Explorando MNIST y redes neuronales profundas

El propósito de este documento es ayudarlo a aprender sobre redes neuronales profundas y explorar cómo el cambio de la arquitectura de una red neuronal afecta el rendimiento de la red.

Antes de poder construir redes neuronales, necesitamos importar algunas cosas de Keras y preparar nuestros datos. El siguiente código extrae el conjunto de datos MNIST, proporcionado por Keras, y corvierte las imágenes de 28x28 píxeles en un vector con una longitud 784. Además, modifica las etiquetas de un valor numérico 0-9 a un vector *one-hot encoded*.

One-hot encoding hace referencia a un grupo de bits entre los cuales las combinaciones validas de valores son solo aquellas con un solo bit alto (1) y todas las demás bajas (0).

```
import tensorflow as tf
import keras
from keras.datasets import mnist
#from tensorflow.keras.utils import plot_model
from keras.utils.vis_utils import plot_model
from keras.layers import Dense #capas fully connected
from keras.models import Sequential
from matplotlib import pyplot as plt
from random import randint
import numpy as np

from keras.utils import to_categorical ##############################

print(tf. __version__)
# Preparar el conjunto de datos
# Configurar la división del entrenamiento y prueba
(x_train, y_train), (x_test, y_test) = mnist.load_data()

# Hacer una copia antes de convertir a 1D
# esta copia se usa para mostrar las imágenes
x_train_drawing = x_train

image_size = 784 # 28 x 28
x_train = x_train.reshape(x_train.shape[0], image_size)  #vector de
784, flattening
x_test = x_test.reshape(x_test.shape[0], image_size)

# Convierte vectores de clase en matrices de clases binarias (one-hot
encoding)
num_classes = 10
y_train = to_categorical(y_train, num_classes)
```

```
y_test = to_categorical(y_test, num_classes)

print(y_test)
print(x_train.shape) #m=num de ejemplos = 60 000, n=784 (features)
print(x_test.shape)
```

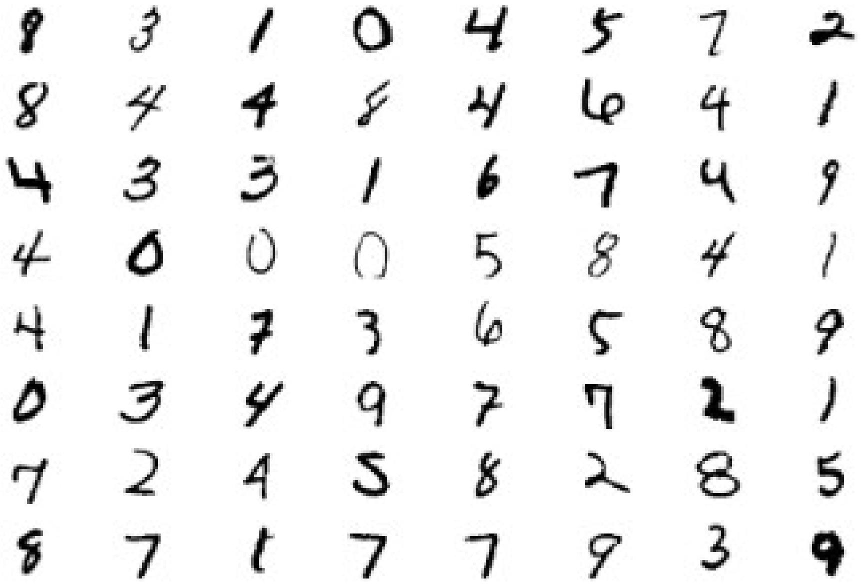
```
2.11.0
Downloading data from https://storage.googleapis.com/tensorflow/tf-
keras-datasets/mnist.npz
11490434/11490434 [==============================] - 0s 0us/step
[[0. 0. 0. ... 1. 0. 0.]
 [0. 0. 1. ... 0. 0. 0.]
 [0. 1. 0. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
(60000, 784)
(10000, 784)
```

Un vistazo a algunos dígitos al azar

Es bueno tener una idea del conjunto de datos con el que estamos trabajando. Ejecute este código varias veces para ver los nuevos dígitos seleccionados al azar del conjunto de entrenamiento.

```
#iteramos desde i=0 hasta 63. De manera general range(start, stop,
step)
print(list(range(5,1,-1)))
for i in range(64):
    ax = plt.subplot(8, 8, i+1)
    ax.axis('off')
    plt.imshow(x_train_drawing[randint(0, x_train.shape[0])],
cmap='Greys')

#verificado: por cada ejecucion, se obtiene una imagen, un conjunto de
datos, distinto
#pero con las mismas caracteristicas
```

```
[5, 4, 3, 2]
```



0.1 Primera red (3 puntos)

Aquí hay una primera red simple para resolver MNIST. Tiene una sola capa oculta con 32 nodos.

1. *La red tiene un total de 25450 parámetros entrenables. Demuestre cómo Keras calcula el número de parámetros entrenables en esta arquitectura*

La red neuronal cuenta con dos capas. Una capa oculta y otra capa de salida. Es por ello qué, el número de parámetros entrenables depende del número de unidades en la capa actual y en la capa anterior. Keras se encarga de calcular el número de parámetros del siguiente modo: La capa de entrada cuenta con 784 nodos, puesto qué es el resultado de la multiplicación de 28 x 28. Por otro lado, la capa oculta tiene 32 nodos y la capa de salida tiene 10 nodos. Ahora, es imprescindible tomar en cuenta qué, al ser una red neural densa, existe bias en cada capa oculta. Es por ello que hay 784 parámetros entrenables. Por tal motivo la red tiene un total de 25450 parámetros entrenables.

El cálculo vendría dado de la siguiente manera (Los términos de suma son debidos al elemento de bias que se debe agregar en la capa de input y en la capa oculta):

- De la capa de input a la capa oculta: 784 x 32 + 32 = 25120
- De la capa oculta a la capa de salida: 32 x 10 + 10 = 330
- El total de parámetros entrenables es: 25450.

1. *Qué significa `None` en la forma de salida (output shape) que se muestra como `(None,32)`?*

El número 32 qué se encuentra en la forma de salida indica qué la capa oculta tiene 32 neuronas, del mismo modo, indica qué la salida de la capa oculta es un vector de 32 elementos. Por otro lado, la palabra "None" indica qué la cantidad de muestras puede ser variable y no está definida.

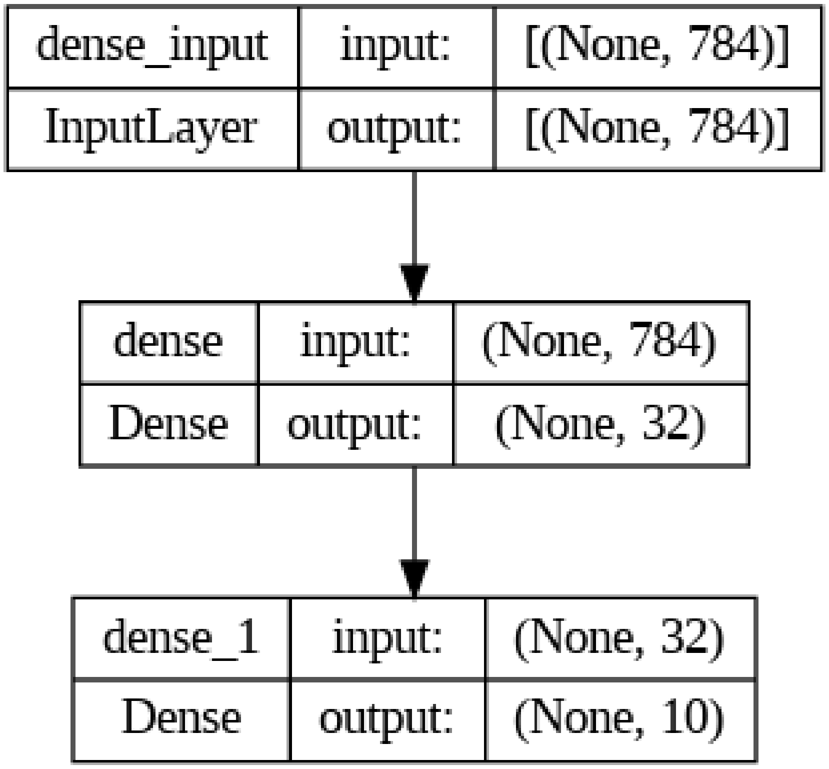
None implica que el input de la red neuronal puede ser más de una muestra a la vez. Es decir, implica que esta dimensión es variable (en algunos ejemplos encontrados en internet se especifica que este campo corresponde al batch size del modelo). Se recuerda que el batch size implica que el modelo hará uso de varias imágenes de 28x28 al mismo tiempo.

1. *cuál es la función de activación softmax y su relación con la sigmoide?*

La función de activación softmax es una función qué se emplea en las redes neuronales para transformar un conjunto de datos en una distribución de probabilidad. La relación qué existe entre la función de activación softmax y la sigmoide es qué la función softmax es una generalización de la función Sigmoide.

```
model = Sequential()

# La capa de entrada requiere el parámetro especial input_shape que
debe
# coincidir con la forma de nuestros datos de entrenamiento.
model.add(Dense(units=32, activation='sigmoid',
input_shape=(image_size,))) #Input and Hidden Layer
model.add(Dense(units=num_classes, activation='softmax')) #output
Layer. Si num_classes = 2 (pos y neg): model.add(Dense(units=1,
activation='sigmoid')) equivalente model.add(Dense(units=2,
activation='softmax'))
model.summary() # REPRESENTAR
plot_model(model, to_file='model_plot.png', show_shapes=True,
show_layer_names=True)
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 32)                25120

 dense_1 (Dense)             (None, 10)                330

=================================================================
Total params: 25,450
Trainable params: 25,450
Non-trainable params: 0
_________________________________________________________________
```



0.2 Entrenar y evaluar la red (7 puntos)

Este código entrena y evalúa el modelo que definimos anteriormente. También usa `matplotlib` y el objeto `history` proporcionado por Keras, que rastrea cómo se comporta el modelo a través de su entrenamiento. Observe que usamos el objeto `history` para trazar la precisión del entrenamiento y la precisión de la validación a lo largo del tiempo (*epochs*).

1. *En el siguiente código, la función de costo o pérdida (loss function) está definida como* `'categorical_crossentropy'`*. Consulte cuál es la forma matemática de esta función y explique sus parámetros.*

La función de costo está definida como:

$$L(y,\hat{y}) = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

Es una función empleada comúnmente en problemas de clasificación en los que las etiquetas de los datos están en formato one-hot encoding. Mide la discrepancia entre la distribución de probabilidad de las predicciones de la red neuronal y la distribución de probabilidad real de las etiquetas.

1. *Cuál es la diferencia entra la función de costo de entropía cruzada (Cross-Entropy loss) con respecto a la función de costo definida para la regresión logística?*

La diferencia qué existe entre la función de costo de entropía cruzada (Cross-Entropy loss) con respecto a la función de costo definida para la regresión logística, es qué la función de costo se emplea para clasificar un conjunto de datos en una clasificación de N clases. Por otro lado, la función de costo definida para la regresión logística se usa para clasificar un conjunto de datos en una clasificación de 2 clases.

1. *Consulte las diferencias entre minibatch gradient descent, batch gradient descent y Stochastic gradient descent.*

Minibatch gradient descent, batch gradient descent y Stochastic gradient descent, son 3 algoritmos de optimización que se emplean en el aprendizaje automático. Sus principales diferencias son:

- Batch gradient descent: Actualiza parámetros del modelo, una vez calculada la suma de las contribuciones de la función de pérdida para todos los ejemplos de entrenamiento. Puede llegar a ser preciso, pero del mismo modo, puede ser lento.

- Stochastic gradient descent : Calcula el gradiente de la función de pérdida para un solo ejemplo de entrenamiento y se utiliza para actualizar los parámetros del modelo antes de pasar al siguiente ejemplo. Es más rápido qué el anterior, pero puede ser menos preciso.

- Minibatch gradient descent: Emplea un subconjunto de tamaño fijo de los datos de entrenamiento. Es una combinación de Batch gradient descent y Stochastic gradient descent.

1. *Según lo consultado en el punto anterior, qué tipo de gradient descent (batch, minibatch o stochastic) describe la función de costo para la red neuronal estudiada en clase? Justifique su respuesta.*

Según lo consultado, la función de costo para gradient descent es Minibatch gradient descent, porque mediante un subconjunto, calcula el gradiente de la función con respecto a los parámetros de la red neuronal.

1. *Explique qué significa una época en una red neuronal (epoch)*

Una época en una red neuronal es una iteración completa a través de todo el conjunto de datos de entrenamiento durante el proceso de entrenamiento del modelo. Para entrenar una red neuronal, usualmente se usan varias épocas.

1. *Explique qué es el tamaño del batch de una red neuronal.*

El tamaño del batch en una red neuronal se refiere a la cantidad de datos de entrenamiento o cantidad de ejemplos de entrenamiento que se emplean para obtener el gradiente de la función de costo.

