# Primera ejecución del modelo CNN simple

## Cambios vs. Version anterior:

* 1. Se aumenta el número de epochs de 30 a 50 para comprobar si el modelo6 hubiera mejorado con el tiempo un poco más.
  2. Se añade la generación de gráficas, tanto para la evolución de Accuracy como de Loss.

## Estructura:

from tensorflow.keras.preprocessing.image import ImageDataGenerator

from tensorflow.keras import Sequential

from tensorflow.keras.layers import Conv2D, MaxPool2D, Dropout, Flatten, \

Dense

import tensorflow as tf

physical\_devices = tf.config.experimental.list\_physical\_devices('GPU')

tf.config.experimental.set\_memory\_growth(physical\_devices[0], True)

training\_datagen = ImageDataGenerator(rescale=1. / 255,

rotation\_range=0.3,

shear\_range=0.1,

zoom\_range=[0.90, 1.2],

horizontal\_flip=True)

test\_datagen = ImageDataGenerator(rescale=1. / 255)

training\_path = '/home/ruben/workspace/tfg/deep-learning-facial-recognition/data/age/training'

test\_path = '/home/ruben/workspace/tfg/deep-learning-facial-recognition/data/age/test'

training\_set = training\_datagen.flow\_from\_directory(training\_path,

target\_size=(64, 64),

batch\_size=32,

class\_mode='categorical',

shuffle=True,

seed=42)

test\_set = test\_datagen.flow\_from\_directory(test\_path,

target\_size=(64, 64),

batch\_size=32,

class\_mode='categorical',

shuffle=True,

seed=42)

age\_classifier = Sequential()

age\_classifier.add(

Conv2D(filters=32, kernel\_size=(3, 3), input\_shape=(64, 64, 3),

activation='relu'))

age\_classifier.add(MaxPool2D(pool\_size=(2, 2)))

age\_classifier.add(

Conv2D(filters=32, kernel\_size=(3, 3), activation='relu'))

age\_classifier.add(MaxPool2D(pool\_size=(2, 2)))

age\_classifier.add(

Conv2D(filters=64, kernel\_size=(3, 3), activation='relu'))

age\_classifier.add(MaxPool2D(pool\_size=(2, 2)))

age\_classifier.add(Flatten())

age\_classifier.add(Dense(units=128, activation='relu'))

age\_classifier.add(Dropout(0.3))

age\_classifier.add(Dense(units=64, activation='relu'))

age\_classifier.add(Dropout(0.3))

age\_classifier.add(Dense(units=6, activation='softmax'))

age\_classifier.compile(optimizer='adam', loss='categorical\_crossentropy',

metrics=['accuracy'])

hist = age\_classifier.fit(training\_set,

steps\_per\_epoch=(18966 // 32),

epochs=50,

validation\_data=test\_set,

validation\_steps=(4742 // 32))

## Resultado:

Epoch 1/50

2020-05-20 17:54:12.465854: I tensorflow/stream\_executor/platform/default/dso\_loader.cc:44] Successfully opened dynamic library libcublas.so.10

2020-05-20 17:54:12.828058: I tensorflow/stream\_executor/platform/default/dso\_loader.cc:44] Successfully opened dynamic library libcudnn.so.7

592/592 [==============================] - 37s 62ms/step - loss: 1.3639 - accuracy: 0.4453 - val\_loss: 1.1027 - val\_accuracy: 0.5268

Epoch 2/50

592/592 [==============================] - 37s 63ms/step - loss: 1.1080 - accuracy: 0.5257 - val\_loss: 1.0210 - val\_accuracy: 0.5625

Epoch 3/50

592/592 [==============================] - 38s 64ms/step - loss: 1.0242 - accuracy: 0.5587 - val\_loss: 0.9870 - val\_accuracy: 0.5775

Epoch 4/50

592/592 [==============================] - 38s 65ms/step - loss: 0.9791 - accuracy: 0.5793 - val\_loss: 0.9134 - val\_accuracy: 0.6045

Epoch 5/50

592/592 [==============================] - 38s 65ms/step - loss: 0.9436 - accuracy: 0.5911 - val\_loss: 0.9030 - val\_accuracy: 0.6047

Epoch 6/50

592/592 [==============================] - 38s 65ms/step - loss: 0.9204 - accuracy: 0.6026 - val\_loss: 0.8835 - val\_accuracy: 0.6168

Epoch 7/50

592/592 [==============================] - 38s 65ms/step - loss: 0.9044 - accuracy: 0.6080 - val\_loss: 0.8645 - val\_accuracy: 0.6233

Epoch 8/50

592/592 [==============================] - 39s 66ms/step - loss: 0.8859 - accuracy: 0.6145 - val\_loss: 0.8489 - val\_accuracy: 0.6294

Epoch 9/50

592/592 [==============================] - 39s 65ms/step - loss: 0.8643 - accuracy: 0.6226 - val\_loss: 0.8860 - val\_accuracy: 0.6185

Epoch 10/50

592/592 [==============================] - 39s 66ms/step - loss: 0.8573 - accuracy: 0.6256 - val\_loss: 0.8352 - val\_accuracy: 0.6294

