## Modelo con Auto Tunner Keras

!pip install -q -U keras-tuner

import kerastuner as kt

from tensorflow import keras

def constructor\_modelos(hp):

  model = tf.keras.models.Sequential()

  model.add(tf.keras.layers.Conv2D(75, (3,3),

activation= "relu", input\_shape = (28, 28, 1)))

  model.add(tf.keras.layers.MaxPool2D((2,2)))

  model.add(tf.keras.layers.Flatten())

  hp\_units = hp.Int("units", min\_value = 32, max\_value = 512, step = 32)

  model.add(tf.keras.layers.Dense(units=hp\_units,

activation = "relu", kernel\_regularizer= regularizers.l2(1e-5)))

  model.add(tf.keras.layers.Dropout(0.2))

  model.add(tf.keras.layers.Dense(128,

activation = "relu", kernel\_regularizer= regularizers.l2(1e-5)))

  model.add(tf.keras.layers.Dropout(0.2))

  model.add(tf.keras.layers.Dense(len(classes), activation = "softmax"))

**hp\_learning\_rate** = hp.Choice('learning\_rate', values = [1e-2, 1e-3, 1e-4])

  model.compile(optimizer = keras.optimizers.Adam(learning\_rate=hp\_learning\_rate),

loss = "categorical\_crossentropy", metrics = ["accuracy"])

  return model

tuner = kt.Hyperband(

    constructor\_modelos,

    objective = "val\_accuracy",

    max\_epochs = 20,

    factor = 3,

    directory = "models/",

    project\_name = "platzi-tunner"

)

tuner.search(train\_generator,

 epochs =20,

 validation\_data = validation\_generator

)

best\_hps = tuner.get\_best\_hyperparameters(num\_trials =1)[0]

print(best\_hps.get("units"))

print(best\_hps.get("learning\_rate"))

hypermodel = tuner.hypermodel.build(best\_hps)

history\_hypermodel = hypermodel.fit(

    train\_generator,

    epochs = 20,

    callbacks = [callback\_early],

    validation\_data = validation\_generator

config\_dict = hypermodel.get\_config()

print(config\_dict)

model\_same\_config = tf.keras.Sequential.from\_config(config\_dict)

model\_same\_config.summary()

Model: "sequential"

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

Layer (type) Output Shape Param #

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

conv2d (Conv2D) (None, 26, 26, 75) 750

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

max\_pooling2d (MaxPooling2D) (None, 13, 13, 75) 0

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

flatten (Flatten) (None, 12675) 0

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

dense (Dense) (None, 416) 5273216

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

dropout (Dropout) (None, 416) 0

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

dense\_1 (Dense) (None, 128) 53376

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

dropout\_1 (Dropout) (None, 128) 0

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

dense\_2 (Dense) (None, 24) 3096

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

Total params: 5,330,438

Trainable params: 5,330,438

Non-trainable params: 0