1. *Cuál es la diferencia entre tamaño del batch y época?*

La principal diferencia entre tamaño del batch y época es qué el tamaño del batch es el número de datos de entrenamiento que se emplean para obtener el gradiente. Por otro lado, una época es cuando se ha recorrido todos los datos de entrenamiento una vez.

```
model.compile(optimizer="sgd", loss='categorical_crossentropy',
metrics=['accuracy']) #LOSS = COST function. EVALUAR.
history = model.fit(x_train, y_train, batch_size=128, epochs=200,
verbose=True, validation_split=.1) #fit = OPTIMIZAR.
loss, accuracy  = model.evaluate(x_test, y_test, verbose=False)
#evalua en la última época, no necesariamente la mejor. Ver
checkpoints y monitors.
mypredictions  = model.predict(x_test)

plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['training', 'validation'], loc='best')
plt.show()

print(f'Test loss: {loss:.3}')
print(f'Test accuracy: {accuracy:.3}')
print(f'Shape of my predictions (test set): {mypredictions.shape}')
yidx = np.argmax(y_test, axis=1) #from one hot encoding to integers
mypidx = mypredictions[1,:]
np.set_printoptions(precision=3, suppress=True)
print(f'First prediction for number {yidx[100]}, probabilities:
{mypidx}')
```

```
Epoch 1/200
422/422 [==============================] - 2s 4ms/step - loss: 1.3516
- accuracy: 0.6539 - val_loss: 0.9080 - val_accuracy: 0.8372
Epoch 2/200
422/422 [==============================] - 2s 4ms/step - loss: 0.8246
- accuracy: 0.8339 - val_loss: 0.6664 - val_accuracy: 0.8723
Epoch 3/200
422/422 [==============================] - 2s 4ms/step - loss: 0.6584
- accuracy: 0.8630 - val_loss: 0.5429 - val_accuracy: 0.8940
Epoch 4/200
422/422 [==============================] - 2s 6ms/step - loss: 0.5669
- accuracy: 0.8774 - val_loss: 0.4794 - val_accuracy: 0.9025
Epoch 5/200
422/422 [==============================] - 2s 5ms/step - loss: 0.5106
- accuracy: 0.8857 - val_loss: 0.4388 - val_accuracy: 0.9110
Epoch 6/200
422/422 [==============================] - 2s 4ms/step - loss: 0.4685
- accuracy: 0.8926 - val_loss: 0.4004 - val_accuracy: 0.9077
Epoch 7/200
422/422 [==============================] - 2s 4ms/step - loss: 0.4370
```

```
- accuracy: 0.8977 - val_loss: 0.3846 - val_accuracy: 0.9157
Epoch 8/200
422/422 [==============================] - 2s 4ms/step - loss: 0.4136
- accuracy: 0.9012 - val_loss: 0.3553 - val_accuracy: 0.9135
Epoch 9/200
422/422 [==============================] - 1s 3ms/step - loss: 0.3952
- accuracy: 0.9026 - val_loss: 0.3402 - val_accuracy: 0.9167
Epoch 10/200
422/422 [==============================] - 2s 4ms/step - loss: 0.3833
- accuracy: 0.9037 - val_loss: 0.3238 - val_accuracy: 0.9220
Epoch 11/200
422/422 [==============================] - 2s 4ms/step - loss: 0.3641
- accuracy: 0.9081 - val_loss: 0.3126 - val_accuracy: 0.9233
Epoch 12/200
422/422 [==============================] - 2s 5ms/step - loss: 0.3565
- accuracy: 0.9098 - val_loss: 0.2992 - val_accuracy: 0.9252
Epoch 13/200
422/422 [==============================] - 1s 3ms/step - loss: 0.3447
- accuracy: 0.9108 - val_loss: 0.2941 - val_accuracy: 0.9243
Epoch 14/200
422/422 [==============================] - 1s 3ms/step - loss: 0.3332
- accuracy: 0.9118 - val_loss: 0.2852 - val_accuracy: 0.9273
Epoch 15/200
422/422 [==============================] - 2s 4ms/step - loss: 0.3301
- accuracy: 0.9144 - val_loss: 0.2882 - val_accuracy: 0.9225
Epoch 16/200
422/422 [==============================] - 2s 4ms/step - loss: 0.3212
- accuracy: 0.9153 - val_loss: 0.2768 - val_accuracy: 0.9283
Epoch 17/200
422/422 [==============================] - 1s 3ms/step - loss: 0.3148
- accuracy: 0.9175 - val_loss: 0.2726 - val_accuracy: 0.9295
Epoch 18/200
422/422 [==============================] - 1s 3ms/step - loss: 0.3079
- accuracy: 0.9182 - val_loss: 0.2615 - val_accuracy: 0.9280
Epoch 19/200
422/422 [==============================] - 2s 5ms/step - loss: 0.3034
- accuracy: 0.9198 - val_loss: 0.2568 - val_accuracy: 0.9325
Epoch 20/200
422/422 [==============================] - 2s 5ms/step - loss: 0.3000
- accuracy: 0.9192 - val_loss: 0.2586 - val_accuracy: 0.9315
Epoch 21/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2962
- accuracy: 0.9204 - val_loss: 0.2573 - val_accuracy: 0.9300
Epoch 22/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2886
- accuracy: 0.9230 - val_loss: 0.2508 - val_accuracy: 0.9350
Epoch 23/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2892
- accuracy: 0.9213 - val_loss: 0.2447 - val_accuracy: 0.9348
Epoch 24/200
```

```
422/422 [==============================] - 1s 3ms/step - loss: 0.2833
- accuracy: 0.9236 - val_loss: 0.2445 - val_accuracy: 0.9337
Epoch 25/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2804
- accuracy: 0.9234 - val_loss: 0.2376 - val_accuracy: 0.9337
Epoch 26/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2762
- accuracy: 0.9256 - val_loss: 0.2356 - val_accuracy: 0.9353
Epoch 27/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2773
- accuracy: 0.9233 - val_loss: 0.2391 - val_accuracy: 0.9358
Epoch 28/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2678
- accuracy: 0.9268 - val_loss: 0.2301 - val_accuracy: 0.9378
Epoch 29/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2619
- accuracy: 0.9280 - val_loss: 0.2297 - val_accuracy: 0.9372
Epoch 30/200
422/422 [==============================] - 3s 6ms/step - loss: 0.2631
- accuracy: 0.9287 - val_loss: 0.2353 - val_accuracy: 0.9347
Epoch 31/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2587
- accuracy: 0.9281 - val_loss: 0.2244 - val_accuracy: 0.9395
Epoch 32/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2538
- accuracy: 0.9303 - val_loss: 0.2246 - val_accuracy: 0.9370
Epoch 33/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2509
- accuracy: 0.9311 - val_loss: 0.2316 - val_accuracy: 0.9362
Epoch 34/200
422/422 [==============================] - 2s 6ms/step - loss: 0.2530
- accuracy: 0.9294 - val_loss: 0.2177 - val_accuracy: 0.9413
Epoch 35/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2523
- accuracy: 0.9311 - val_loss: 0.2236 - val_accuracy: 0.9375
Epoch 36/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2466
- accuracy: 0.9315 - val_loss: 0.2258 - val_accuracy: 0.9343
Epoch 37/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2489
- accuracy: 0.9314 - val_loss: 0.2157 - val_accuracy: 0.9407
Epoch 38/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2455
- accuracy: 0.9319 - val_loss: 0.2186 - val_accuracy: 0.9398
Epoch 39/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2481
- accuracy: 0.9303 - val_loss: 0.2179 - val_accuracy: 0.9392
Epoch 40/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2429
- accuracy: 0.9321 - val_loss: 0.2173 - val_accuracy: 0.9380
```

```
Epoch 41/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2390
- accuracy: 0.9330 - val_loss: 0.2099 - val_accuracy: 0.9428
Epoch 42/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2383
- accuracy: 0.9338 - val_loss: 0.2065 - val_accuracy: 0.9468
Epoch 43/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2368
- accuracy: 0.9335 - val_loss: 0.2116 - val_accuracy: 0.9415
Epoch 44/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2329
- accuracy: 0.9346 - val_loss: 0.2119 - val_accuracy: 0.9423
Epoch 45/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2384
- accuracy: 0.9338 - val_loss: 0.2133 - val_accuracy: 0.9432
Epoch 46/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2318
- accuracy: 0.9354 - val_loss: 0.2070 - val_accuracy: 0.9422
Epoch 47/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2296
- accuracy: 0.9354 - val_loss: 0.2085 - val_accuracy: 0.9402
Epoch 48/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2267
- accuracy: 0.9360 - val_loss: 0.1971 - val_accuracy: 0.9440
Epoch 49/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2241
- accuracy: 0.9369 - val_loss: 0.2006 - val_accuracy: 0.9455
Epoch 50/200
422/422 [==============================] - 2s 5ms/step - loss: 0.2200
- accuracy: 0.9380 - val_loss: 0.2016 - val_accuracy: 0.9443
Epoch 51/200
422/422 [==============================] - 1s 4ms/step - loss: 0.2262
- accuracy: 0.9351 - val_loss: 0.2003 - val_accuracy: 0.9460
Epoch 52/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2231
- accuracy: 0.9374 - val_loss: 0.2021 - val_accuracy: 0.9447
Epoch 53/200
422/422 [==============================] - 1s 4ms/step - loss: 0.2279
- accuracy: 0.9345 - val_loss: 0.1999 - val_accuracy: 0.9440
Epoch 54/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2203
- accuracy: 0.9379 - val_loss: 0.1946 - val_accuracy: 0.9448
Epoch 55/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2186
- accuracy: 0.9382 - val_loss: 0.1932 - val_accuracy: 0.9462
Epoch 56/200
422/422 [==============================] - 1s 4ms/step - loss: 0.2194
- accuracy: 0.9363 - val_loss: 0.1967 - val_accuracy: 0.9463
Epoch 57/200
422/422 [==============================] - 3s 7ms/step - loss: 0.2187
```

```
- accuracy: 0.9380 - val_loss: 0.1935 - val_accuracy: 0.9445
Epoch 58/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2157
- accuracy: 0.9385 - val_loss: 0.2004 - val_accuracy: 0.9455
Epoch 59/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2146
- accuracy: 0.9386 - val_loss: 0.1925 - val_accuracy: 0.9485
Epoch 60/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2142
- accuracy: 0.9389 - val_loss: 0.1980 - val_accuracy: 0.9442
Epoch 61/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2144
- accuracy: 0.9394 - val_loss: 0.1921 - val_accuracy: 0.9455
Epoch 62/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2094
- accuracy: 0.9403 - val_loss: 0.1933 - val_accuracy: 0.9477
Epoch 63/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2107
- accuracy: 0.9401 - val_loss: 0.1946 - val_accuracy: 0.9475
Epoch 64/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2088
- accuracy: 0.9408 - val_loss: 0.1858 - val_accuracy: 0.9483
Epoch 65/200
422/422 [==============================] - 2s 6ms/step - loss: 0.2093
- accuracy: 0.9405 - val_loss: 0.1905 - val_accuracy: 0.9490
Epoch 66/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2065
- accuracy: 0.9411 - val_loss: 0.1923 - val_accuracy: 0.9460
Epoch 67/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2064
- accuracy: 0.9409 - val_loss: 0.1954 - val_accuracy: 0.9453
Epoch 68/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2050
- accuracy: 0.9396 - val_loss: 0.1844 - val_accuracy: 0.9482
Epoch 69/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2043
- accuracy: 0.9400 - val_loss: 0.1863 - val_accuracy: 0.9493
Epoch 70/200
422/422 [==============================] - 1s 3ms/step - loss: 0.2020
- accuracy: 0.9419 - val_loss: 0.1908 - val_accuracy: 0.9475
Epoch 71/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2004
- accuracy: 0.9421 - val_loss: 0.1843 - val_accuracy: 0.9487
Epoch 72/200
422/422 [==============================] - 2s 4ms/step - loss: 0.2013
- accuracy: 0.9422 - val_loss: 0.1811 - val_accuracy: 0.9482
Epoch 73/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1989
- accuracy: 0.9426 - val_loss: 0.1877 - val_accuracy: 0.9475
Epoch 74/200
```

```
422/422 [==============================] - 2s 4ms/step - loss: 0.2022
- accuracy: 0.9411 - val_loss: 0.1861 - val_accuracy: 0.9472
Epoch 75/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1975
- accuracy: 0.9433 - val_loss: 0.1837 - val_accuracy: 0.9487
Epoch 76/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1950
- accuracy: 0.9434 - val_loss: 0.1812 - val_accuracy: 0.9492
Epoch 77/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1977
- accuracy: 0.9434 - val_loss: 0.1872 - val_accuracy: 0.9480
Epoch 78/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1939
- accuracy: 0.9427 - val_loss: 0.1803 - val_accuracy: 0.9478
Epoch 79/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1933
- accuracy: 0.9437 - val_loss: 0.1802 - val_accuracy: 0.9490
Epoch 80/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1956
- accuracy: 0.9426 - val_loss: 0.1823 - val_accuracy: 0.9488
Epoch 81/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1903
- accuracy: 0.9461 - val_loss: 0.1808 - val_accuracy: 0.9497
Epoch 82/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1975
- accuracy: 0.9434 - val_loss: 0.1819 - val_accuracy: 0.9467
Epoch 83/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1954
- accuracy: 0.9434 - val_loss: 0.1841 - val_accuracy: 0.9495
Epoch 84/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1909
- accuracy: 0.9440 - val_loss: 0.1769 - val_accuracy: 0.9498
Epoch 85/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1893
- accuracy: 0.9445 - val_loss: 0.1763 - val_accuracy: 0.9497
Epoch 86/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1863
- accuracy: 0.9457 - val_loss: 0.1719 - val_accuracy: 0.9517
Epoch 87/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1887
- accuracy: 0.9453 - val_loss: 0.1727 - val_accuracy: 0.9505
Epoch 88/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1874
- accuracy: 0.9457 - val_loss: 0.1808 - val_accuracy: 0.9462
Epoch 89/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1852
- accuracy: 0.9459 - val_loss: 0.1820 - val_accuracy: 0.9458
Epoch 90/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1883
- accuracy: 0.9456 - val_loss: 0.1727 - val_accuracy: 0.9513
```

```
Epoch 91/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1850
- accuracy: 0.9476 - val_loss: 0.1789 - val_accuracy: 0.9477
Epoch 92/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1795
- accuracy: 0.9480 - val_loss: 0.1754 - val_accuracy: 0.9505
Epoch 93/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1832
- accuracy: 0.9465 - val_loss: 0.1766 - val_accuracy: 0.9497
Epoch 94/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1864
- accuracy: 0.9460 - val_loss: 0.1783 - val_accuracy: 0.9472
Epoch 95/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1838
- accuracy: 0.9458 - val_loss: 0.1738 - val_accuracy: 0.9482
Epoch 96/200
422/422 [==============================] - 3s 6ms/step - loss: 0.1863
- accuracy: 0.9465 - val_loss: 0.1722 - val_accuracy: 0.9487
Epoch 97/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1824
- accuracy: 0.9468 - val_loss: 0.1784 - val_accuracy: 0.9467
Epoch 98/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1802
- accuracy: 0.9472 - val_loss: 0.1807 - val_accuracy: 0.9440
Epoch 99/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1774
- accuracy: 0.9487 - val_loss: 0.1735 - val_accuracy: 0.9483
Epoch 100/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1781
- accuracy: 0.9479 - val_loss: 0.1879 - val_accuracy: 0.9435
Epoch 101/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1799
- accuracy: 0.9482 - val_loss: 0.1770 - val_accuracy: 0.9507
Epoch 102/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1790
- accuracy: 0.9476 - val_loss: 0.1712 - val_accuracy: 0.9510
Epoch 103/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1784
- accuracy: 0.9471 - val_loss: 0.1783 - val_accuracy: 0.9473
Epoch 104/200
422/422 [==============================] - 2s 6ms/step - loss: 0.1787
- accuracy: 0.9479 - val_loss: 0.1857 - val_accuracy: 0.9445
Epoch 105/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1783
- accuracy: 0.9486 - val_loss: 0.1758 - val_accuracy: 0.9463
Epoch 106/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1783
- accuracy: 0.9484 - val_loss: 0.1741 - val_accuracy: 0.9473
Epoch 107/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1747
```

```
- accuracy: 0.9484 - val_loss: 0.1738 - val_accuracy: 0.9472
Epoch 108/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1754
- accuracy: 0.9491 - val_loss: 0.1723 - val_accuracy: 0.9487
Epoch 109/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1789
- accuracy: 0.9484 - val_loss: 0.1722 - val_accuracy: 0.9490
Epoch 110/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1758
- accuracy: 0.9490 - val_loss: 0.1717 - val_accuracy: 0.9487
Epoch 111/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1781
- accuracy: 0.9486 - val_loss: 0.1703 - val_accuracy: 0.9488
Epoch 112/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1757
- accuracy: 0.9479 - val_loss: 0.1768 - val_accuracy: 0.9462
Epoch 113/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1739
- accuracy: 0.9488 - val_loss: 0.1679 - val_accuracy: 0.9482
Epoch 114/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1685
- accuracy: 0.9505 - val_loss: 0.1687 - val_accuracy: 0.9510
Epoch 115/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1690
- accuracy: 0.9506 - val_loss: 0.1710 - val_accuracy: 0.9470
Epoch 116/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1721
- accuracy: 0.9491 - val_loss: 0.1723 - val_accuracy: 0.9505
Epoch 117/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1703
- accuracy: 0.9497 - val_loss: 0.1707 - val_accuracy: 0.9520
Epoch 118/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1659
- accuracy: 0.9503 - val_loss: 0.1744 - val_accuracy: 0.9515
Epoch 119/200
422/422 [==============================] - 3s 6ms/step - loss: 0.1703
- accuracy: 0.9493 - val_loss: 0.1683 - val_accuracy: 0.9503
Epoch 120/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1674
- accuracy: 0.9507 - val_loss: 0.1654 - val_accuracy: 0.9502
Epoch 121/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1672
- accuracy: 0.9503 - val_loss: 0.1711 - val_accuracy: 0.9502
Epoch 122/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1648
- accuracy: 0.9511 - val_loss: 0.1709 - val_accuracy: 0.9482
Epoch 123/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1654
- accuracy: 0.9507 - val_loss: 0.1797 - val_accuracy: 0.9493
Epoch 124/200
```

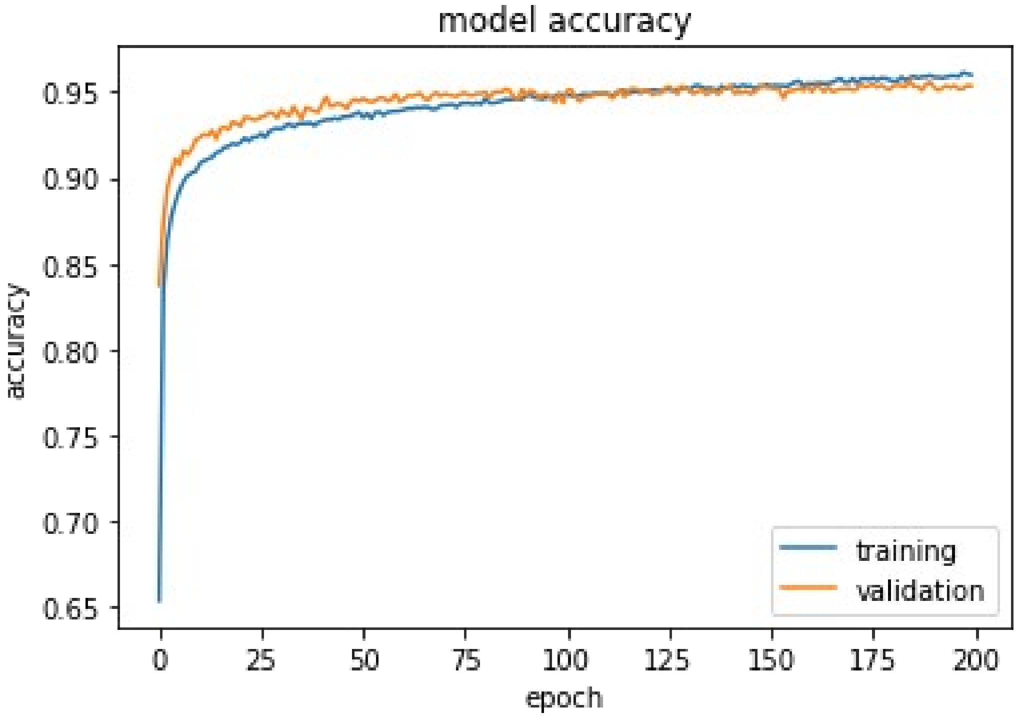
```
422/422 [==============================] - 2s 4ms/step - loss: 0.1683
- accuracy: 0.9503 - val_loss: 0.1752 - val_accuracy: 0.9490
Epoch 125/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1665
- accuracy: 0.9511 - val_loss: 0.1683 - val_accuracy: 0.9520
Epoch 126/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1650
- accuracy: 0.9513 - val_loss: 0.1679 - val_accuracy: 0.9530
Epoch 127/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1668
- accuracy: 0.9506 - val_loss: 0.1699 - val_accuracy: 0.9513
Epoch 128/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1635
- accuracy: 0.9515 - val_loss: 0.1741 - val_accuracy: 0.9485
Epoch 129/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1643
- accuracy: 0.9514 - val_loss: 0.1656 - val_accuracy: 0.9532
Epoch 130/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1604
- accuracy: 0.9526 - val_loss: 0.1678 - val_accuracy: 0.9525
Epoch 131/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1638
- accuracy: 0.9511 - val_loss: 0.1655 - val_accuracy: 0.9505
Epoch 132/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1589
- accuracy: 0.9531 - val_loss: 0.1708 - val_accuracy: 0.9488
Epoch 133/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1584
- accuracy: 0.9536 - val_loss: 0.1650 - val_accuracy: 0.9500
Epoch 134/200
422/422 [==============================] - 2s 6ms/step - loss: 0.1597
- accuracy: 0.9529 - val_loss: 0.1650 - val_accuracy: 0.9513
Epoch 135/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1652
- accuracy: 0.9518 - val_loss: 0.1656 - val_accuracy: 0.9485
Epoch 136/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1620
- accuracy: 0.9522 - val_loss: 0.1642 - val_accuracy: 0.9495
Epoch 137/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1624
- accuracy: 0.9515 - val_loss: 0.1622 - val_accuracy: 0.9508
Epoch 138/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1640
- accuracy: 0.9509 - val_loss: 0.1713 - val_accuracy: 0.9485
Epoch 139/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1579
- accuracy: 0.9533 - val_loss: 0.1670 - val_accuracy: 0.9510
Epoch 140/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1638
- accuracy: 0.9517 - val_loss: 0.1603 - val_accuracy: 0.9523
```

```
Epoch 141/200
422/422 [==============================] - 3s 7ms/step - loss: 0.1590
- accuracy: 0.9526 - val_loss: 0.1615 - val_accuracy: 0.9535
Epoch 142/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1549
- accuracy: 0.9538 - val_loss: 0.1709 - val_accuracy: 0.9490
Epoch 143/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1596
- accuracy: 0.9522 - val_loss: 0.1633 - val_accuracy: 0.9495
Epoch 144/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1552
- accuracy: 0.9539 - val_loss: 0.1678 - val_accuracy: 0.9493
Epoch 145/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1569
- accuracy: 0.9532 - val_loss: 0.1614 - val_accuracy: 0.9503
Epoch 146/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1560
- accuracy: 0.9533 - val_loss: 0.1720 - val_accuracy: 0.9487
Epoch 147/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1629
- accuracy: 0.9517 - val_loss: 0.1685 - val_accuracy: 0.9500
Epoch 148/200
422/422 [==============================] - 3s 6ms/step - loss: 0.1571
- accuracy: 0.9533 - val_loss: 0.1653 - val_accuracy: 0.9520
Epoch 149/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1589
- accuracy: 0.9531 - val_loss: 0.1639 - val_accuracy: 0.9528
Epoch 150/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1551
- accuracy: 0.9540 - val_loss: 0.1570 - val_accuracy: 0.9525
Epoch 151/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1570
- accuracy: 0.9532 - val_loss: 0.1630 - val_accuracy: 0.9522
Epoch 152/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1577
- accuracy: 0.9537 - val_loss: 0.1652 - val_accuracy: 0.9517
Epoch 153/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1542
- accuracy: 0.9533 - val_loss: 0.1623 - val_accuracy: 0.9512
Epoch 154/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1561
- accuracy: 0.9535 - val_loss: 0.1723 - val_accuracy: 0.9462
Epoch 155/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1564
- accuracy: 0.9531 - val_loss: 0.1715 - val_accuracy: 0.9505
Epoch 156/200
422/422 [==============================] - 3s 6ms/step - loss: 0.1535
- accuracy: 0.9544 - val_loss: 0.1654 - val_accuracy: 0.9503
Epoch 157/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1508
```

```
- accuracy: 0.9558 - val_loss: 0.1613 - val_accuracy: 0.9513
Epoch 158/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1497
- accuracy: 0.9560 - val_loss: 0.1656 - val_accuracy: 0.9518
Epoch 159/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1581
- accuracy: 0.9529 - val_loss: 0.1671 - val_accuracy: 0.9507
Epoch 160/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1529
- accuracy: 0.9547 - val_loss: 0.1596 - val_accuracy: 0.9533
Epoch 161/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1559
- accuracy: 0.9543 - val_loss: 0.1660 - val_accuracy: 0.9500
Epoch 162/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1520
- accuracy: 0.9548 - val_loss: 0.1676 - val_accuracy: 0.9532
Epoch 163/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1510
- accuracy: 0.9550 - val_loss: 0.1628 - val_accuracy: 0.9523
Epoch 164/200
422/422 [==============================] - 2s 6ms/step - loss: 0.1517
- accuracy: 0.9546 - val_loss: 0.1605 - val_accuracy: 0.9500
Epoch 165/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1512
- accuracy: 0.9549 - val_loss: 0.1592 - val_accuracy: 0.9530
Epoch 166/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1482
- accuracy: 0.9554 - val_loss: 0.1632 - val_accuracy: 0.9500
Epoch 167/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1464
- accuracy: 0.9566 - val_loss: 0.1588 - val_accuracy: 0.9507
Epoch 168/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1458
- accuracy: 0.9576 - val_loss: 0.1588 - val_accuracy: 0.9515
Epoch 169/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1484
- accuracy: 0.9554 - val_loss: 0.1659 - val_accuracy: 0.9515
Epoch 170/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1501
- accuracy: 0.9554 - val_loss: 0.1595 - val_accuracy: 0.9520
Epoch 171/200
422/422 [==============================] - 3s 6ms/step - loss: 0.1457
- accuracy: 0.9576 - val_loss: 0.1594 - val_accuracy: 0.9508
Epoch 172/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1501
- accuracy: 0.9550 - val_loss: 0.1595 - val_accuracy: 0.9513
Epoch 173/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1458
- accuracy: 0.9574 - val_loss: 0.1531 - val_accuracy: 0.9547
Epoch 174/200
```

```
422/422 [==============================] - 2s 4ms/step - loss: 0.1450
- accuracy: 0.9575 - val_loss: 0.1631 - val_accuracy: 0.9528
Epoch 175/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1474
- accuracy: 0.9569 - val_loss: 0.1621 - val_accuracy: 0.9533
Epoch 176/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1446
- accuracy: 0.9579 - val_loss: 0.1570 - val_accuracy: 0.9543
Epoch 177/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1455
- accuracy: 0.9565 - val_loss: 0.1621 - val_accuracy: 0.9525
Epoch 178/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1436
- accuracy: 0.9578 - val_loss: 0.1573 - val_accuracy: 0.9538
Epoch 179/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1444
- accuracy: 0.9577 - val_loss: 0.1621 - val_accuracy: 0.9523
Epoch 180/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1454
- accuracy: 0.9568 - val_loss: 0.1603 - val_accuracy: 0.9513
Epoch 181/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1466
- accuracy: 0.9559 - val_loss: 0.1530 - val_accuracy: 0.9533
Epoch 182/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1473
- accuracy: 0.9562 - val_loss: 0.1583 - val_accuracy: 0.9538
Epoch 183/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1409
- accuracy: 0.9584 - val_loss: 0.1570 - val_accuracy: 0.9535
Epoch 184/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1456
- accuracy: 0.9565 - val_loss: 0.1617 - val_accuracy: 0.9528
Epoch 185/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1420
- accuracy: 0.9580 - val_loss: 0.1526 - val_accuracy: 0.9565
Epoch 186/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1392
- accuracy: 0.9588 - val_loss: 0.1572 - val_accuracy: 0.9530
Epoch 187/200
422/422 [==============================] - 2s 5ms/step - loss: 0.1402
- accuracy: 0.9578 - val_loss: 0.1561 - val_accuracy: 0.9530
Epoch 188/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1395
- accuracy: 0.9585 - val_loss: 0.1567 - val_accuracy: 0.9508
Epoch 189/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1386
- accuracy: 0.9586 - val_loss: 0.1571 - val_accuracy: 0.9543
Epoch 190/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1369
- accuracy: 0.9593 - val_loss: 0.1570 - val_accuracy: 0.9555
```

```
Epoch 191/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1392
- accuracy: 0.9584 - val_loss: 0.1585 - val_accuracy: 0.9525
Epoch 192/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1409
- accuracy: 0.9582 - val_loss: 0.1586 - val_accuracy: 0.9510
Epoch 193/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1403
- accuracy: 0.9579 - val_loss: 0.1553 - val_accuracy: 0.9520
Epoch 194/200
422/422 [==============================] - 2s 6ms/step - loss: 0.1384
- accuracy: 0.9584 - val_loss: 0.1559 - val_accuracy: 0.9535
Epoch 195/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1391
- accuracy: 0.9590 - val_loss: 0.1563 - val_accuracy: 0.9523
Epoch 196/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1384
- accuracy: 0.9591 - val_loss: 0.1570 - val_accuracy: 0.9535
Epoch 197/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1385
- accuracy: 0.9586 - val_loss: 0.1587 - val_accuracy: 0.9517
Epoch 198/200
422/422 [==============================] - 2s 4ms/step - loss: 0.1338
- accuracy: 0.9610 - val_loss: 0.1589 - val_accuracy: 0.9517
Epoch 199/200
422/422 [==============================] - 1s 3ms/step - loss: 0.1355
- accuracy: 0.9604 - val_loss: 0.1541 - val_accuracy: 0.9535
Epoch 200/200
422/422 [==============================] - 1s 4ms/step - loss: 0.1376
- accuracy: 0.9596 - val_loss: 0.1541 - val_accuracy: 0.9533
313/313 [==============================] - 1s 2ms/step
```



```
Test loss: 0.187
Test accuracy: 0.945
Shape of my predictions (test set): (10000, 10)
First prediction for number 6, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Algunas Ayudas

Hay un par de cosas que haremos repetidamente en este notebook:

- Construir un modelo, y
- Evaluar ese modelo.

Estas funciones nos ayudarán a comparar "manzanas con manzanas", ya que podemos estar seguros de que cuando llamamos a `create_dense` y `evaluate` nuestros modelos y régimen de entrenamiento utilizarán los mismos **hiperparámetros**. Ambos usan algunas de las variables declaradas anteriormente y, por lo tanto, ambos están explícitamente destinados a trabajar con el conjunto de datos MNIST.

`create_dense` acepta una matriz del tamaños de la capa y devuelve un modelo Keras de una red neuronal completamente conectada con los tamaños de capa especificados. Por ejemplo, `create_dense ([32, 64, 128])` devolverá una red neuronal profundamente conectada con tres capas ocultas, la primera con 32 nodos, la segunda con 64 nodos y la tercera con 128 nodos.

`create_dense` usa la variable `image_size` declarada anteriormente, lo que significa que asume que los datos de entrada serán un vector con 784 unidades. Todas las capas ocultas usan la función de activación sigmoid, excepto la capa de salida, que usa softmax.

`evaluate` imprime un resumen del modelo, entrena el modelo y luego imprime la pérdida y la precisión. Esta función siempre ejecuta 5 épocas de entrenamiento y utiliza un *tamaño de batch* fijo de 128 entradas por *batch*. También utiliza los datos MNIST extraídos de Keras que procesamos anteriormente.