Epoch 11/50

592/592 [==============================] - 39s 66ms/step - loss: 0.8501 - accuracy: 0.6281 - val\_loss: 0.8356 - val\_accuracy: 0.6387

Epoch 12/50

592/592 [==============================] - 39s 67ms/step - loss: 0.8386 - accuracy: 0.6355 - val\_loss: 0.8352 - val\_accuracy: 0.6372

Epoch 13/50

592/592 [==============================] - 39s 65ms/step - loss: 0.8242 - accuracy: 0.6424 - val\_loss: 0.8195 - val\_accuracy: 0.6425

Epoch 14/50

592/592 [==============================] - 39s 65ms/step - loss: 0.8239 - accuracy: 0.6392 - val\_loss: 0.8105 - val\_accuracy: 0.6408

Epoch 15/50

592/592 [==============================] - 38s 65ms/step - loss: 0.8195 - accuracy: 0.6395 - val\_loss: 0.8032 - val\_accuracy: 0.6444

Epoch 16/50

592/592 [==============================] - 39s 65ms/step - loss: 0.8014 - accuracy: 0.6495 - val\_loss: 0.8047 - val\_accuracy: 0.6497

Epoch 17/50

592/592 [==============================] - 39s 65ms/step - loss: 0.8000 - accuracy: 0.6479 - val\_loss: 0.8231 - val\_accuracy: 0.6370

Epoch 18/50

592/592 [==============================] - 39s 65ms/step - loss: 0.7998 - accuracy: 0.6447 - val\_loss: 0.8009 - val\_accuracy: 0.6470

Epoch 19/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7846 - accuracy: 0.6515 - val\_loss: 0.8220 - val\_accuracy: 0.6436

Epoch 20/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7812 - accuracy: 0.6543 - val\_loss: 0.8139 - val\_accuracy: 0.6537

Epoch 21/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7719 - accuracy: 0.6560 - val\_loss: 0.8200 - val\_accuracy: 0.6451

Epoch 22/50

592/592 [==============================] - 39s 65ms/step - loss: 0.7810 - accuracy: 0.6587 - val\_loss: 0.8048 - val\_accuracy: 0.6478

Epoch 23/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7688 - accuracy: 0.6600 - val\_loss: 0.7987 - val\_accuracy: 0.6562

Epoch 24/50

592/592 [==============================] - 39s 65ms/step - loss: 0.7595 - accuracy: 0.6650 - val\_loss: 0.7890 - val\_accuracy: 0.6558

Epoch 25/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7622 - accuracy: 0.6617 - val\_loss: 0.7837 - val\_accuracy: 0.6598

Epoch 26/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7484 - accuracy: 0.6675 - val\_loss: 0.7967 - val\_accuracy: 0.6537

Epoch 27/50

592/592 [==============================] - 39s 65ms/step - loss: 0.7501 - accuracy: 0.6682 - val\_loss: 0.7957 - val\_accuracy: 0.6636

Epoch 28/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7497 - accuracy: 0.6683 - val\_loss: 0.8073 - val\_accuracy: 0.6501

Epoch 29/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7403 - accuracy: 0.6738 - val\_loss: 0.7918 - val\_accuracy: 0.6603

Epoch 30/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7367 - accuracy: 0.6734 - val\_loss: 0.7773 - val\_accuracy: 0.6613

Epoch 31/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7467 - accuracy: 0.6727 - val\_loss: 0.7891 - val\_accuracy: 0.6582

Epoch 32/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7372 - accuracy: 0.6740 - val\_loss: 0.8160 - val\_accuracy: 0.6434

Epoch 33/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7298 - accuracy: 0.6776 - val\_loss: 0.7971 - val\_accuracy: 0.6518

Epoch 34/50

592/592 [==============================] - 39s 65ms/step - loss: 0.7225 - accuracy: 0.6813 - val\_loss: 0.8087 - val\_accuracy: 0.6529

Epoch 35/50

592/592 [==============================] - 41s 69ms/step - loss: 0.7223 - accuracy: 0.6804 - val\_loss: 0.8080 - val\_accuracy: 0.6531

Epoch 36/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7239 - accuracy: 0.6790 - val\_loss: 0.8064 - val\_accuracy: 0.6539

Epoch 37/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7179 - accuracy: 0.6829 - val\_loss: 0.7935 - val\_accuracy: 0.6634

Epoch 38/50

592/592 [==============================] - 40s 67ms/step - loss: 0.7194 - accuracy: 0.6837 - val\_loss: 0.8034 - val\_accuracy: 0.6560

Epoch 39/50

592/592 [==============================] - 39s 67ms/step - loss: 0.7190 - accuracy: 0.6790 - val\_loss: 0.7988 - val\_accuracy: 0.6548

Epoch 40/50

592/592 [==============================] - 40s 68ms/step - loss: 0.7173 - accuracy: 0.6837 - val\_loss: 0.8292 - val\_accuracy: 0.6505

Epoch 41/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7133 - accuracy: 0.6840 - val\_loss: 0.8009 - val\_accuracy: 0.6577

