

Práctica 03 – Python Expert

- 1) Documente y explique la utilidad cada línea de código del proceso de creación de una red neuronal simple en Keras

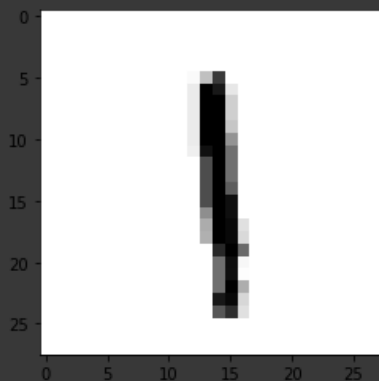
```
import keras # importar la libreria Keras para Deep Learning AI
from keras.datasets import mnist
(x_train, y_train), (x_test, y_test) = mnist.load_data()
# importamos el dataset MNIST compuesto de 70,000 imagenes de digitos escritos a mano
# en la forma de 4 arrays

import matplotlib.pyplot as plt # importar matplotlib.pyplot para graficado
plt.imshow(x_train[8], cmap=plt.cm.binary) # mostrar la imagen 8, en formato de mapa de dos colores (escala de grises)
print(y_train[8]) # mostrar la etiqueta para la imagen 8
print(x_train.ndim) # dimensiones del array x
print(x_train.shape) # ejes del array x
print(x_train.dtype) # tipo de datos del array
x_train = x_train.astype('float32') # crear copia del array, con casting del tipo float32
x_test = x_test.astype('float32')
x_train = x_train.reshape(60000, 784) # cambiar forma del array en (x,y)
x_test = x_test.reshape(10000, 784)

print('x_train:', x_train)
print('x_test:', x_test)
```

Salidas: (No en orden necesariamente)

```
1
3
(60000, 28, 28)
uint8
x_train: [[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
x_test : [[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
```



- 2) Según lo trabajado en clase y apoyado con las instrucciones anteriores explique el resultado al ejecutar "model.summary()":

Summary: Visualización de la arquitectura de un modelo secuencial (RNC)

El método summary es un resumen del modelo "sequential_1" expuesto. Permite ver información sobre los parámetros y formato de los tensores de cada capa.

La primera capa (dense_1) posee arrays de entrada con 10 elementos y 784 nodos. Se aplica la función de activación **sigmoide**.

Después se adiciona una capa densamente conectada (dense_2) con la función de activación **softmax**.

La forma de la tupla de salida de cada capa es de 10 elementos.

Se obtiene un total de 7960 parámetros entrenables.