```
def create_dense(layer_sizes):
    model = Sequential()
    #from tensorflow.keras import regularizers
    #kernel_regularizer=regularizers.L2(1e-4)
    model.add(Dense(layer_sizes[0], activation='sigmoid',
input_shape=(image_size,)))#aqui añadir kernel_regularizer

    for s in layer_sizes[1:]:
        model.add(Dense(units = s, activation = 'sigmoid')) #aqui
añadir kernel_regularizer

    model.add(Dense(units=num_classes, activation='softmax'))
    plot_model(model, to_file='model_plot.png', show_shapes=True,
show_layer_names=True)
    return model

def evaluate(model, batch_size=128, epochs=5, verbose=False):
    model.summary()
    model.compile(optimizer='sgd', loss='categorical_crossentropy',
metrics=['accuracy']) #accuracy = 1 - error
    history = model.fit(x_train, y_train, batch_size=batch_size,
epochs=epochs, validation_split=.1, verbose=verbose) #entrenando
    loss, accuracy  = model.evaluate(x_test, y_test, verbose=False)
#YA NO ENTRENA PERO EVALUA EN EL CONJ DE TEST.

    plt.plot(history.history['accuracy'])
    plt.plot(history.history['val_accuracy'])
    plt.title('model accuracy')
    plt.ylabel('accuracy')
    plt.xlabel('epoch')
    plt.legend(['training', 'validation'], loc='best')
    plt.show()

    print()
    print(f'Test loss: {loss:.3}')
    print(f'Test accuracy: {accuracy:.3}')
    print(f'Shape of my predictions (test set):
{mypredictions.shape}')
    yidx = np.argmax(y_test, axis=1) #from one hot encoding to
integers
    mypidx = mypredictions[1,:]
    np.set_printoptions(precision=3, suppress=True)
    print(f'First prediction for number {yidx[1]}, probabilities:
{mypidx}')
```

Ejemplo de uso de las funciones creadas

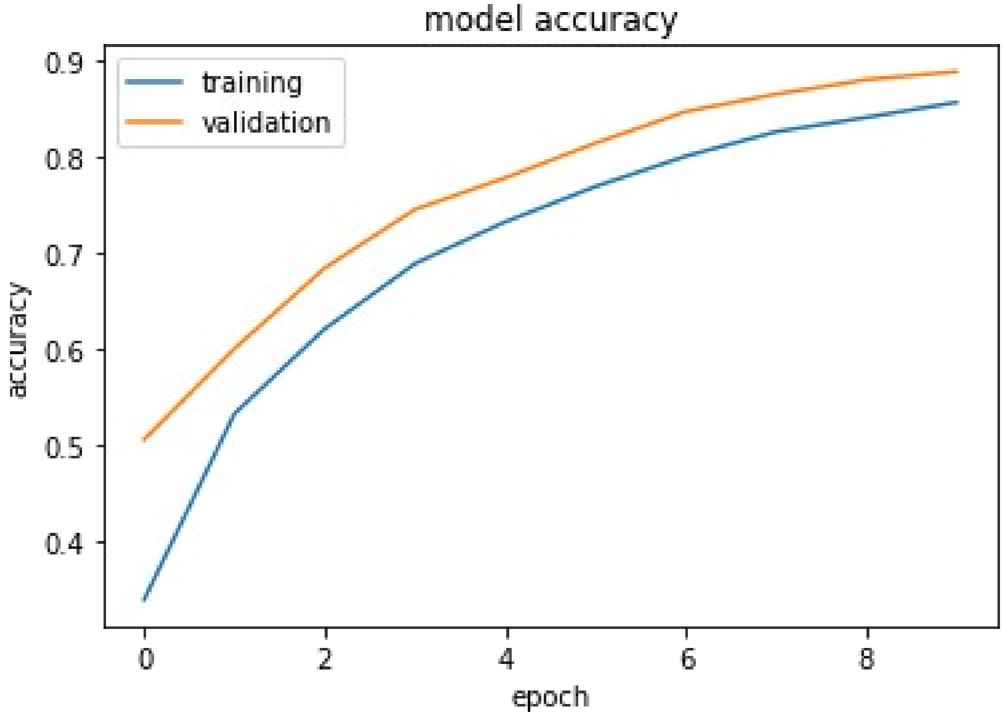
A continuacion se muestra un ejemplo de como usar las anteriores funciones. El lazo `for` genera 2 iteraciones. En la primera iteración, layers = 1 y se genera un modelo con 2 capas de 32 nodos cada una. En la segunda iteración, layers = 2 y se genera un modelo con 4 capas de 32 nodos debido a que se repite dos veces la matriz [32, 32] * 2 = [32, 32, 32, 32].

Para la evaluacion del modelo se usa la funcion `evaluate` con los parametros batch_size=128 y epochs=10.

```
for layers in [1, 2]:
    #print(i)
    model = create_dense([32, 32] * layers)
    evaluate(model, batch_size=128, epochs=10, verbose=True) #verbose
por defecto es false

#EQUIVALENTE del for
#model = create_dense([32, 32]) #2 hidden layers de 32 nodos cada una
#evaluate(model, batch_size=128, epochs=10, verbose=True)

#model = create_dense([32, 32, 32, 32]) #4 hidden layers de 32 nodos
cada una
#evaluate(model, batch_size=128, epochs=10, verbose=True)
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_2 (Dense)             (None, 32)                25120

 dense_3 (Dense)             (None, 32)                1056

 dense_4 (Dense)             (None, 10)                330

=================================================================
Total params: 26,506
Trainable params: 26,506
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 2.1547
- accuracy: 0.3389 - val_loss: 2.0253 - val_accuracy: 0.5055
Epoch 2/10
422/422 [==============================] - 2s 4ms/step - loss: 1.9355
- accuracy: 0.5321 - val_loss: 1.8238 - val_accuracy: 0.5998
```

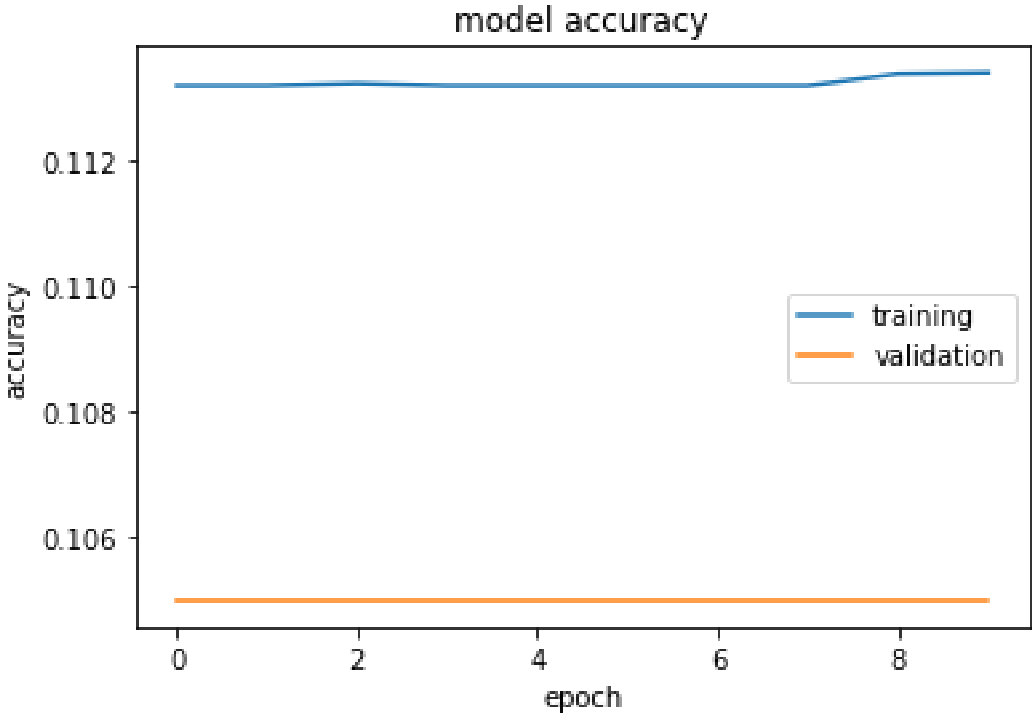
```
Epoch 3/10
422/422 [==============================] - 2s 4ms/step - loss: 1.7432
- accuracy: 0.6204 - val_loss: 1.6270 - val_accuracy: 0.6837
Epoch 4/10
422/422 [==============================] - 2s 4ms/step - loss: 1.5538
- accuracy: 0.6883 - val_loss: 1.4347 - val_accuracy: 0.7445
Epoch 5/10
422/422 [==============================] - 2s 4ms/step - loss: 1.3718
- accuracy: 0.7316 - val_loss: 1.2538 - val_accuracy: 0.7775
Epoch 6/10
422/422 [==============================] - 1s 3ms/step - loss: 1.2073
- accuracy: 0.7685 - val_loss: 1.0983 - val_accuracy: 0.8137
Epoch 7/10
422/422 [==============================] - 2s 5ms/step - loss: 1.0655
- accuracy: 0.8000 - val_loss: 0.9655 - val_accuracy: 0.8465
Epoch 8/10
422/422 [==============================] - 2s 4ms/step - loss: 0.9486
- accuracy: 0.8255 - val_loss: 0.8493 - val_accuracy: 0.8647
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 0.8515
- accuracy: 0.8404 - val_loss: 0.7647 - val_accuracy: 0.8795
Epoch 10/10
422/422 [==============================] - 2s 4ms/step - loss: 0.7715
- accuracy: 0.8561 - val_loss: 0.6929 - val_accuracy: 0.8880
```



```
Test loss: 0.727
Test accuracy: 0.868
```

```
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_5 (Dense)             (None, 32)                25120     
                                                                 
 dense_6 (Dense)             (None, 32)                1056      
                                                                 
 dense_7 (Dense)             (None, 32)                1056      
                                                                 
 dense_8 (Dense)             (None, 32)                1056      
                                                                 
 dense_9 (Dense)             (None, 10)                330       
                                                                 
=================================================================
Total params: 28,618
Trainable params: 28,618
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 4ms/step - loss: 2.3230
- accuracy: 0.1132 - val_loss: 2.2986 - val_accuracy: 0.1050
Epoch 2/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2973
- accuracy: 0.1132 - val_loss: 2.2972 - val_accuracy: 0.1050
Epoch 3/10
422/422 [==============================] - 3s 6ms/step - loss: 2.2957
- accuracy: 0.1132 - val_loss: 2.2954 - val_accuracy: 0.1050
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 2.2939
- accuracy: 0.1132 - val_loss: 2.2936 - val_accuracy: 0.1050
Epoch 5/10
422/422 [==============================] - 2s 5ms/step - loss: 2.2922
- accuracy: 0.1132 - val_loss: 2.2914 - val_accuracy: 0.1050
Epoch 6/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2902
- accuracy: 0.1132 - val_loss: 2.2891 - val_accuracy: 0.1050
Epoch 7/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2879
- accuracy: 0.1132 - val_loss: 2.2866 - val_accuracy: 0.1050
Epoch 8/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2852
- accuracy: 0.1132 - val_loss: 2.2836 - val_accuracy: 0.1050
Epoch 9/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2821
- accuracy: 0.1134 - val_loss: 2.2803 - val_accuracy: 0.1050
Epoch 10/10
```

```
422/422 [==============================] - 3s 6ms/step - loss: 2.2783
- accuracy: 0.1134 - val_loss: 2.2755 - val_accuracy: 0.1050
```



```
Test loss: 2.28
Test accuracy: 0.113
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

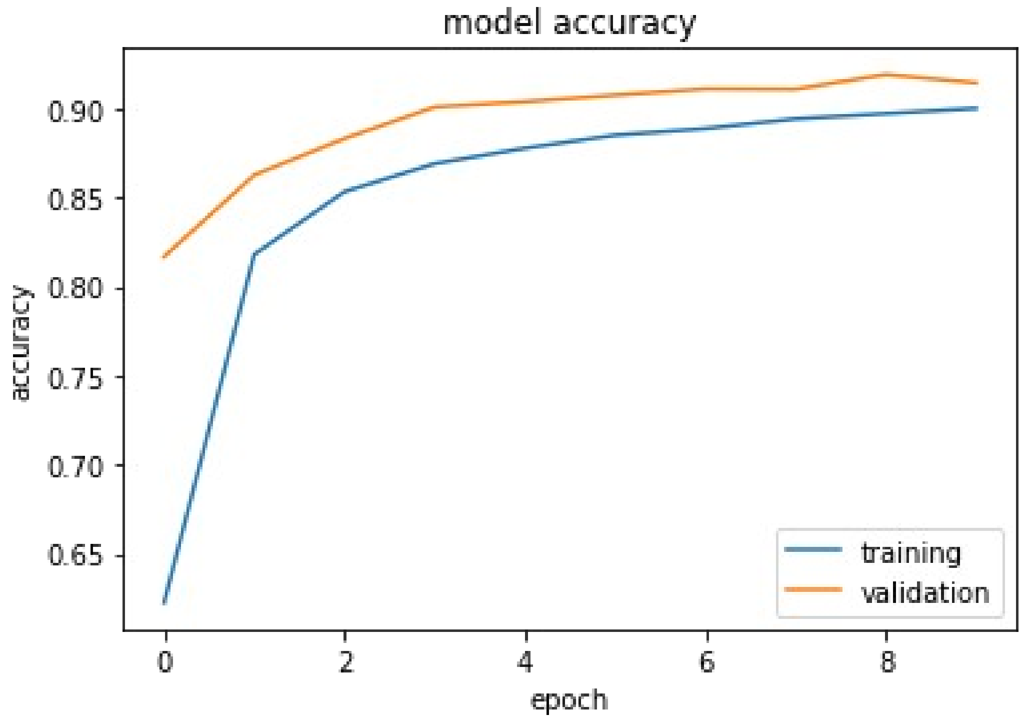
1. Comparar redes más complejas (4 puntos)

- Ahora entrene y evalúe modelos con **diferente números de capas** ocultas. Todas las capas ocultas deben tener 32 nodos. El primer modelo tiene 1 capa oculta, el segundo 2 ... hasta cuatro capas. Analice la exactitud obtenida en cada caso.

Evalue el modelo con los parametros por defecto

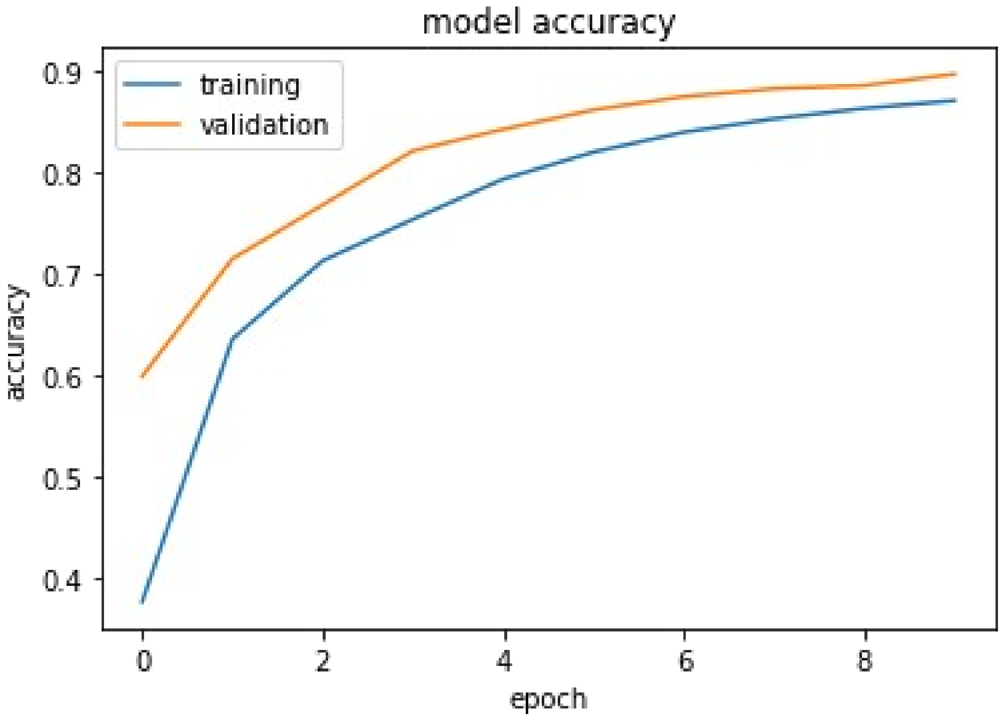
```
#ESCRIBA SU CÓDIGO AQUÍ.
#ejemplo para 4 capas ocultas de 32 nodos
#model = create_dense([32, 32, 32, 32]) #4 hidden layers de 32 nodos
cada una
#evaluate(model, batch_size=128, epochs=100, verbose=True)

for layers in [1, 2, 3, 4]:
    #print(i)
    model = create_dense([32] * layers)
    evaluate(model, batch_size=128, epochs=10, verbose=True) #verbose
por defecto es false
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_10 (Dense)            (None, 32)                25120     
                                                                 
 dense_11 (Dense)            (None, 10)                330       
                                                                 
=================================================================
Total params: 25,450
Trainable params: 25,450
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 1.4687
- accuracy: 0.6226 - val_loss: 0.9988 - val_accuracy: 0.8167
Epoch 2/10
422/422 [==============================] - 1s 3ms/step - loss: 0.9011
- accuracy: 0.8179 - val_loss: 0.7333 - val_accuracy: 0.8625
Epoch 3/10
422/422 [==============================] - 1s 3ms/step - loss: 0.7144
- accuracy: 0.8531 - val_loss: 0.6083 - val_accuracy: 0.8830
Epoch 4/10
422/422 [==============================] - 2s 4ms/step - loss: 0.6125
- accuracy: 0.8689 - val_loss: 0.5129 - val_accuracy: 0.9007
Epoch 5/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5436
- accuracy: 0.8775 - val_loss: 0.4595 - val_accuracy: 0.9037
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4966
- accuracy: 0.8849 - val_loss: 0.4181 - val_accuracy: 0.9073
Epoch 7/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4601
- accuracy: 0.8886 - val_loss: 0.3969 - val_accuracy: 0.9108
Epoch 8/10
422/422 [==============================] - 1s 3ms/step - loss: 0.4341
- accuracy: 0.8940 - val_loss: 0.3701 - val_accuracy: 0.9107
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 0.4147
- accuracy: 0.8970 - val_loss: 0.3458 - val_accuracy: 0.9188
Epoch 10/10
422/422 [==============================] - 1s 3ms/step - loss: 0.3972
- accuracy: 0.8999 - val_loss: 0.3451 - val_accuracy: 0.9142
```