Epoch 42/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7055 - accuracy: 0.6863 - val\_loss: 0.8097 - val\_accuracy: 0.6554

Epoch 43/50

592/592 [==============================] - 39s 66ms/step - loss: 0.7091 - accuracy: 0.6841 - val\_loss: 0.7898 - val\_accuracy: 0.6575

Epoch 44/50

592/592 [==============================] - 40s 67ms/step - loss: 0.7049 - accuracy: 0.6896 - val\_loss: 0.8109 - val\_accuracy: 0.6501

Epoch 45/50

592/592 [==============================] - 40s 67ms/step - loss: 0.7088 - accuracy: 0.6918 - val\_loss: 0.7965 - val\_accuracy: 0.6607

Epoch 46/50

592/592 [==============================] - 40s 67ms/step - loss: 0.6975 - accuracy: 0.6913 - val\_loss: 0.8018 - val\_accuracy: 0.6598

Epoch 47/50

592/592 [==============================] - 40s 67ms/step - loss: 0.6971 - accuracy: 0.6931 - val\_loss: 0.7893 - val\_accuracy: 0.6630

Epoch 48/50

592/592 [==============================] - 42s 70ms/step - loss: 0.6953 - accuracy: 0.6890 - val\_loss: 0.8222 - val\_accuracy: 0.6541

Epoch 49/50

592/592 [==============================] - 40s 68ms/step - loss: 0.6896 - accuracy: 0.6924 - val\_loss: 0.8049 - val\_accuracy: 0.6550

Epoch 50/50

592/592 [==============================] - 40s 67ms/step - loss: 0.6944 - accuracy: 0.6908 - val\_loss: 0.8092 - val\_accuracy: 0.6548

Model: "sequential"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

conv2d (Conv2D) (None, 62, 62, 32) 896

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d (MaxPooling2D) (None, 31, 31, 32) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv2d\_1 (Conv2D) (None, 29, 29, 32) 9248

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d\_1 (MaxPooling2 (None, 14, 14, 32) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv2d\_2 (Conv2D) (None, 12, 12, 64) 18496

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling2d\_2 (MaxPooling2 (None, 6, 6, 64) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten (Flatten) (None, 2304) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense (Dense) (None, 128) 295040

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout (Dropout) (None, 128) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_1 (Dense) (None, 64) 8256

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_1 (Dropout) (None, 64) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_2 (Dense) (None, 6) 390

=================================================================

Total params: 332,326

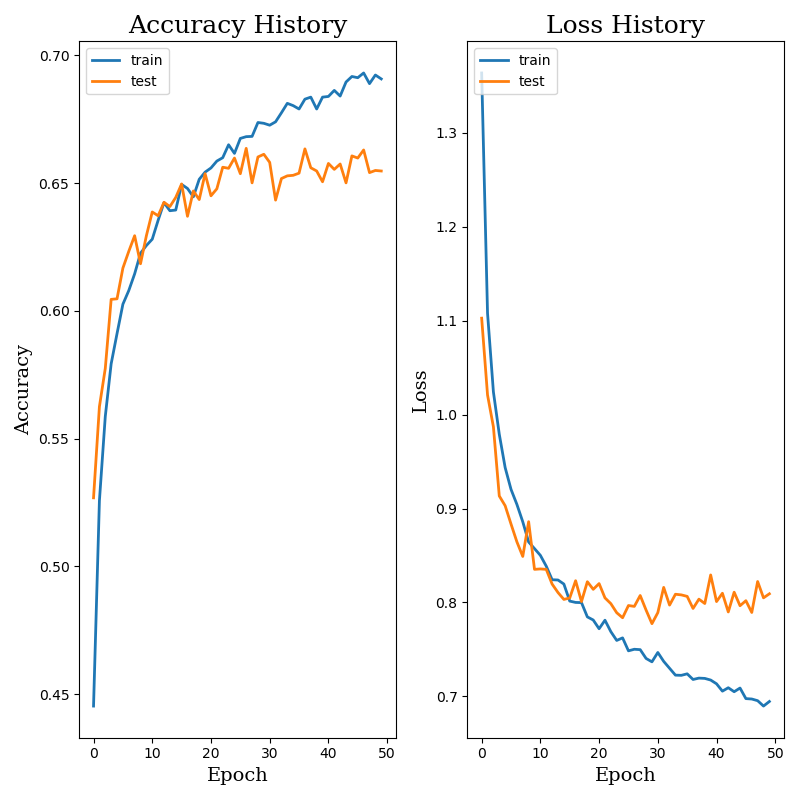
Trainable params: 332,326

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

None

Process finished with exit code 0



## Conclusiones y Resultado vs. Version Anterior:

Acc. Test converge a partir de la epoch 25 aproximadamente. Se queda en un valor aproximado del 65-66% (65.48% en la última epoch). Con respecto al Acc. Train, continúa mejorando en cada epoch, llegando hasta el 69.08% de precisión.

Overfitting.

El modelo no mejora con esta estructura.

Valorar meter más capas Dense y/o Conv2D.

Tamaño imágenes a 128x128?