from matplotlib import pyplot

from keras.datasets import cifar10

from keras.utils import to\_categorical

from keras.models import Sequential

from keras.layers import Conv2D

from keras.layers import MaxPooling2D

from keras.layers import Dense

from keras.layers import Flatten

from keras.optimizers import SGD

(trainX, trainY), (testX, testY) = cifar10.load\_data()

trainY = to\_categorical(trainY)

testY = to\_categorical(testY)

trainX = trainX.astype('float32')

testX = testX.astype('float32')

# normalize to range 0-1

trainX = trainX / 255.0

testX = testX / 255.0

model = Sequential()

model.add(Conv2D(32, (3, 3), activation='relu', kernel\_initializer='he\_uniform', padding='same', input\_shape=(32, 32, 3)))

model.add(Conv2D(32, (3, 3), activation='relu', kernel\_initializer='he\_uniform', padding='same'))

model.add(MaxPooling2D((2, 2)))

model.add(Flatten())

model.add(Dense(128, activation='relu', kernel\_initializer='he\_uniform'))

model.add(Dense(10, activation='softmax'))

# compile model

opt = SGD(lr=0.001, momentum=0.9)

model.compile(optimizer=opt, loss='categorical\_crossentropy', metrics=['accuracy'])

history = model.fit(trainX, trainY, epochs=100, batch\_size=64, validation\_data=(testX, testY), verbose=0)

\_, acc = model.evaluate(testX, testY, verbose=0)

print('> %.3f' % (acc \* 100.0))

pyplot.title('Cross Entropy Loss')

pyplot.plot(history.history['loss'], color='blue', label='train')

pyplot.plot(history.history['val\_loss'], color='orange', label='test')

# plot accuracy

pyplot.title('Classification Accuracy')

pyplot.plot(history.history['accuracy'], color='blue', label='train')

pyplot.plot(history.history['val\_accuracy'], color='orange', label='test')