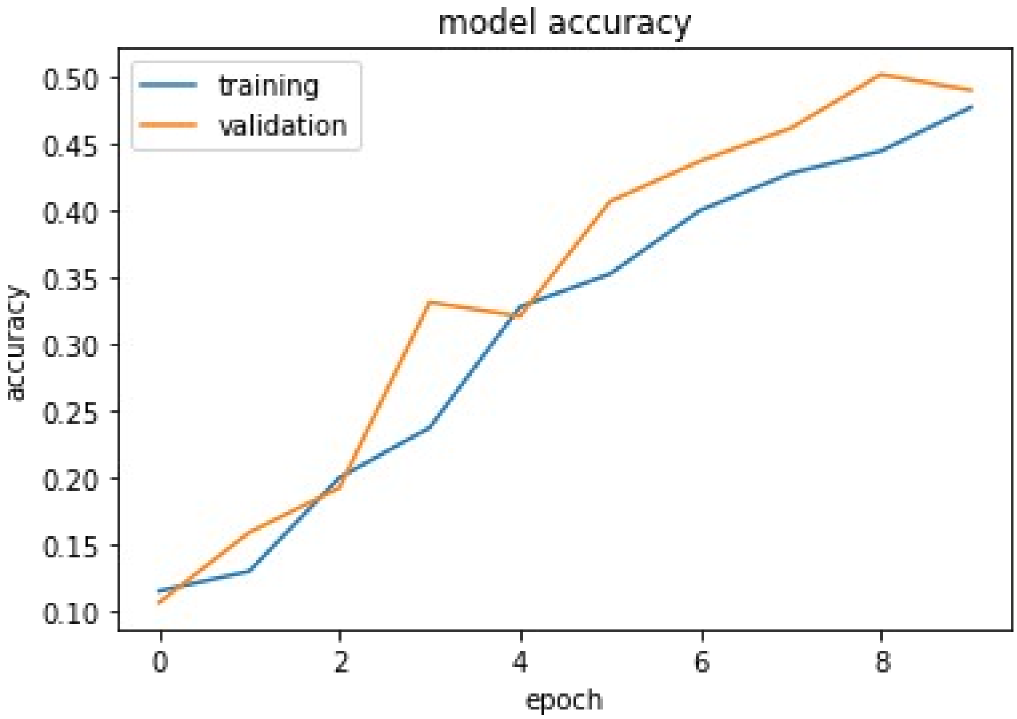
```
Test loss: 0.394
Test accuracy: 0.898
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_12 (Dense)            (None, 32)                25120     
                                                                 
 dense_13 (Dense)            (None, 32)                1056      
                                                                 
 dense_14 (Dense)            (None, 10)                330       
                                                                 
=================================================================
Total params: 26,506
Trainable params: 26,506
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 2.1828
- accuracy: 0.3756 - val_loss: 2.0203 - val_accuracy: 0.5987
Epoch 2/10
422/422 [==============================] - 1s 3ms/step - loss: 1.9153
- accuracy: 0.6353 - val_loss: 1.7818 - val_accuracy: 0.7147
Epoch 3/10
```

```
422/422 [==============================] - 2s 4ms/step - loss: 1.6805
- accuracy: 0.7128 - val_loss: 1.5385 - val_accuracy: 0.7680
Epoch 4/10
422/422 [==============================] - 2s 6ms/step - loss: 1.4486
- accuracy: 0.7537 - val_loss: 1.3095 - val_accuracy: 0.8215
Epoch 5/10
422/422 [==============================] - 1s 3ms/step - loss: 1.2402
- accuracy: 0.7936 - val_loss: 1.1151 - val_accuracy: 0.8428
Epoch 6/10
422/422 [==============================] - 1s 4ms/step - loss: 1.0678
- accuracy: 0.8203 - val_loss: 0.9544 - val_accuracy: 0.8620
Epoch 7/10
422/422 [==============================] - 1s 3ms/step - loss: 0.9297
- accuracy: 0.8399 - val_loss: 0.8311 - val_accuracy: 0.8752
Epoch 8/10
422/422 [==============================] - 1s 3ms/step - loss: 0.8236
- accuracy: 0.8532 - val_loss: 0.7322 - val_accuracy: 0.8830
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 0.7392
- accuracy: 0.8636 - val_loss: 0.6612 - val_accuracy: 0.8862
Epoch 10/10
422/422 [==============================] - 1s 3ms/step - loss: 0.6754
- accuracy: 0.8712 - val_loss: 0.5955 - val_accuracy: 0.8975
```



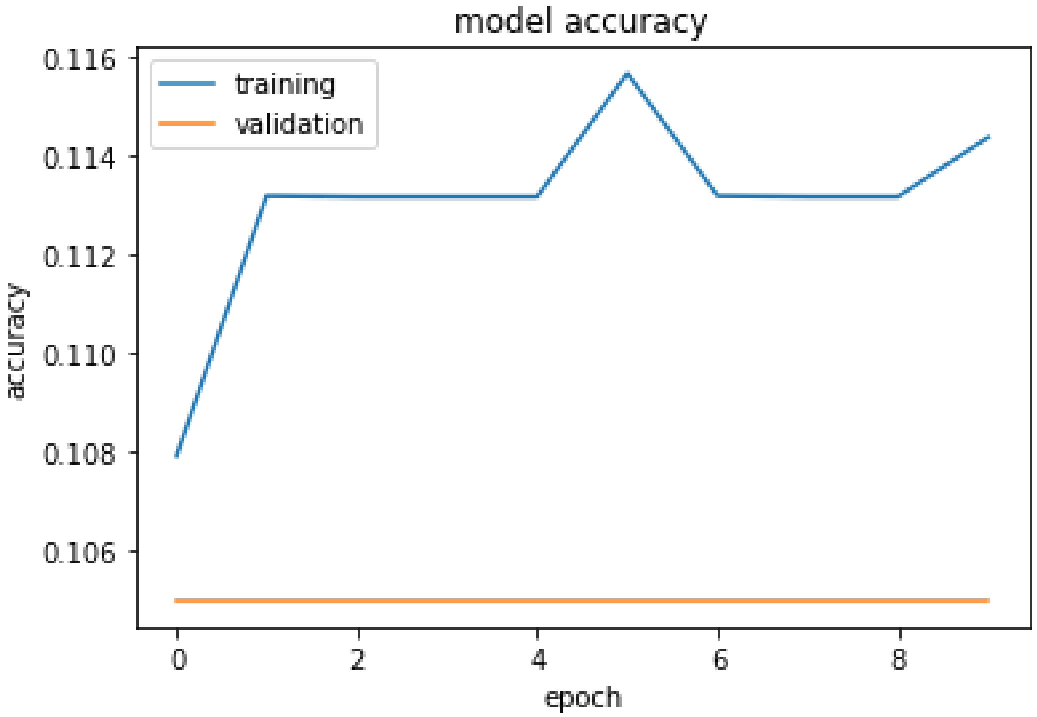
```
Test loss: 0.634
Test accuracy: 0.88
Shape of my predictions (test set): (10000, 10)
```

```
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_15 (Dense)            (None, 32)                25120

 dense_16 (Dense)            (None, 32)                1056

 dense_17 (Dense)            (None, 32)                1056

 dense_18 (Dense)            (None, 10)                330

=================================================================
Total params: 27,562
Trainable params: 27,562
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 2.3169
- accuracy: 0.1142 - val_loss: 2.2852 - val_accuracy: 0.1055
Epoch 2/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2777
- accuracy: 0.1290 - val_loss: 2.2686 - val_accuracy: 0.1582
Epoch 3/10
422/422 [==============================] - 1s 4ms/step - loss: 2.2611
- accuracy: 0.1994 - val_loss: 2.2514 - val_accuracy: 0.1917
Epoch 4/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2419
- accuracy: 0.2365 - val_loss: 2.2288 - val_accuracy: 0.3307
Epoch 5/10
422/422 [==============================] - 2s 5ms/step - loss: 2.2166
- accuracy: 0.3272 - val_loss: 2.1995 - val_accuracy: 0.3202
Epoch 6/10
422/422 [==============================] - 2s 4ms/step - loss: 2.1814
- accuracy: 0.3520 - val_loss: 2.1580 - val_accuracy: 0.4063
Epoch 7/10
422/422 [==============================] - 1s 3ms/step - loss: 2.1342
- accuracy: 0.3997 - val_loss: 2.1046 - val_accuracy: 0.4367
Epoch 8/10
422/422 [==============================] - 1s 3ms/step - loss: 2.0743
- accuracy: 0.4273 - val_loss: 2.0368 - val_accuracy: 0.4612
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 2.0021
- accuracy: 0.4440 - val_loss: 1.9583 - val_accuracy: 0.5013
Epoch 10/10
422/422 [==============================] - 1s 3ms/step - loss: 1.9211
- accuracy: 0.4769 - val_loss: 1.8727 - val_accuracy: 0.4897
```



```
Test loss: 1.88
Test accuracy: 0.48
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_6"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_19 (Dense)            (None, 32)                25120

 dense_20 (Dense)            (None, 32)                1056

 dense_21 (Dense)            (None, 32)                1056

 dense_22 (Dense)            (None, 32)                1056

 dense_23 (Dense)            (None, 10)                330

=================================================================
Total params: 28,618
Trainable params: 28,618
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 2.3364
- accuracy: 0.1079 - val_loss: 2.3000 - val_accuracy: 0.1050
```

```
Epoch 2/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2984
- accuracy: 0.1132 - val_loss: 2.2979 - val_accuracy: 0.1050
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 2.2967
- accuracy: 0.1132 - val_loss: 2.2961 - val_accuracy: 0.1050
Epoch 4/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2948
- accuracy: 0.1132 - val_loss: 2.2941 - val_accuracy: 0.1050
Epoch 5/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2931
- accuracy: 0.1132 - val_loss: 2.2921 - val_accuracy: 0.1050
Epoch 6/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2911
- accuracy: 0.1157 - val_loss: 2.2902 - val_accuracy: 0.1050
Epoch 7/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2890
- accuracy: 0.1132 - val_loss: 2.2877 - val_accuracy: 0.1050
Epoch 8/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2868
- accuracy: 0.1132 - val_loss: 2.2855 - val_accuracy: 0.1050
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 2.2840
- accuracy: 0.1132 - val_loss: 2.2825 - val_accuracy: 0.1050
Epoch 10/10
422/422 [==============================] - 2s 4ms/step - loss: 2.2808
- accuracy: 0.1144 - val_loss: 2.2788 - val_accuracy: 0.1050
```



```
Test loss: 2.28
Test accuracy: 0.113
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Análisis de la exactitud obtenida en cada caso

- **Caso de 1 capa de 32 nodos:** Exactitud de 0.898
- **Caso de 2 capas de 32 nodos:** Exactitud de 0.88
- **Caso de 3 capas de 32 nodos:** Exactitud de 0.48
- **Caso de 4 capas de 32 nodos:** Exactitud de 0.113

Como se puede ver, la exactitud va bajando conforme se aumentan capas. Esto puede deberse al bajo número de epochs que se está utilizando.

2. Redes más profundas tardan más en entrenar (4 puntos)

Segun lo observado en el ejemplo anterior, las redes más profundas toman más tiempo para entrenar. Esto tiene que ver con la retropropagación (backpropagation), el descenso de gradiente y la forma en que funcionan los algoritmos de optimización: esos detalles están más allá del alcance de este ejercicio. Sin embargo, tenga en cuenta lo que sucede cuando dejamos que la red anterior de 3 capas ocultas, que tenía un rendimiento mediocre, entrene por más tiempo. Para esto, realice lo siguiente

- Cree una red con 3 capas ocultas de 32 nodos ([32, 32, 32]) pero esta vez entrene durante 40 épocas. Qué sucedió? Comente sus resultados.

Mantenga el resto de parámetros por defecto. Puede usar la opción `verbose=True` para llamar a la función `evaluate` para ver en pantalla los resultados por época. Discuta sus resultados.

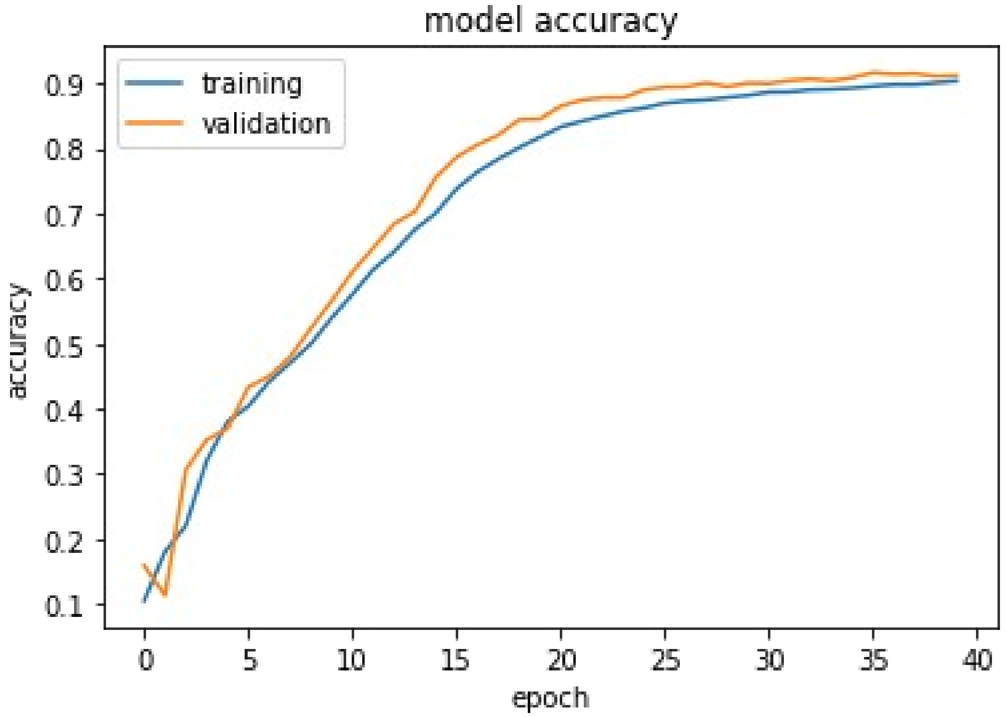
```
model = create_dense([32] * 3)
evaluate(model, batch_size=128, epochs=40, verbose=True) #verbose por
defecto es false
```

```
Model: "sequential_7"

_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_24 (Dense)            (None, 32)                25120     
                                                                 
 dense_25 (Dense)            (None, 32)                1056      
                                                                 
 dense_26 (Dense)            (None, 32)                1056      
                                                                 
 dense_27 (Dense)            (None, 10)                330       
```

```
=================================================================
Total params: 27,562
Trainable params: 27,562
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 3s 5ms/step - loss: 2.3068
- accuracy: 0.1051 - val_loss: 2.2748 - val_accuracy: 0.1590
Epoch 2/40
422/422 [==============================] - 1s 3ms/step - loss: 2.2649
- accuracy: 0.1800 - val_loss: 2.2528 - val_accuracy: 0.1130
Epoch 3/40
422/422 [==============================] - 1s 4ms/step - loss: 2.2425
- accuracy: 0.2205 - val_loss: 2.2272 - val_accuracy: 0.3062
Epoch 4/40
422/422 [==============================] - 2s 6ms/step - loss: 2.2158
- accuracy: 0.3201 - val_loss: 2.1966 - val_accuracy: 0.3520
Epoch 5/40
422/422 [==============================] - 3s 6ms/step - loss: 2.1813
- accuracy: 0.3799 - val_loss: 2.1557 - val_accuracy: 0.3690
Epoch 6/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1350
- accuracy: 0.4036 - val_loss: 2.1012 - val_accuracy: 0.4330
Epoch 7/40
422/422 [==============================] - 1s 3ms/step - loss: 2.0744
- accuracy: 0.4414 - val_loss: 2.0311 - val_accuracy: 0.4497
Epoch 8/40
422/422 [==============================] - 1s 3ms/step - loss: 1.9979
- accuracy: 0.4699 - val_loss: 1.9443 - val_accuracy: 0.4793
Epoch 9/40
422/422 [==============================] - 2s 4ms/step - loss: 1.9064
- accuracy: 0.4993 - val_loss: 1.8442 - val_accuracy: 0.5232
Epoch 10/40
422/422 [==============================] - 1s 3ms/step - loss: 1.8031
- accuracy: 0.5392 - val_loss: 1.7327 - val_accuracy: 0.5653
Epoch 11/40
422/422 [==============================] - 1s 3ms/step - loss: 1.6895
- accuracy: 0.5756 - val_loss: 1.6114 - val_accuracy: 0.6102
Epoch 12/40
422/422 [==============================] - 2s 4ms/step - loss: 1.5693
- accuracy: 0.6139 - val_loss: 1.4865 - val_accuracy: 0.6467
Epoch 13/40
422/422 [==============================] - 2s 5ms/step - loss: 1.4492
- accuracy: 0.6408 - val_loss: 1.3646 - val_accuracy: 0.6838
Epoch 14/40
422/422 [==============================] - 1s 3ms/step - loss: 1.3342
- accuracy: 0.6749 - val_loss: 1.2493 - val_accuracy: 0.7023
Epoch 15/40
422/422 [==============================] - 1s 3ms/step - loss: 1.2301
- accuracy: 0.7001 - val_loss: 1.1490 - val_accuracy: 0.7552
```

```
Epoch 16/40
422/422 [==============================] - 1s 3ms/step - loss: 1.1381
- accuracy: 0.7375 - val_loss: 1.0585 - val_accuracy: 0.7863
Epoch 17/40
422/422 [==============================] - 1s 3ms/step - loss: 1.0590
- accuracy: 0.7636 - val_loss: 0.9842 - val_accuracy: 0.8053
Epoch 18/40
422/422 [==============================] - 1s 3ms/step - loss: 0.9888
- accuracy: 0.7832 - val_loss: 0.9161 - val_accuracy: 0.8200
Epoch 19/40
422/422 [==============================] - 1s 3ms/step - loss: 0.9272
- accuracy: 0.8011 - val_loss: 0.8591 - val_accuracy: 0.8440
Epoch 20/40
422/422 [==============================] - 1s 3ms/step - loss: 0.8762
- accuracy: 0.8169 - val_loss: 0.8093 - val_accuracy: 0.8448
Epoch 21/40
422/422 [==============================] - 2s 5ms/step - loss: 0.8267
- accuracy: 0.8328 - val_loss: 0.7569 - val_accuracy: 0.8650
Epoch 22/40
422/422 [==============================] - 1s 3ms/step - loss: 0.7811
- accuracy: 0.8416 - val_loss: 0.7147 - val_accuracy: 0.8742
Epoch 23/40
422/422 [==============================] - 2s 5ms/step - loss: 0.7433
- accuracy: 0.8493 - val_loss: 0.6828 - val_accuracy: 0.8777
Epoch 24/40
422/422 [==============================] - 2s 5ms/step - loss: 0.7069
- accuracy: 0.8569 - val_loss: 0.6460 - val_accuracy: 0.8777
Epoch 25/40
422/422 [==============================] - 2s 5ms/step - loss: 0.6758
- accuracy: 0.8621 - val_loss: 0.6205 - val_accuracy: 0.8898
Epoch 26/40
422/422 [==============================] - 2s 4ms/step - loss: 0.6435
- accuracy: 0.8693 - val_loss: 0.5878 - val_accuracy: 0.8940
Epoch 27/40
422/422 [==============================] - 2s 5ms/step - loss: 0.6165
- accuracy: 0.8726 - val_loss: 0.5599 - val_accuracy: 0.8945
Epoch 28/40
422/422 [==============================] - 2s 5ms/step - loss: 0.5929
- accuracy: 0.8744 - val_loss: 0.5277 - val_accuracy: 0.9005
Epoch 29/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5712
- accuracy: 0.8781 - val_loss: 0.5136 - val_accuracy: 0.8947
Epoch 30/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5485
- accuracy: 0.8814 - val_loss: 0.4930 - val_accuracy: 0.9003
Epoch 31/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5282
- accuracy: 0.8864 - val_loss: 0.4817 - val_accuracy: 0.8997
Epoch 32/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5098
```

