocessing_functio

In [8]: #CARREGANDO PRÓPRIO DATASET PARA USO

#definindo gerador de imagens de treino
train_generator = train_data_gen.flow_from_directory('shapes_split/train',
                                                    target_size=(224, 224), # tamanho da
                                                    batch_size=batch,
                                                    class_mode='categorical',
                                                    shuffle=True)

#definindo gerador de imagens de teste
test_generator = test_data_gen.flow_from_directory('shapes_split/test',
                                                    target_size=(224, 224), # tamanho da
                                                    batch_size=batch,
                                                    class_mode='categorical',
                                                    shuffle=True)

Found 240 images belonging to 3 classes.
Found 60 images belonging to 3 classes.

In [9]: lr = tf.keras.optimizers.Adam(learning_rate=0.001) #estabelecendo taxa de otimização
model.compile(optimizer=lr, loss='categorical_crossentropy', metrics=['accuracy'])

In [10]: #defineco dos steps
step_size_train = train_generator.n//train_generator.batch_size
step_size_test = test_generator.n//test_generator.batch_size

In [11]: #treinando e testando o modelo
history = model.fit_generator(generator=train_generator,
                              steps_per_epoch=step_size_train,
                              epochs=epochs,
                              validation_data=test_generator,
                              validation_steps=step_size_test)

Epoch 1/10
c:\Users\vini\cvs\appdata\local\programs\python\python39\lib\site-packages\tensorflow
\python\keras\engine\training.py:1973: UserWarning: Model.fit_generator() is deprecate
d and will be removed in a future version. Please use 'Model.fit', which supports gene
rators.
warnings.warn('Model.fit_generator() is deprecated and '
7/7 ===== - 27s 3s/step - loss: 1.1831 - accuracy: 0.4636 -
val_loss: 0.2945 - val_accuracy: 1.0000
Epoch 2/10
7/7 ===== - 25s 4s/step - loss: 0.4145 - accuracy: 0.7980 -
val_loss: 0.1427 - val_accuracy: 1.0000
Epoch 3/10
7/7 ===== - 26s 4s/step - loss: 0.2659 - accuracy: 0.8815 -
val_loss: 0.0545 - val_accuracy: 1.0000
Epoch 4/10
7/7 ===== - 25s 4s/step - loss: 0.1944 - accuracy: 0.9351 -
val_loss: 0.0127 - val_accuracy: 1.0000
Epoch 5/10
7/7 ===== - 24s 3s/step - loss: 0.1775 - accuracy: 0.9363 -
val_loss: 0.0084 - val_accuracy: 1.0000
Epoch 6/10
7/7 ===== - 24s 3s/step - loss: 0.1642 - accuracy: 0.9237 -
val_loss: 0.0047 - val_accuracy: 1.0000
Epoch 7/10
7/7 ===== - 24s 3s/step - loss: 0.1009 - accuracy: 0.9682 -
val_loss: 0.0034 - val_accuracy: 1.0000
Epoch 8/10
7/7 ===== - 24s 3s/step - loss: 0.1021 - accuracy: 0.9526 -
val_loss: 0.0019 - val_accuracy: 1.0000
Epoch 9/10
7/7 ===== - 24s 3s/step - loss: 0.1231 - accuracy: 0.9180 -
val_loss: 0.0011 - val_accuracy: 1.0000
Epoch 10/10
7/7 ===== - 24s 3s/step - loss: 0.0764 - accuracy: 0.9807 -
val_loss: 0.0015 - val_accuracy: 1.0000

In [12]: #Avaliando o modelo
loss_train, train_acc = model.evaluate_generator(train_generator, steps=step_size_train)
loss_test, test_acc = model.evaluate_generator(test_generator, steps=step_size_test)
print(f'--CLASSIFICATION--')
print(classification_report(test_generator.classes, y_pred, target_names=
matrix = confusion_matrix(test_generator.classes, y_pred)
df_cm = pd.DataFrame(matrix, index = [i for i in range(3)],
plt.figure(figsize = (10,7))
print('-----MATRIX-----')
sns.heatmap(df_cm, annot=True, linewidths=2.5)

-----CLASSIFICATION-----
precision recall f1-score support
circles 0.35 0.35 0.35 20
squares 0.50 0.50 0.50 20
triangles 0.35 0.35 0.35 20
accuracy 0.40 0.40 0.40 60
macro avg 0.40 0.40 0.40 60
weighted avg 0.40 0.40 0.40 60

-----MATRIX-----
<axes:Subplot>

In [13]: #Apresentando resultados em graficos
plt.title('Loss')
plt.plot(history.history('loss'), label='train')
plt.plot(history.history('val_loss'), label='test')
plt.legend()
plt.show()

In [14]: # Criando graficos para visualizacao dos resultados
plt.title('Accuracy')
plt.plot(history.history('accuracy'), label='train')
plt.plot(history.history('val_accuracy'), label='test')
plt.legend()
plt.show()

In [15]: print('Criando classificações..')
labels = os.listdir('shapes_split/test')
print('Rotulos', labels)
#criando estruturas para métricas de avaliação, processo um pouco mais demorado
Y_pred = model.predict_generator(test_generator)
Print('Preds Created')
columns = [i for i in range(3)]
Y_pred = np.argmax(Y_pred, axis=1)
print('Preds ID created')

Criando classificações..
Rotulos ['circles', 'squares', 'triangles']
c:\Users\vini\cvs\appdata\local\programs\python\python39\lib\site-packages\tensorflow
\python\keras\engine\training.py:1973: UserWarning: Model.evaluate_generator() is depre
rts generators.
warnings.warn('Model.evaluate_generator() is deprecated and '
Preds Created
Preds ID created

In [16]: classification = classification_report(test_generator.classes, y_pred, target_names=
print(classification_report(test_generator.classes, y_pred, target_names=
matrix = confusion_matrix(test_generator.classes, y_pred)
df_cm = pd.DataFrame(matrix, index = [i for i in range(3)],
plt.figure(figsize = (10,7))
print('-----MATRIX-----')
sns.heatmap(df_cm, annot=True, linewidths=2.5)

-----CLASSIFICATION-----
precision recall f1-score support
circles 0.35 0.35 0.35 20
squares 0.50 0.50 0.50 20
triangles 0.35 0.35 0.35 20
accuracy 0.40 0.40 0.40 60
macro avg 0.40 0.40 0.40 60
weighted avg 0.40 0.40 0.40 60

-----MATRIX-----
<axes:Subplot>

In [17]:
```