```
- accuracy: 0.8867 - val_loss: 0.4510 - val_accuracy: 0.9038
Epoch 33/40
422/422 [==============================] - 2s 4ms/step - loss: 0.4932
- accuracy: 0.8899 - val_loss: 0.4424 - val_accuracy: 0.9063
Epoch 34/40
422/422 [==============================] - 1s 4ms/step - loss: 0.4777
- accuracy: 0.8906 - val_loss: 0.4329 - val_accuracy: 0.9035
Epoch 35/40
422/422 [==============================] - 2s 5ms/step - loss: 0.4630
- accuracy: 0.8925 - val_loss: 0.4137 - val_accuracy: 0.9083
Epoch 36/40
422/422 [==============================] - 2s 4ms/step - loss: 0.4496
- accuracy: 0.8954 - val_loss: 0.3958 - val_accuracy: 0.9162
Epoch 37/40
422/422 [==============================] - 1s 3ms/step - loss: 0.4358
- accuracy: 0.8980 - val_loss: 0.3886 - val_accuracy: 0.9138
Epoch 38/40
422/422 [==============================] - 1s 3ms/step - loss: 0.4273
- accuracy: 0.8979 - val_loss: 0.3735 - val_accuracy: 0.9148
Epoch 39/40
422/422 [==============================] - 1s 3ms/step - loss: 0.4182
- accuracy: 0.9007 - val_loss: 0.3729 - val_accuracy: 0.9103
Epoch 40/40
422/422 [==============================] - 1s 3ms/step - loss: 0.4049
- accuracy: 0.9034 - val_loss: 0.3739 - val_accuracy: 0.9113
```



```
Test loss: 0.41
Test accuracy: 0.899
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Discusión de resultados

Al aumentar el número de epochs de 10 a 40, el tiempo de ejecución de la capa aumenta considerablemente, pero en lo obtenido se puede observar dos cosas:

- **El loss obtenido es mucho menor, pasando de** 2.3068 **a** 0.41
- **El accuracy obtenido es mucho mayor, pasando de** 0.1051 **a** 0.899

3. Comparación del número de nodos (i.e., unidades o neuronas) por capa (4 puntos)

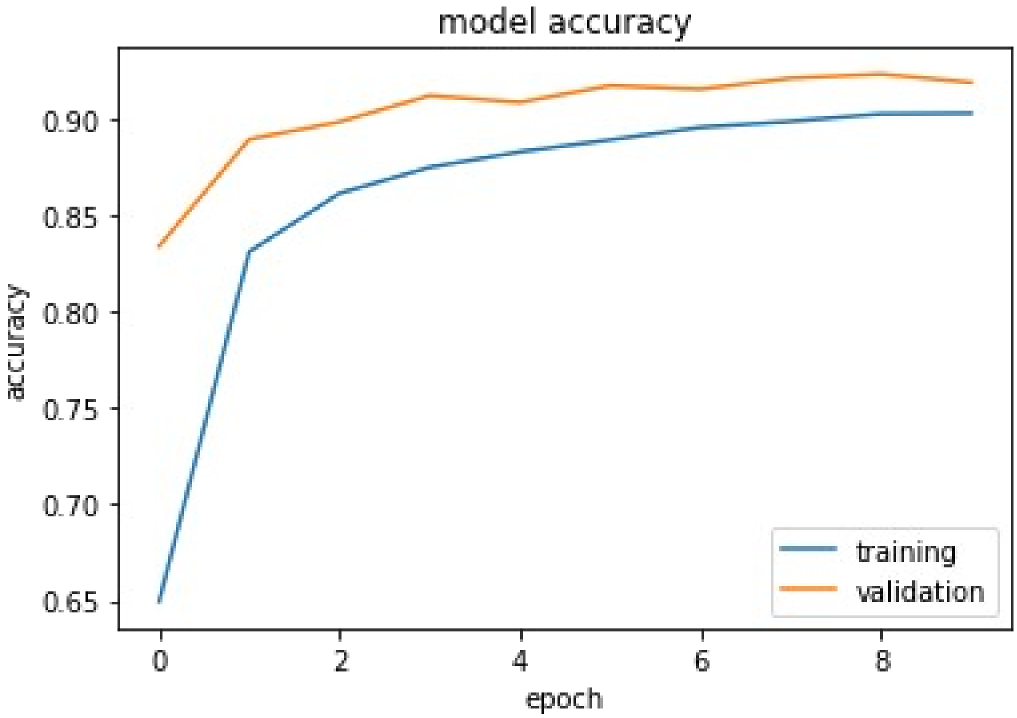
Otra forma de incrementar la complejidad es agregar más nodos **(i.e., unidades o neuronas)** a cada capa oculta. Cree varias redes neuronales de una capa, con cada vez más nodos en esa capa. Pruebe con 32, 64, 128, 256, 512, 1024 y 2048 nodos. Comente sus resultados.

Note que esta vez al crear el modelo usando `create_dense`, el lazo `for` iterara sobre el numero de nodos [32, 64, 128, 256, 512, 1024, 2048]

```
for nodes in [32, 64, 128, 256, 512, 1024, 2048]:
    #print(i)
    model = create_dense([nodes] * 1)
    evaluate(model, batch_size=128, epochs=10, verbose=True) #verbose
por defecto es false
```

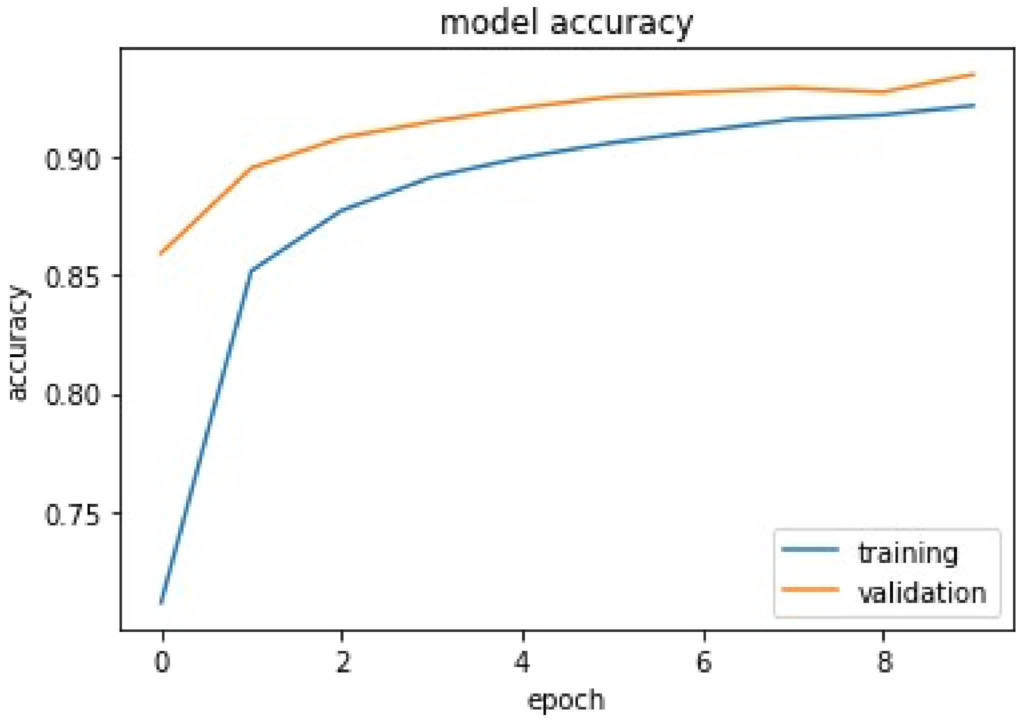
```
Model: "sequential_8"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_28 (Dense)            (None, 32)                25120     
                                                                 
 dense_29 (Dense)            (None, 10)                330       
                                                                 
=================================================================
Total params: 25,450
Trainable params: 25,450
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 6ms/step - loss: 1.3690
- accuracy: 0.6495 - val_loss: 0.8933 - val_accuracy: 0.8340
Epoch 2/10
422/422 [==============================] - 1s 3ms/step - loss: 0.8217
- accuracy: 0.8311 - val_loss: 0.6339 - val_accuracy: 0.8895
Epoch 3/10
```

```
422/422 [==============================] - 1s 3ms/step - loss: 0.6522
- accuracy: 0.8614 - val_loss: 0.5251 - val_accuracy: 0.8985
Epoch 4/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5631
- accuracy: 0.8749 - val_loss: 0.4654 - val_accuracy: 0.9122
Epoch 5/10
422/422 [==============================] - 1s 3ms/step - loss: 0.5061
- accuracy: 0.8830 - val_loss: 0.4283 - val_accuracy: 0.9087
Epoch 6/10
422/422 [==============================] - 2s 4ms/step - loss: 0.4653
- accuracy: 0.8892 - val_loss: 0.3898 - val_accuracy: 0.9173
Epoch 7/10
422/422 [==============================] - 1s 3ms/step - loss: 0.4377
- accuracy: 0.8957 - val_loss: 0.3695 - val_accuracy: 0.9155
Epoch 8/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4130
- accuracy: 0.8988 - val_loss: 0.3471 - val_accuracy: 0.9212
Epoch 9/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3950
- accuracy: 0.9027 - val_loss: 0.3287 - val_accuracy: 0.9233
Epoch 10/10
422/422 [==============================] - 1s 3ms/step - loss: 0.3833
- accuracy: 0.9030 - val_loss: 0.3288 - val_accuracy: 0.9190
```



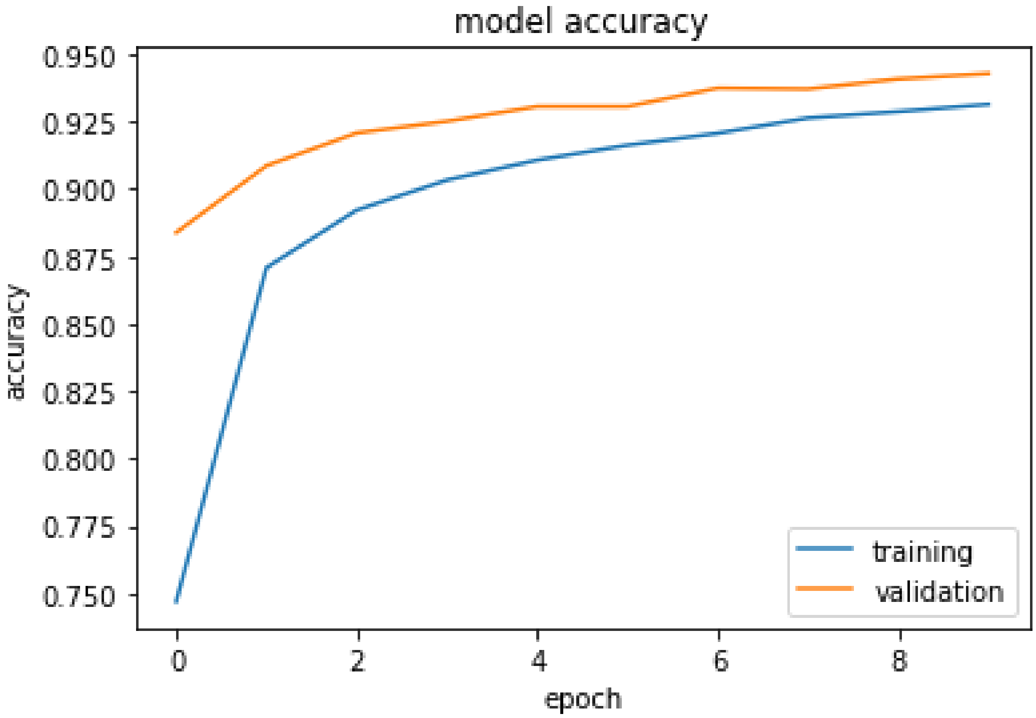
```
Test loss: 0.376
Test accuracy: 0.906
Shape of my predictions (test set): (10000, 10)
```

```
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_9"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_30 (Dense)            (None, 64)                50240

 dense_31 (Dense)            (None, 10)                650

=================================================================
Total params: 50,890
Trainable params: 50,890
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 1.1490
- accuracy: 0.7115 - val_loss: 0.6821 - val_accuracy: 0.8592
Epoch 2/10
422/422 [==============================] - 2s 4ms/step - loss: 0.6401
- accuracy: 0.8517 - val_loss: 0.4928 - val_accuracy: 0.8952
Epoch 3/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5113
- accuracy: 0.8772 - val_loss: 0.4063 - val_accuracy: 0.9080
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4440
- accuracy: 0.8913 - val_loss: 0.3649 - val_accuracy: 0.9148
Epoch 5/10
422/422 [==============================] - 2s 6ms/step - loss: 0.4003
- accuracy: 0.8996 - val_loss: 0.3301 - val_accuracy: 0.9207
Epoch 6/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3690
- accuracy: 0.9059 - val_loss: 0.3077 - val_accuracy: 0.9253
Epoch 7/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3457
- accuracy: 0.9108 - val_loss: 0.2909 - val_accuracy: 0.9273
Epoch 8/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3281
- accuracy: 0.9158 - val_loss: 0.2746 - val_accuracy: 0.9290
Epoch 9/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3153
- accuracy: 0.9176 - val_loss: 0.2701 - val_accuracy: 0.9273
Epoch 10/10
422/422 [==============================] - 2s 4ms/step - loss: 0.2994
- accuracy: 0.9216 - val_loss: 0.2598 - val_accuracy: 0.9347
```



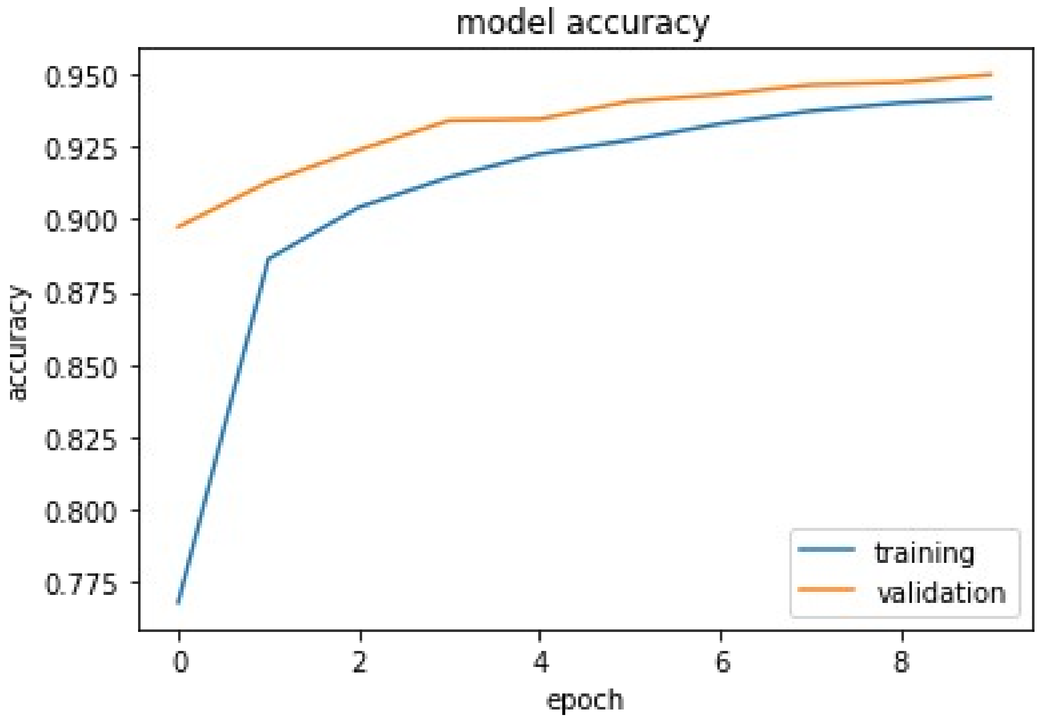
```
Test loss: 0.299
Test accuracy: 0.921
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_32 (Dense)            (None, 128)               100480

 dense_33 (Dense)            (None, 10)                1290

=================================================================
Total params: 101,770
Trainable params: 101,770
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 4s 7ms/step - loss: 1.0030
- accuracy: 0.7470 - val_loss: 0.5463 - val_accuracy: 0.8838
Epoch 2/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5307
- accuracy: 0.8708 - val_loss: 0.3946 - val_accuracy: 0.9087
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4255
- accuracy: 0.8922 - val_loss: 0.3351 - val_accuracy: 0.9208
```

```
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3718
- accuracy: 0.9034 - val_loss: 0.2978 - val_accuracy: 0.9252
Epoch 5/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3365
- accuracy: 0.9108 - val_loss: 0.2759 - val_accuracy: 0.9307
Epoch 6/10
422/422 [==============================] - 4s 10ms/step - loss: 0.3127
- accuracy: 0.9163 - val_loss: 0.2578 - val_accuracy: 0.9307
Epoch 7/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2934
- accuracy: 0.9207 - val_loss: 0.2395 - val_accuracy: 0.9375
Epoch 8/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2755
- accuracy: 0.9264 - val_loss: 0.2337 - val_accuracy: 0.9372
Epoch 9/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2627
- accuracy: 0.9288 - val_loss: 0.2219 - val_accuracy: 0.9408
Epoch 10/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2525
- accuracy: 0.9315 - val_loss: 0.2177 - val_accuracy: 0.9428
```



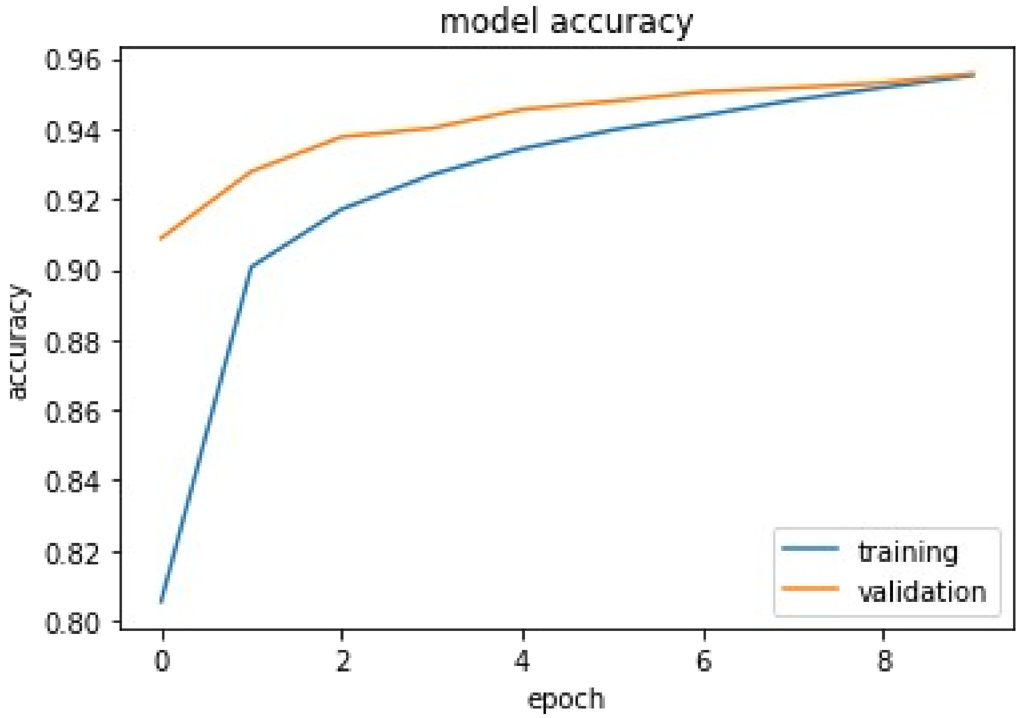
```
Test loss: 0.251
Test accuracy: 0.934
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

```
Model: "sequential_11"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_34 (Dense)            (None, 256)               200960    
                                                                 
 dense_35 (Dense)            (None, 10)                2570      
                                                                 
=================================================================
Total params: 203,530
Trainable params: 203,530
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 6ms/step - loss: 0.8798
- accuracy: 0.7679 - val_loss: 0.4524 - val_accuracy: 0.8973
Epoch 2/10
422/422 [==============================] - 3s 6ms/step - loss: 0.4465
- accuracy: 0.8863 - val_loss: 0.3364 - val_accuracy: 0.9128
Epoch 3/10
422/422 [==============================] - 4s 8ms/step - loss: 0.3605
- accuracy: 0.9041 - val_loss: 0.2896 - val_accuracy: 0.9238
Epoch 4/10
422/422 [==============================] - 2s 6ms/step - loss: 0.3171
- accuracy: 0.9144 - val_loss: 0.2570 - val_accuracy: 0.9340
Epoch 5/10
422/422 [==============================] - 2s 6ms/step - loss: 0.2859
- accuracy: 0.9225 - val_loss: 0.2386 - val_accuracy: 0.9345
Epoch 6/10
422/422 [==============================] - 2s 6ms/step - loss: 0.2639
- accuracy: 0.9273 - val_loss: 0.2259 - val_accuracy: 0.9407
Epoch 7/10
422/422 [==============================] - 2s 6ms/step - loss: 0.2461
- accuracy: 0.9329 - val_loss: 0.2127 - val_accuracy: 0.9430
Epoch 8/10
422/422 [==============================] - 3s 8ms/step - loss: 0.2312
- accuracy: 0.9373 - val_loss: 0.2005 - val_accuracy: 0.9463
Epoch 9/10
422/422 [==============================] - 3s 6ms/step - loss: 0.2209
- accuracy: 0.9401 - val_loss: 0.1962 - val_accuracy: 0.9472
Epoch 10/10
422/422 [==============================] - 2s 6ms/step - loss: 0.2105
- accuracy: 0.9418 - val_loss: 0.1869 - val_accuracy: 0.9498
```



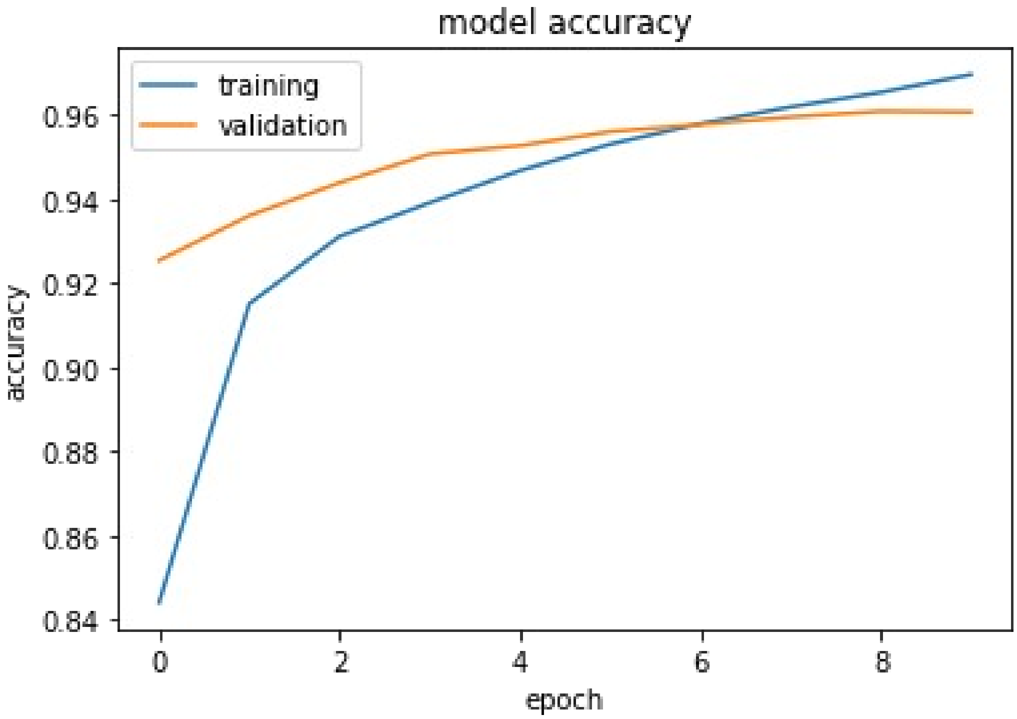
```
Test loss: 0.218
Test accuracy: 0.938
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_12"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_36 (Dense)            (None, 512)               401920

 dense_37 (Dense)            (None, 10)                5130

=================================================================
Total params: 407,050
Trainable params: 407,050
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 8s 18ms/step - loss: 0.7501
- accuracy: 0.8055 - val_loss: 0.3755 - val_accuracy: 0.9088
Epoch 2/10
422/422 [==============================] - 4s 9ms/step - loss: 0.3793
- accuracy: 0.9007 - val_loss: 0.2841 - val_accuracy: 0.9278
Epoch 3/10
422/422 [==============================] - 4s 10ms/step - loss: 0.3091
- accuracy: 0.9171 - val_loss: 0.2455 - val_accuracy: 0.9377
```

```
Epoch 4/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2692
- accuracy: 0.9269 - val_loss: 0.2243 - val_accuracy: 0.9402
Epoch 5/10
422/422 [==============================] - 4s 9ms/step - loss: 0.2424
- accuracy: 0.9343 - val_loss: 0.2098 - val_accuracy: 0.9455
Epoch 6/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2222
- accuracy: 0.9396 - val_loss: 0.1972 - val_accuracy: 0.9478
Epoch 7/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2071
- accuracy: 0.9437 - val_loss: 0.1845 - val_accuracy: 0.9505
Epoch 8/10
422/422 [==============================] - 4s 9ms/step - loss: 0.1923
- accuracy: 0.9481 - val_loss: 0.1774 - val_accuracy: 0.9517
Epoch 9/10
422/422 [==============================] - 4s 9ms/step - loss: 0.1805
- accuracy: 0.9517 - val_loss: 0.1706 - val_accuracy: 0.9530
Epoch 10/10
422/422 [==============================] - 5s 12ms/step - loss: 0.1686
- accuracy: 0.9552 - val_loss: 0.1668 - val_accuracy: 0.9557
```



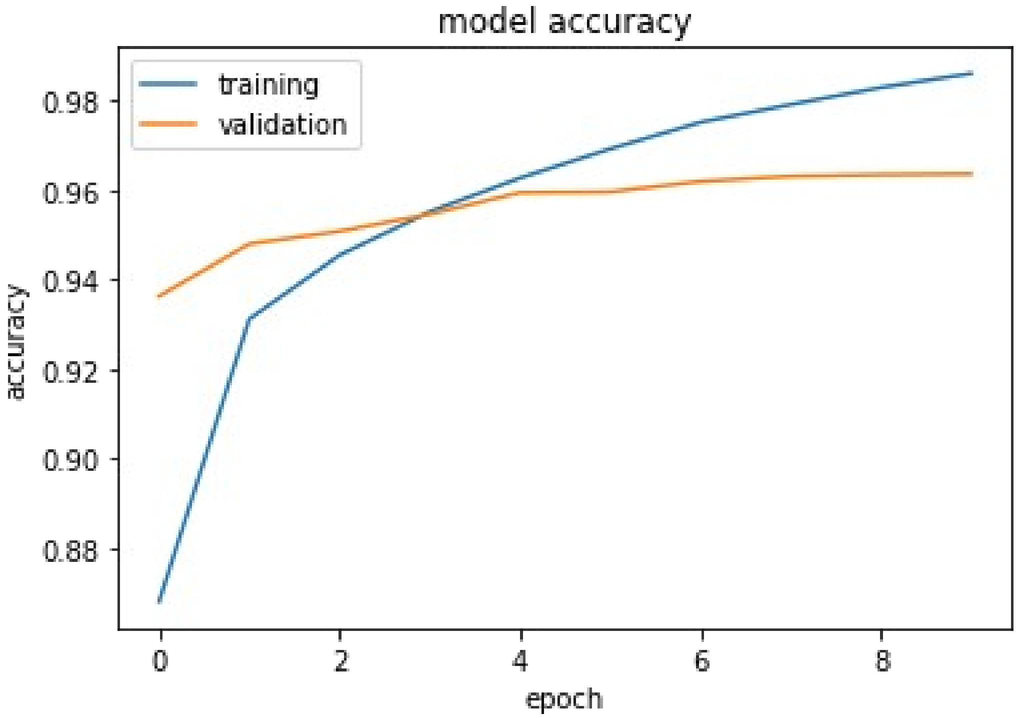
```
Test loss: 0.19
Test accuracy: 0.946
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

```
Model: "sequential_13"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_38 (Dense)            (None, 1024)              803840

 dense_39 (Dense)            (None, 10)                10250

=================================================================
Total params: 814,090
Trainable params: 814,090
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 7s 16ms/step - loss: 0.6061
- accuracy: 0.8440 - val_loss: 0.3093 - val_accuracy: 0.9253
Epoch 2/10
422/422 [==============================] - 7s 16ms/step - loss: 0.3187
- accuracy: 0.9151 - val_loss: 0.2439 - val_accuracy: 0.9360
Epoch 3/10
422/422 [==============================] - 7s 17ms/step - loss: 0.2592
- accuracy: 0.9311 - val_loss: 0.2125 - val_accuracy: 0.9438
Epoch 4/10
422/422 [==============================] - 7s 15ms/step - loss: 0.2249
- accuracy: 0.9391 - val_loss: 0.1897 - val_accuracy: 0.9507
Epoch 5/10
422/422 [==============================] - 7s 17ms/step - loss: 0.1984
- accuracy: 0.9467 - val_loss: 0.1779 - val_accuracy: 0.9527
Epoch 6/10
422/422 [==============================] - 6s 15ms/step - loss: 0.1789
- accuracy: 0.9530 - val_loss: 0.1665 - val_accuracy: 0.9560
Epoch 7/10
422/422 [==============================] - 7s 18ms/step - loss: 0.1624
- accuracy: 0.9579 - val_loss: 0.1577 - val_accuracy: 0.9577
Epoch 8/10
422/422 [==============================] - 6s 15ms/step - loss: 0.1488
- accuracy: 0.9619 - val_loss: 0.1525 - val_accuracy: 0.9595
Epoch 9/10
422/422 [==============================] - 8s 19ms/step - loss: 0.1372
- accuracy: 0.9654 - val_loss: 0.1469 - val_accuracy: 0.9608
Epoch 10/10
422/422 [==============================] - 7s 16ms/step - loss: 0.1257
- accuracy: 0.9696 - val_loss: 0.1425 - val_accuracy: 0.9607
```



```
Test loss: 0.168
Test accuracy: 0.952
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_14"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_40 (Dense)            (None, 2048)              1607680

 dense_41 (Dense)            (None, 10)                20490

=================================================================
Total params: 1,628,170
Trainable params: 1,628,170
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 14s 33ms/step - loss:
0.4938 - accuracy: 0.8680 - val_loss: 0.2497 - val_accuracy: 0.9363
Epoch 2/10
422/422 [==============================] - 14s 32ms/step - loss:
0.2576 - accuracy: 0.9312 - val_loss: 0.2001 - val_accuracy: 0.9480
Epoch 3/10
422/422 [==============================] - 14s 33ms/step - loss:
0.2043 - accuracy: 0.9455 - val_loss: 0.1798 - val_accuracy: 0.9508
```

```
Epoch 4/10
422/422 [==============================] - 14s 32ms/step - loss:
0.1713 - accuracy: 0.9552 - val_loss: 0.1632 - val_accuracy: 0.9547
Epoch 5/10
422/422 [==============================] - 14s 32ms/step - loss:
0.1466 - accuracy: 0.9628 - val_loss: 0.1521 - val_accuracy: 0.9595
Epoch 6/10
422/422 [==============================] - 14s 33ms/step - loss:
0.1271 - accuracy: 0.9693 - val_loss: 0.1456 - val_accuracy: 0.9597
Epoch 7/10
422/422 [==============================] - 15s 36ms/step - loss:
0.1110 - accuracy: 0.9752 - val_loss: 0.1383 - val_accuracy: 0.9620
Epoch 8/10
422/422 [==============================] - 14s 32ms/step - loss:
0.0974 - accuracy: 0.9792 - val_loss: 0.1355 - val_accuracy: 0.9632
Epoch 9/10
422/422 [==============================] - 14s 32ms/step - loss:
0.0861 - accuracy: 0.9830 - val_loss: 0.1290 - val_accuracy: 0.9635
Epoch 10/10
422/422 [==============================] - 13s 32ms/step - loss:
0.0763 - accuracy: 0.9861 - val_loss: 0.1252 - val_accuracy: 0.9637
```



```
Test loss: 0.146
Test accuracy: 0.957
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Comentario sobre resultados:

Al aumentar el número de nodos utilizados en una sola capa, se obtiene el mismo resultado que al aumentar el número de epochs del ejemplo anterior: **el accuracy obtenido mejora (aumenta) al igual que el loss obtenido (disminuye).** Esto se puede ver en los resultados:

- Una capa con 32 nodos: **Accuracy de** 0.906
- Una capa con 64 nodos: **Accuracy de** 0.921
- Una capa con 128 nodos: **Accuracy de** 0.934
- Una capa con 256 nodos: **Accuracy de** 0.938
- Una capa con 512 nodos: **Accuracy de** 0.946
- Una capa con 1024 nodos: **Accuracy de** 0.952
- Una capa con 2048 nodos: **Accuracy de** 0.957

4. Más nodos y más capas (4 puntos)

Ahora que hemos visto la cantidad de nodos y la cantidad de capas en un contexto aislado, veamos qué sucede cuando combinamos estos dos factores.

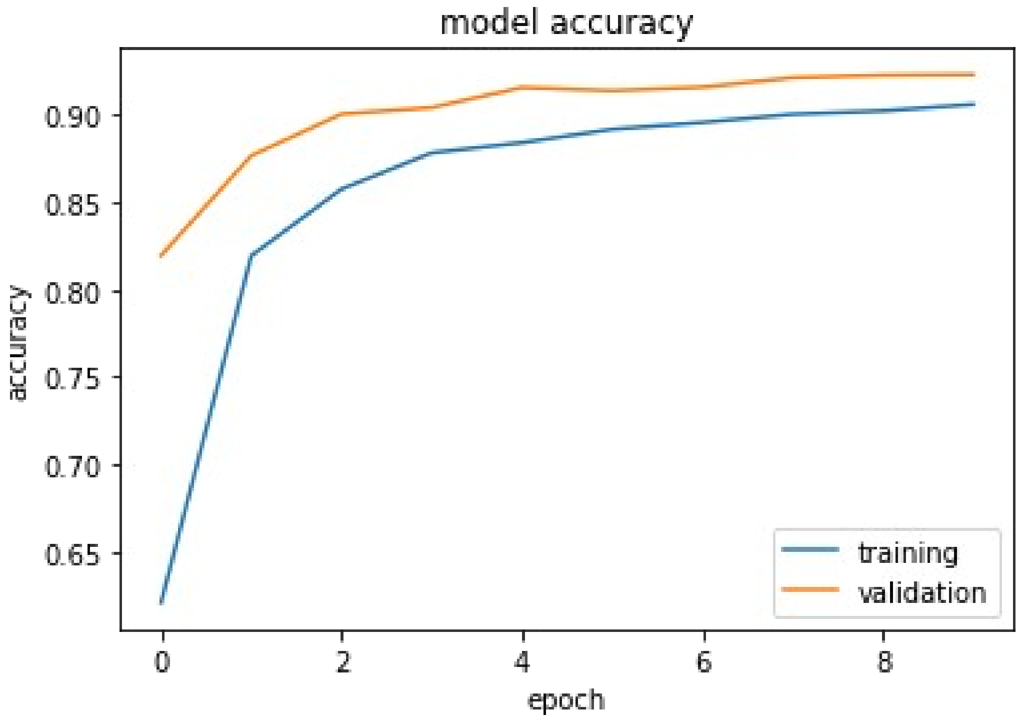
- 4.1 Cree un código que genere modelos con un numero de capas que se incrementan de 1 a 5. Cada capa debe tener 32 nodos. Entrene el modelo con 10 épocas por cada capa, i.e., `epochs=10*layers`. De este modo, el primero modelo tendrá 1 capa de 32 nodos y entrenará durante 10 épocas, el segundo modelo tendrá 2 capas de 32 nodos y entrenará durante 20 épocas y así sucesivamente.
- 4.2 Repita el código anterior pero esta vez cada capa tendrá 128 nodos.
- 4.3 Repita el código anterior pero esta vez cada capa tendrá 512 nodos.

Discuta sus resultados.

```
#PARTE 4.1 ESCRIBA SU CÓDIGO AQUÍ.
for layers in [1, 2, 3, 4, 5]:
    model = create_dense([32] * layers)
    evaluate(model, batch_size=128, epochs=10*layers, verbose=True)
#verbose por defecto es false
```

```
Model: "sequential_15"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_42 (Dense)            (None, 32)                25120

 dense_43 (Dense)            (None, 10)                330

=================================================================
Total params: 25,450
Trainable params: 25,450
Non-trainable params: 0
```

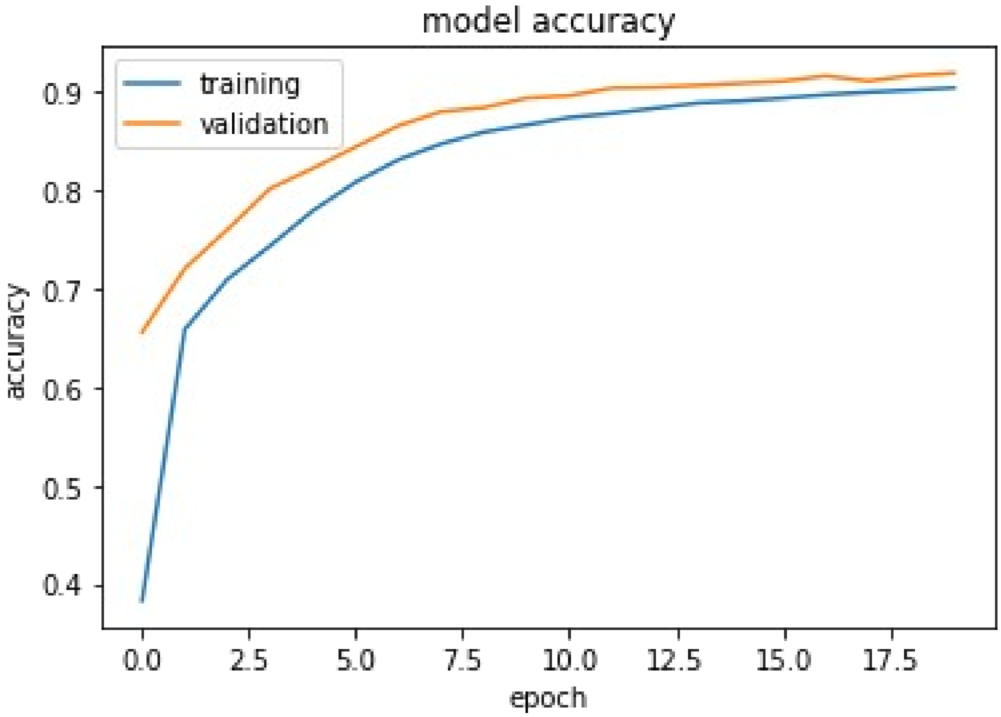
```
Epoch 1/10
422/422 [==============================] - 2s 4ms/step - loss: 1.4622
- accuracy: 0.6209 - val_loss: 0.9809 - val_accuracy: 0.8193
Epoch 2/10
422/422 [==============================] - 1s 3ms/step - loss: 0.8842
- accuracy: 0.8192 - val_loss: 0.7018 - val_accuracy: 0.8765
Epoch 3/10
422/422 [==============================] - 2s 4ms/step - loss: 0.6962
- accuracy: 0.8574 - val_loss: 0.5606 - val_accuracy: 0.9003
Epoch 4/10
422/422 [==============================] - 1s 3ms/step - loss: 0.5843
- accuracy: 0.8781 - val_loss: 0.4929 - val_accuracy: 0.9040
Epoch 5/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5222
- accuracy: 0.8839 - val_loss: 0.4276 - val_accuracy: 0.9155
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4750
- accuracy: 0.8916 - val_loss: 0.4047 - val_accuracy: 0.9137
Epoch 7/10
422/422 [==============================] - 1s 3ms/step - loss: 0.4442
- accuracy: 0.8955 - val_loss: 0.3720 - val_accuracy: 0.9157
Epoch 8/10
422/422 [==============================] - 1s 3ms/step - loss: 0.4200
- accuracy: 0.9002 - val_loss: 0.3498 - val_accuracy: 0.9210
Epoch 9/10
422/422 [==============================] - 1s 3ms/step - loss: 0.3955
- accuracy: 0.9021 - val_loss: 0.3434 - val_accuracy: 0.9225
Epoch 10/10
422/422 [==============================] - 1s 3ms/step - loss: 0.3793
- accuracy: 0.9059 - val_loss: 0.3160 - val_accuracy: 0.9228
```



```
Test loss: 0.362
Test accuracy: 0.909
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_16"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_44 (Dense)            (None, 32)                25120

 dense_45 (Dense)            (None, 32)                1056

 dense_46 (Dense)            (None, 10)                330

=================================================================
Total params: 26,506
Trainable params: 26,506
Non-trainable params: 0
_________________________________________________________________
Epoch 1/20
422/422 [==============================] - 2s 4ms/step - loss: 2.1300
- accuracy: 0.3829 - val_loss: 1.9395 - val_accuracy: 0.6560
Epoch 2/20
422/422 [==============================] - 3s 6ms/step - loss: 1.8308
- accuracy: 0.6588 - val_loss: 1.6892 - val_accuracy: 0.7207
Epoch 3/20
```

```
422/422 [==============================] - 2s 5ms/step - loss: 1.5932
- accuracy: 0.7099 - val_loss: 1.4545 - val_accuracy: 0.7603
Epoch 4/20
422/422 [==============================] - 1s 4ms/step - loss: 1.3806
- accuracy: 0.7438 - val_loss: 1.2502 - val_accuracy: 0.8020
Epoch 5/20
422/422 [==============================] - 1s 3ms/step - loss: 1.2007
- accuracy: 0.7794 - val_loss: 1.0807 - val_accuracy: 0.8225
Epoch 6/20
422/422 [==============================] - 1s 3ms/step - loss: 1.0521
- accuracy: 0.8085 - val_loss: 0.9430 - val_accuracy: 0.8443
Epoch 7/20
422/422 [==============================] - 1s 3ms/step - loss: 0.9312
- accuracy: 0.8312 - val_loss: 0.8344 - val_accuracy: 0.8658
Epoch 8/20
422/422 [==============================] - 1s 3ms/step - loss: 0.8344
- accuracy: 0.8474 - val_loss: 0.7437 - val_accuracy: 0.8802
Epoch 9/20
422/422 [==============================] - 1s 3ms/step - loss: 0.7551
- accuracy: 0.8594 - val_loss: 0.6745 - val_accuracy: 0.8843
Epoch 10/20
422/422 [==============================] - 2s 5ms/step - loss: 0.6904
- accuracy: 0.8667 - val_loss: 0.6088 - val_accuracy: 0.8940
Epoch 11/20
422/422 [==============================] - 2s 5ms/step - loss: 0.6363
- accuracy: 0.8741 - val_loss: 0.5618 - val_accuracy: 0.8965
Epoch 12/20
422/422 [==============================] - 1s 3ms/step - loss: 0.5918
- accuracy: 0.8784 - val_loss: 0.5162 - val_accuracy: 0.9042
Epoch 13/20
422/422 [==============================] - 1s 3ms/step - loss: 0.5551
- accuracy: 0.8836 - val_loss: 0.4913 - val_accuracy: 0.9048
Epoch 14/20
422/422 [==============================] - 1s 3ms/step - loss: 0.5230
- accuracy: 0.8890 - val_loss: 0.4582 - val_accuracy: 0.9068
Epoch 15/20
422/422 [==============================] - 1s 3ms/step - loss: 0.4951
- accuracy: 0.8910 - val_loss: 0.4374 - val_accuracy: 0.9090
Epoch 16/20
422/422 [==============================] - 1s 3ms/step - loss: 0.4729
- accuracy: 0.8940 - val_loss: 0.4159 - val_accuracy: 0.9110
Epoch 17/20
422/422 [==============================] - 1s 4ms/step - loss: 0.4561
- accuracy: 0.8974 - val_loss: 0.3923 - val_accuracy: 0.9163
Epoch 18/20
422/422 [==============================] - 2s 5ms/step - loss: 0.4366
- accuracy: 0.9000 - val_loss: 0.3815 - val_accuracy: 0.9112
Epoch 19/20
422/422 [==============================] - 2s 5ms/step - loss: 0.4202
- accuracy: 0.9021 - val_loss: 0.3671 - val_accuracy: 0.9167
```

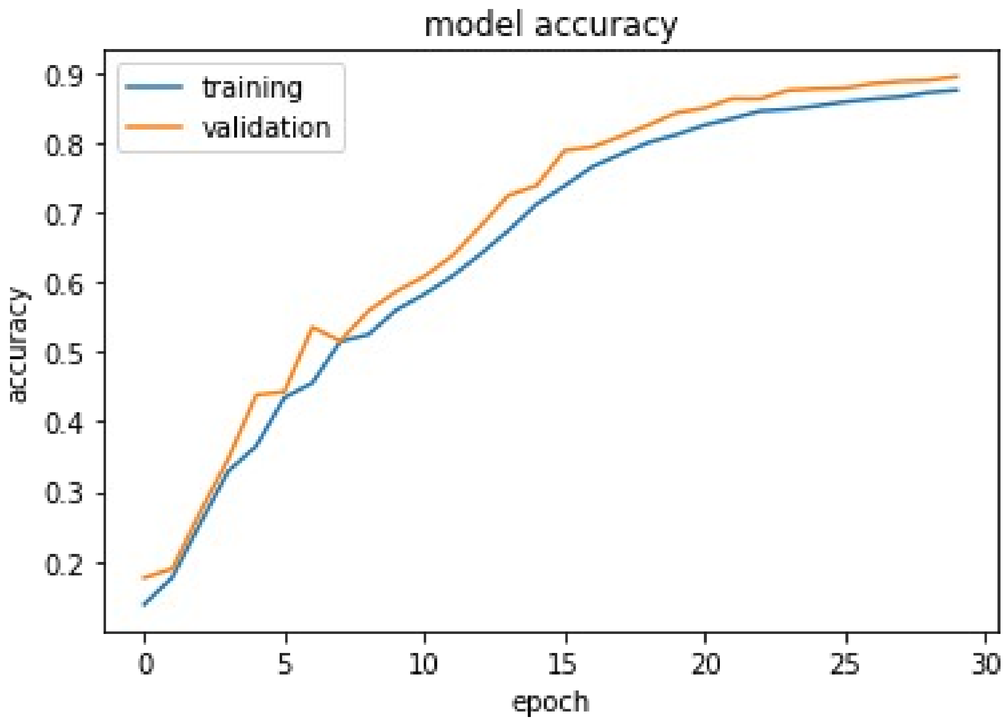
```
Epoch 20/20
422/422 [==============================] - 1s 4ms/step - loss: 0.4059
- accuracy: 0.9043 - val_loss: 0.3535 - val_accuracy: 0.9193
```



```
Test loss: 0.392
Test accuracy: 0.907
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_17"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_47 (Dense)            (None, 32)                25120

 dense_48 (Dense)            (None, 32)                1056

 dense_49 (Dense)            (None, 32)                1056

 dense_50 (Dense)            (None, 10)                330

=================================================================
Total params: 27,562
Trainable params: 27,562
Non-trainable params: 0
_________________________________________________________________
Epoch 1/30
```

```
422/422 [==============================] - 2s 4ms/step - loss: 2.3301
- accuracy: 0.1380 - val_loss: 2.2827 - val_accuracy: 0.1763
Epoch 2/30
422/422 [==============================] - 2s 4ms/step - loss: 2.2732
- accuracy: 0.1770 - val_loss: 2.2620 - val_accuracy: 0.1893
Epoch 3/30
422/422 [==============================] - 2s 4ms/step - loss: 2.2519
- accuracy: 0.2546 - val_loss: 2.2387 - val_accuracy: 0.2712
Epoch 4/30
422/422 [==============================] - 1s 3ms/step - loss: 2.2264
- accuracy: 0.3294 - val_loss: 2.2097 - val_accuracy: 0.3470
Epoch 5/30
422/422 [==============================] - 2s 5ms/step - loss: 2.1941
- accuracy: 0.3651 - val_loss: 2.1718 - val_accuracy: 0.4385
Epoch 6/30
422/422 [==============================] - 2s 4ms/step - loss: 2.1517
- accuracy: 0.4345 - val_loss: 2.1222 - val_accuracy: 0.4428
Epoch 7/30
422/422 [==============================] - 2s 4ms/step - loss: 2.0954
- accuracy: 0.4554 - val_loss: 2.0561 - val_accuracy: 0.5353
Epoch 8/30
422/422 [==============================] - 2s 4ms/step - loss: 2.0225
- accuracy: 0.5151 - val_loss: 1.9732 - val_accuracy: 0.5157
Epoch 9/30
422/422 [==============================] - 2s 4ms/step - loss: 1.9320
- accuracy: 0.5247 - val_loss: 1.8733 - val_accuracy: 0.5587
Epoch 10/30
422/422 [==============================] - 1s 3ms/step - loss: 1.8259
- accuracy: 0.5601 - val_loss: 1.7579 - val_accuracy: 0.5868
Epoch 11/30
422/422 [==============================] - 1s 3ms/step - loss: 1.7085
- accuracy: 0.5827 - val_loss: 1.6360 - val_accuracy: 0.6083
Epoch 12/30
422/422 [==============================] - 1s 3ms/step - loss: 1.5861
- accuracy: 0.6086 - val_loss: 1.5104 - val_accuracy: 0.6375
Epoch 13/30
422/422 [==============================] - 2s 6ms/step - loss: 1.4672
- accuracy: 0.6397 - val_loss: 1.3942 - val_accuracy: 0.6797
Epoch 14/30
422/422 [==============================] - 2s 4ms/step - loss: 1.3576
- accuracy: 0.6738 - val_loss: 1.2900 - val_accuracy: 0.7243
Epoch 15/30
422/422 [==============================] - 2s 4ms/step - loss: 1.2599
- accuracy: 0.7115 - val_loss: 1.1958 - val_accuracy: 0.7382
Epoch 16/30
422/422 [==============================] - 1s 3ms/step - loss: 1.1719
- accuracy: 0.7378 - val_loss: 1.1049 - val_accuracy: 0.7883
Epoch 17/30
422/422 [==============================] - 2s 4ms/step - loss: 1.0940
- accuracy: 0.7655 - val_loss: 1.0306 - val_accuracy: 0.7937
```

```
Epoch 18/30
422/422 [==============================] - 1s 3ms/step - loss: 1.0227
- accuracy: 0.7831 - val_loss: 0.9603 - val_accuracy: 0.8087
Epoch 19/30
422/422 [==============================] - 1s 3ms/step - loss: 0.9619
- accuracy: 0.8001 - val_loss: 0.9077 - val_accuracy: 0.8252
Epoch 20/30
422/422 [==============================] - 1s 3ms/step - loss: 0.9054
- accuracy: 0.8113 - val_loss: 0.8458 - val_accuracy: 0.8425
Epoch 21/30
422/422 [==============================] - 2s 5ms/step - loss: 0.8534
- accuracy: 0.8250 - val_loss: 0.7981 - val_accuracy: 0.8493
Epoch 22/30
422/422 [==============================] - 2s 4ms/step - loss: 0.8062
- accuracy: 0.8345 - val_loss: 0.7501 - val_accuracy: 0.8632
Epoch 23/30
422/422 [==============================] - 1s 4ms/step - loss: 0.7620
- accuracy: 0.8452 - val_loss: 0.7077 - val_accuracy: 0.8628
Epoch 24/30
422/422 [==============================] - 1s 4ms/step - loss: 0.7266
- accuracy: 0.8475 - val_loss: 0.6761 - val_accuracy: 0.8750
Epoch 25/30
422/422 [==============================] - 1s 3ms/step - loss: 0.6941
- accuracy: 0.8524 - val_loss: 0.6410 - val_accuracy: 0.8770
Epoch 26/30
422/422 [==============================] - 1s 3ms/step - loss: 0.6598
- accuracy: 0.8582 - val_loss: 0.6100 - val_accuracy: 0.8782
Epoch 27/30
422/422 [==============================] - 1s 3ms/step - loss: 0.6327
- accuracy: 0.8624 - val_loss: 0.5863 - val_accuracy: 0.8842
Epoch 28/30
422/422 [==============================] - 2s 4ms/step - loss: 0.6093
- accuracy: 0.8655 - val_loss: 0.5570 - val_accuracy: 0.8875
Epoch 29/30
422/422 [==============================] - 2s 6ms/step - loss: 0.5826
- accuracy: 0.8718 - val_loss: 0.5347 - val_accuracy: 0.8890
Epoch 30/30
422/422 [==============================] - 2s 4ms/step - loss: 0.5617
- accuracy: 0.8751 - val_loss: 0.5138 - val_accuracy: 0.8943
```

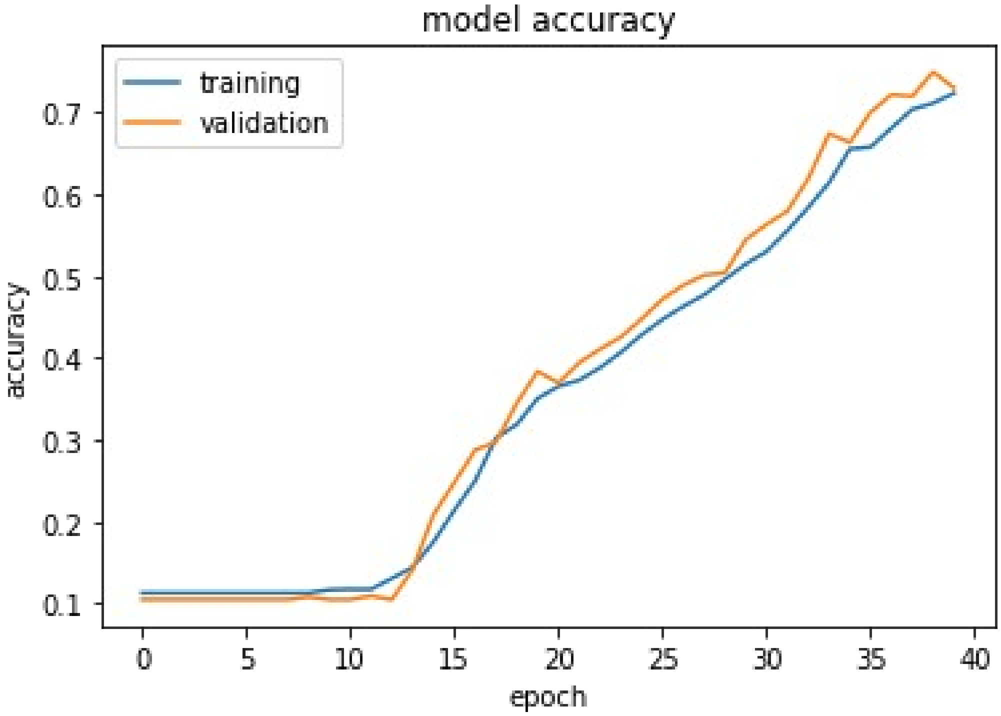


```
Test loss: 0.548
Test accuracy: 0.88
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_18"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_51 (Dense)            (None, 32)                25120

 dense_52 (Dense)            (None, 32)                1056

 dense_53 (Dense)            (None, 32)                1056

 dense_54 (Dense)            (None, 32)                1056

 dense_55 (Dense)            (None, 10)                330

=================================================================
Total params: 28,618
Trainable params: 28,618
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 2s 4ms/step - loss: 2.3349
- accuracy: 0.1132 - val_loss: 2.2993 - val_accuracy: 0.1050
```

```
Epoch 2/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2975
- accuracy: 0.1132 - val_loss: 2.2971 - val_accuracy: 0.1050
Epoch 3/40
422/422 [==============================] - 1s 3ms/step - loss: 2.2959
- accuracy: 0.1132 - val_loss: 2.2958 - val_accuracy: 0.1050
Epoch 4/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2943
- accuracy: 0.1132 - val_loss: 2.2937 - val_accuracy: 0.1050
Epoch 5/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2927
- accuracy: 0.1132 - val_loss: 2.2921 - val_accuracy: 0.1050
Epoch 6/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2910
- accuracy: 0.1132 - val_loss: 2.2904 - val_accuracy: 0.1050
Epoch 7/40
422/422 [==============================] - 2s 5ms/step - loss: 2.2891
- accuracy: 0.1132 - val_loss: 2.2882 - val_accuracy: 0.1050
Epoch 8/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2871
- accuracy: 0.1132 - val_loss: 2.2860 - val_accuracy: 0.1050
Epoch 9/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2847
- accuracy: 0.1132 - val_loss: 2.2831 - val_accuracy: 0.1083
Epoch 10/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2819
- accuracy: 0.1171 - val_loss: 2.2803 - val_accuracy: 0.1050
Epoch 11/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2786
- accuracy: 0.1177 - val_loss: 2.2770 - val_accuracy: 0.1050
Epoch 12/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2745
- accuracy: 0.1176 - val_loss: 2.2720 - val_accuracy: 0.1095
Epoch 13/40
422/422 [==============================] - 2s 5ms/step - loss: 2.2697
- accuracy: 0.1309 - val_loss: 2.2667 - val_accuracy: 0.1050
Epoch 14/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2636
- accuracy: 0.1446 - val_loss: 2.2602 - val_accuracy: 0.1418
Epoch 15/40
422/422 [==============================] - 2s 5ms/step - loss: 2.2560
- accuracy: 0.1754 - val_loss: 2.2514 - val_accuracy: 0.2087
Epoch 16/40
422/422 [==============================] - 1s 3ms/step - loss: 2.2460
- accuracy: 0.2141 - val_loss: 2.2396 - val_accuracy: 0.2478
Epoch 17/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2328
- accuracy: 0.2500 - val_loss: 2.2241 - val_accuracy: 0.2875
Epoch 18/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2153
```

```
- accuracy: 0.3022 - val_loss: 2.2042 - val_accuracy: 0.2973
Epoch 19/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1916
- accuracy: 0.3189 - val_loss: 2.1758 - val_accuracy: 0.3450
Epoch 20/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1583
- accuracy: 0.3507 - val_loss: 2.1356 - val_accuracy: 0.3837
Epoch 21/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1113
- accuracy: 0.3654 - val_loss: 2.0794 - val_accuracy: 0.3695
Epoch 22/40
422/422 [==============================] - 3s 6ms/step - loss: 2.0465
- accuracy: 0.3730 - val_loss: 2.0039 - val_accuracy: 0.3945
Epoch 23/40
422/422 [==============================] - 1s 3ms/step - loss: 1.9621
- accuracy: 0.3881 - val_loss: 1.9081 - val_accuracy: 0.4108
Epoch 24/40
422/422 [==============================] - 1s 4ms/step - loss: 1.8616
- accuracy: 0.4070 - val_loss: 1.7994 - val_accuracy: 0.4257
Epoch 25/40
422/422 [==============================] - 2s 4ms/step - loss: 1.7534
- accuracy: 0.4283 - val_loss: 1.6889 - val_accuracy: 0.4478
Epoch 26/40
422/422 [==============================] - 2s 4ms/step - loss: 1.6467
- accuracy: 0.4472 - val_loss: 1.5811 - val_accuracy: 0.4720
Epoch 27/40
422/422 [==============================] - 2s 4ms/step - loss: 1.5487
- accuracy: 0.4631 - val_loss: 1.4868 - val_accuracy: 0.4890
Epoch 28/40
422/422 [==============================] - 2s 4ms/step - loss: 1.4632
- accuracy: 0.4771 - val_loss: 1.4067 - val_accuracy: 0.5013
Epoch 29/40
422/422 [==============================] - 2s 5ms/step - loss: 1.3929
- accuracy: 0.4962 - val_loss: 1.3440 - val_accuracy: 0.5040
Epoch 30/40
422/422 [==============================] - 2s 5ms/step - loss: 1.3338
- accuracy: 0.5148 - val_loss: 1.2829 - val_accuracy: 0.5443
Epoch 31/40
422/422 [==============================] - 1s 4ms/step - loss: 1.2844
- accuracy: 0.5299 - val_loss: 1.2387 - val_accuracy: 0.5630
Epoch 32/40
422/422 [==============================] - 2s 4ms/step - loss: 1.2418
- accuracy: 0.5555 - val_loss: 1.1967 - val_accuracy: 0.5793
Epoch 33/40
422/422 [==============================] - 2s 4ms/step - loss: 1.2040
- accuracy: 0.5836 - val_loss: 1.1609 - val_accuracy: 0.6188
Epoch 34/40
422/422 [==============================] - 1s 4ms/step - loss: 1.1682
- accuracy: 0.6139 - val_loss: 1.1272 - val_accuracy: 0.6738
Epoch 35/40
```

```
422/422 [==============================] - 2s 4ms/step - loss: 1.1336
- accuracy: 0.6550 - val_loss: 1.0928 - val_accuracy: 0.6628
Epoch 36/40
422/422 [==============================] - 2s 4ms/step - loss: 1.1021
- accuracy: 0.6575 - val_loss: 1.0677 - val_accuracy: 0.7002
Epoch 37/40
422/422 [==============================] - 2s 5ms/step - loss: 1.0711
- accuracy: 0.6808 - val_loss: 1.0336 - val_accuracy: 0.7215
Epoch 38/40
422/422 [==============================] - 2s 5ms/step - loss: 1.0431
- accuracy: 0.7036 - val_loss: 1.0079 - val_accuracy: 0.7197
Epoch 39/40
422/422 [==============================] - 2s 4ms/step - loss: 1.0178
- accuracy: 0.7111 - val_loss: 0.9729 - val_accuracy: 0.7490
Epoch 40/40
422/422 [==============================] - 2s 4ms/step - loss: 0.9914
- accuracy: 0.7234 - val_loss: 0.9554 - val_accuracy: 0.7300
```

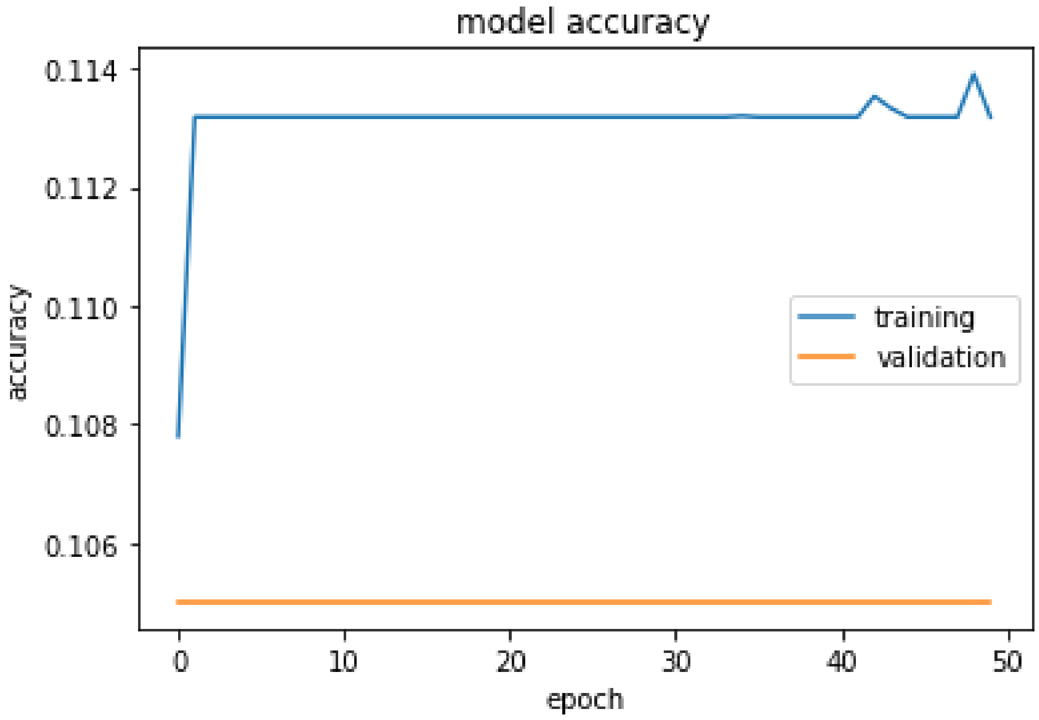


```
Test loss: 0.984
Test accuracy: 0.725
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.    ]
Model: "sequential_19"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_56 (Dense)            (None, 32)                25120

 dense_57 (Dense)            (None, 32)                1056

 dense_58 (Dense)            (None, 32)                1056

 dense_59 (Dense)            (None, 32)                1056

 dense_60 (Dense)            (None, 32)                1056

 dense_61 (Dense)            (None, 10)                330

=================================================================
Total params: 29,674
Trainable params: 29,674
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
422/422 [==============================] - 3s 5ms/step - loss: 2.3275
- accuracy: 0.1078 - val_loss: 2.3021 - val_accuracy: 0.1050
Epoch 2/50
422/422 [==============================] - 2s 5ms/step - loss: 2.3016
- accuracy: 0.1132 - val_loss: 2.3022 - val_accuracy: 0.1050
Epoch 3/50
422/422 [==============================] - 3s 6ms/step - loss: 2.3015
- accuracy: 0.1132 - val_loss: 2.3020 - val_accuracy: 0.1050
Epoch 4/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3014
- accuracy: 0.1132 - val_loss: 2.3018 - val_accuracy: 0.1050
Epoch 5/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3013
- accuracy: 0.1132 - val_loss: 2.3021 - val_accuracy: 0.1050
Epoch 6/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3012
- accuracy: 0.1132 - val_loss: 2.3016 - val_accuracy: 0.1050
Epoch 7/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3011
- accuracy: 0.1132 - val_loss: 2.3015 - val_accuracy: 0.1050
Epoch 8/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3010
- accuracy: 0.1132 - val_loss: 2.3016 - val_accuracy: 0.1050
Epoch 9/50
422/422 [==============================] - 2s 6ms/step - loss: 2.3009
- accuracy: 0.1132 - val_loss: 2.3016 - val_accuracy: 0.1050
Epoch 10/50
422/422 [==============================] - 2s 5ms/step - loss: 2.3008
- accuracy: 0.1132 - val_loss: 2.3012 - val_accuracy: 0.1050
Epoch 11/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3007
- accuracy: 0.1132 - val_loss: 2.3011 - val_accuracy: 0.1050
```

```
Epoch 12/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3006
- accuracy: 0.1132 - val_loss: 2.3014 - val_accuracy: 0.1050
Epoch 13/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3005
- accuracy: 0.1132 - val_loss: 2.3018 - val_accuracy: 0.1050
Epoch 14/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3004
- accuracy: 0.1132 - val_loss: 2.3009 - val_accuracy: 0.1050
Epoch 15/50
422/422 [==============================] - 2s 4ms/step - loss: 2.3003
- accuracy: 0.1132 - val_loss: 2.3011 - val_accuracy: 0.1050
Epoch 16/50
422/422 [==============================] - 3s 6ms/step - loss: 2.3001
- accuracy: 0.1132 - val_loss: 2.3008 - val_accuracy: 0.1050
Epoch 17/50
422/422 [==============================] - 2s 5ms/step - loss: 2.3001
- accuracy: 0.1132 - val_loss: 2.3005 - val_accuracy: 0.1050
Epoch 18/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2999
- accuracy: 0.1132 - val_loss: 2.3007 - val_accuracy: 0.1050
Epoch 19/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2998
- accuracy: 0.1132 - val_loss: 2.3002 - val_accuracy: 0.1050
Epoch 20/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2996
- accuracy: 0.1132 - val_loss: 2.3006 - val_accuracy: 0.1050
Epoch 21/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2995
- accuracy: 0.1132 - val_loss: 2.3000 - val_accuracy: 0.1050
Epoch 22/50
422/422 [==============================] - 2s 5ms/step - loss: 2.2994
- accuracy: 0.1132 - val_loss: 2.2999 - val_accuracy: 0.1050
Epoch 23/50
422/422 [==============================] - 3s 6ms/step - loss: 2.2992
- accuracy: 0.1132 - val_loss: 2.3002 - val_accuracy: 0.1050
Epoch 24/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2991
- accuracy: 0.1132 - val_loss: 2.2997 - val_accuracy: 0.1050
Epoch 25/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2989
- accuracy: 0.1132 - val_loss: 2.2993 - val_accuracy: 0.1050
Epoch 26/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2987
- accuracy: 0.1132 - val_loss: 2.2994 - val_accuracy: 0.1050
Epoch 27/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2986
- accuracy: 0.1132 - val_loss: 2.2990 - val_accuracy: 0.1050
Epoch 28/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2985
```

```
- accuracy: 0.1132 - val_loss: 2.2988 - val_accuracy: 0.1050
Epoch 29/50
422/422 [==============================] - 2s 5ms/step - loss: 2.2982
- accuracy: 0.1132 - val_loss: 2.2988 - val_accuracy: 0.1050
Epoch 30/50
422/422 [==============================] - 2s 6ms/step - loss: 2.2981
- accuracy: 0.1132 - val_loss: 2.2988 - val_accuracy: 0.1050
Epoch 31/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2979
- accuracy: 0.1132 - val_loss: 2.2984 - val_accuracy: 0.1050
Epoch 32/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2977
- accuracy: 0.1132 - val_loss: 2.2981 - val_accuracy: 0.1050
Epoch 33/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2975
- accuracy: 0.1132 - val_loss: 2.2982 - val_accuracy: 0.1050
Epoch 34/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2973
- accuracy: 0.1132 - val_loss: 2.2975 - val_accuracy: 0.1050
Epoch 35/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2970
- accuracy: 0.1132 - val_loss: 2.2975 - val_accuracy: 0.1050
Epoch 36/50
422/422 [==============================] - 2s 5ms/step - loss: 2.2968
- accuracy: 0.1132 - val_loss: 2.2968 - val_accuracy: 0.1050
Epoch 37/50
422/422 [==============================] - 2s 5ms/step - loss: 2.2965
- accuracy: 0.1132 - val_loss: 2.2972 - val_accuracy: 0.1050
Epoch 38/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2963
- accuracy: 0.1132 - val_loss: 2.2967 - val_accuracy: 0.1050
Epoch 39/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2960
- accuracy: 0.1132 - val_loss: 2.2962 - val_accuracy: 0.1050
Epoch 40/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2956
- accuracy: 0.1132 - val_loss: 2.2962 - val_accuracy: 0.1050
Epoch 41/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2952
- accuracy: 0.1132 - val_loss: 2.2958 - val_accuracy: 0.1050
Epoch 42/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2949
- accuracy: 0.1132 - val_loss: 2.2954 - val_accuracy: 0.1050
Epoch 43/50
422/422 [==============================] - 3s 6ms/step - loss: 2.2943
- accuracy: 0.1135 - val_loss: 2.2950 - val_accuracy: 0.1050
Epoch 44/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2940
- accuracy: 0.1133 - val_loss: 2.2946 - val_accuracy: 0.1050
Epoch 45/50
```

```
422/422 [==============================] - 2s 4ms/step - loss: 2.2935
- accuracy: 0.1132 - val_loss: 2.2938 - val_accuracy: 0.1050
Epoch 46/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2929
- accuracy: 0.1132 - val_loss: 2.2932 - val_accuracy: 0.1050
Epoch 47/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2923
- accuracy: 0.1132 - val_loss: 2.2925 - val_accuracy: 0.1050
Epoch 48/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2916
- accuracy: 0.1132 - val_loss: 2.2915 - val_accuracy: 0.1050
Epoch 49/50
422/422 [==============================] - 2s 4ms/step - loss: 2.2908
- accuracy: 0.1139 - val_loss: 2.2908 - val_accuracy: 0.1050
Epoch 50/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2900
- accuracy: 0.1132 - val_loss: 2.2901 - val_accuracy: 0.1050
```



```
Test loss: 2.29
Test accuracy: 0.113
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Discusión

Test loss: 0.362 Test accuracy: 0.909

Test loss: 0.392 Test accuracy: 0.907

Test loss: 0.548 Test accuracy: 0.88

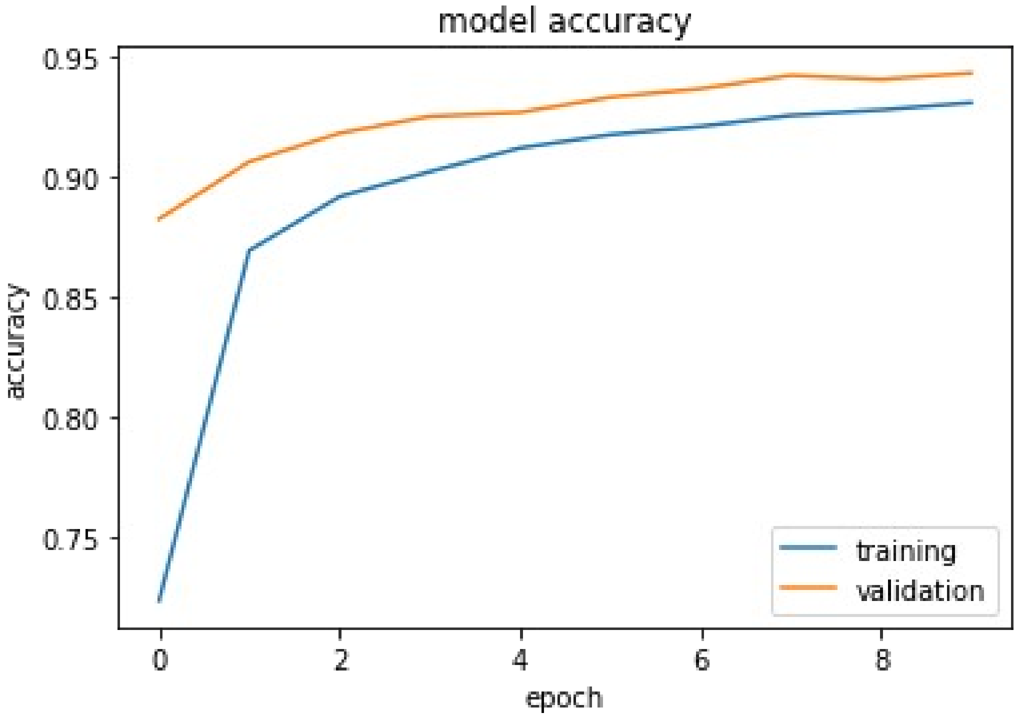
Test loss: 0.984 Test accuracy: 0.725

Test loss: 2.29 Test accuracy: 0.113

```
#PARTE 4.2 ESCRIBA SU CÓDIGO AQUÍ.
for layers in [1, 2, 3, 4, 5]:
    model = create_dense([128] * layers)
    evaluate(model, batch_size=128, epochs=10*layers, verbose=True)
#verbose por defecto es false
```

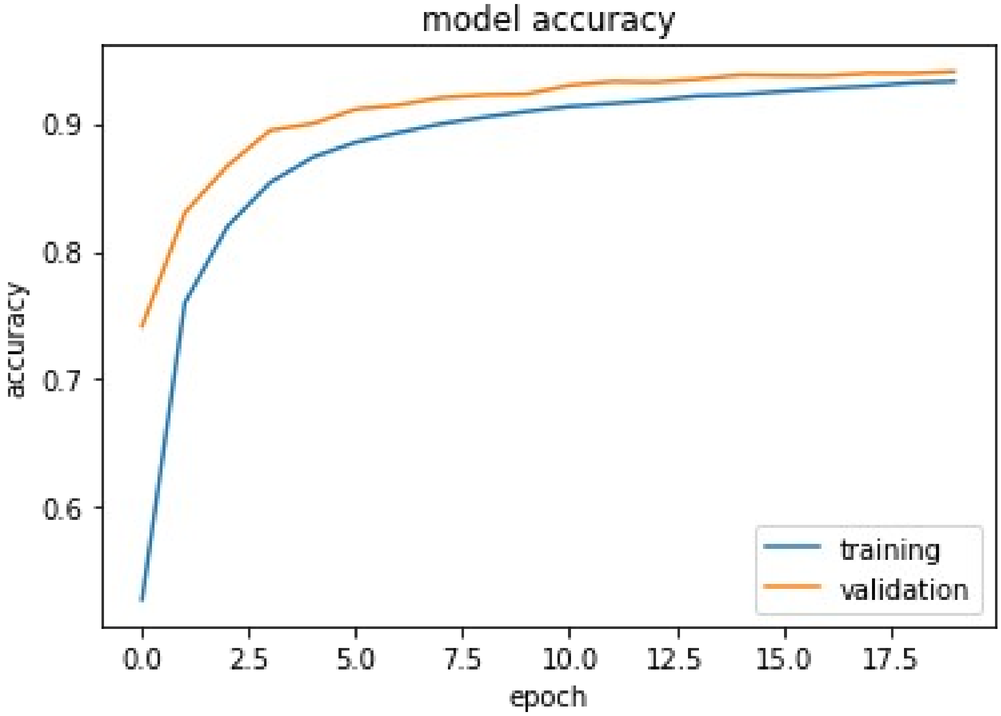
```
Model: "sequential_20"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_62 (Dense)            (None, 128)               100480    
                                                                 
 dense_63 (Dense)            (None, 10)                1290      
                                                                 
=================================================================
Total params: 101,770
Trainable params: 101,770
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 5ms/step - loss: 1.0763
- accuracy: 0.7236 - val_loss: 0.5830 - val_accuracy: 0.8823
Epoch 2/10
422/422 [==============================] - 2s 5ms/step - loss: 0.5508
- accuracy: 0.8692 - val_loss: 0.4174 - val_accuracy: 0.9062
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4335
- accuracy: 0.8916 - val_loss: 0.3478 - val_accuracy: 0.9180
Epoch 4/10
422/422 [==============================] - 3s 7ms/step - loss: 0.3751
- accuracy: 0.9020 - val_loss: 0.3048 - val_accuracy: 0.9250
Epoch 5/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3373
- accuracy: 0.9119 - val_loss: 0.2798 - val_accuracy: 0.9267
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3132
- accuracy: 0.9174 - val_loss: 0.2630 - val_accuracy: 0.9330
Epoch 7/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2934
```

```
- accuracy: 0.9208 - val_loss: 0.2475 - val_accuracy: 0.9365
Epoch 8/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2759
- accuracy: 0.9254 - val_loss: 0.2339 - val_accuracy: 0.9420
Epoch 9/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2636
- accuracy: 0.9277 - val_loss: 0.2291 - val_accuracy: 0.9403
Epoch 10/10
422/422 [==============================] - 3s 7ms/step - loss: 0.2535
- accuracy: 0.9307 - val_loss: 0.2182 - val_accuracy: 0.9430
```



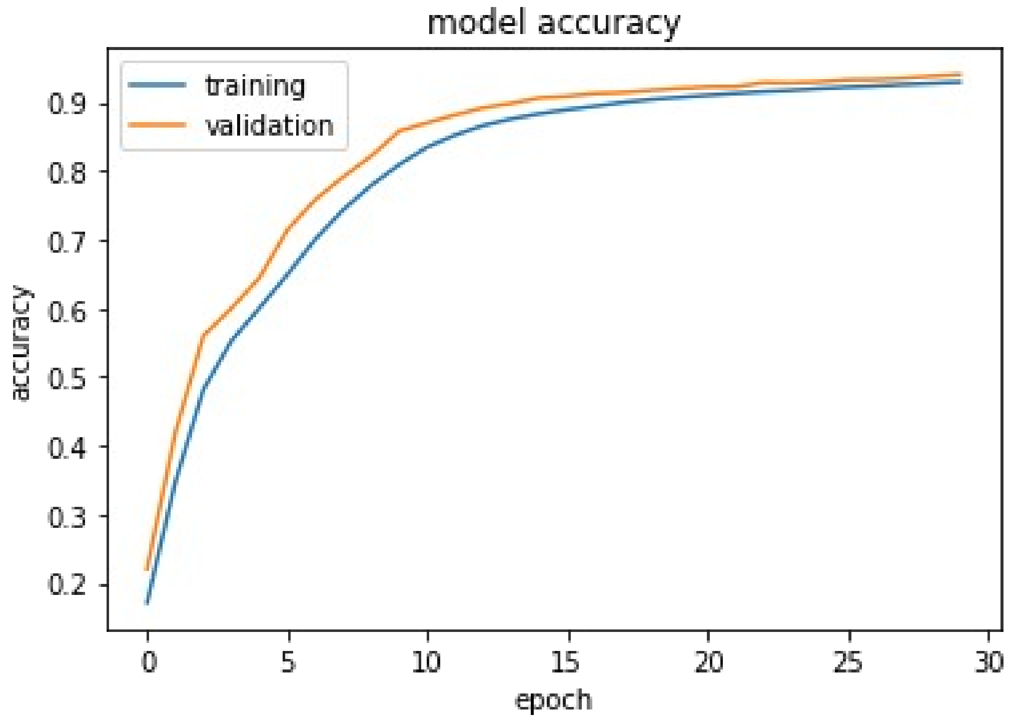
```
Test loss: 0.246
Test accuracy: 0.932
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_21"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_64 (Dense)            (None, 128)               100480

 dense_65 (Dense)            (None, 128)               16512

 dense_66 (Dense)            (None, 10)                1290

=================================================================
Total params: 118,282
Trainable params: 118,282
Non-trainable params: 0
_________________________________________________________________
Epoch 1/20
422/422 [==============================] - 3s 7ms/step - loss: 1.9777
- accuracy: 0.5280 - val_loss: 1.6441 - val_accuracy: 0.7415
Epoch 2/20
422/422 [==============================] - 2s 6ms/step - loss: 1.4170
- accuracy: 0.7598 - val_loss: 1.1498 - val_accuracy: 0.8300
Epoch 3/20
422/422 [==============================] - 2s 6ms/step - loss: 1.0249
- accuracy: 0.8195 - val_loss: 0.8318 - val_accuracy: 0.8668
Epoch 4/20
422/422 [==============================] - 3s 8ms/step - loss: 0.7821
- accuracy: 0.8538 - val_loss: 0.6424 - val_accuracy: 0.8945
Epoch 5/20
422/422 [==============================] - 3s 6ms/step - loss: 0.6343
- accuracy: 0.8735 - val_loss: 0.5285 - val_accuracy: 0.8998
Epoch 6/20
422/422 [==============================] - 2s 6ms/step - loss: 0.5402
- accuracy: 0.8852 - val_loss: 0.4523 - val_accuracy: 0.9110
Epoch 7/20
422/422 [==============================] - 2s 6ms/step - loss: 0.4765
- accuracy: 0.8926 - val_loss: 0.3998 - val_accuracy: 0.9143
Epoch 8/20
422/422 [==============================] - 3s 6ms/step - loss: 0.4300
- accuracy: 0.8996 - val_loss: 0.3611 - val_accuracy: 0.9200
Epoch 9/20
422/422 [==============================] - 3s 7ms/step - loss: 0.3955
- accuracy: 0.9048 - val_loss: 0.3339 - val_accuracy: 0.9222
Epoch 10/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3690
- accuracy: 0.9092 - val_loss: 0.3159 - val_accuracy: 0.9228
Epoch 11/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3478
- accuracy: 0.9131 - val_loss: 0.2959 - val_accuracy: 0.9297
Epoch 12/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3308
- accuracy: 0.9155 - val_loss: 0.2854 - val_accuracy: 0.9327
Epoch 13/20
422/422 [==============================] - 3s 8ms/step - loss: 0.3171
- accuracy: 0.9179 - val_loss: 0.2704 - val_accuracy: 0.9322
Epoch 14/20
422/422 [==============================] - 4s 9ms/step - loss: 0.3021
- accuracy: 0.9214 - val_loss: 0.2580 - val_accuracy: 0.9343
Epoch 15/20
422/422 [==============================] - 2s 6ms/step - loss: 0.2907
- accuracy: 0.9225 - val_loss: 0.2483 - val_accuracy: 0.9377
Epoch 16/20
```

```
422/422 [==============================] - 2s 6ms/step - loss: 0.2805
- accuracy: 0.9249 - val_loss: 0.2441 - val_accuracy: 0.9372
Epoch 17/20
422/422 [==============================] - 3s 6ms/step - loss: 0.2712
- accuracy: 0.9273 - val_loss: 0.2375 - val_accuracy: 0.9370
Epoch 18/20
422/422 [==============================] - 3s 8ms/step - loss: 0.2635
- accuracy: 0.9290 - val_loss: 0.2302 - val_accuracy: 0.9390
Epoch 19/20
422/422 [==============================] - 2s 6ms/step - loss: 0.2550
- accuracy: 0.9316 - val_loss: 0.2280 - val_accuracy: 0.9388
Epoch 20/20
422/422 [==============================] - 2s 5ms/step - loss: 0.2478
- accuracy: 0.9328 - val_loss: 0.2211 - val_accuracy: 0.9403
```



```
Test loss: 0.252
Test accuracy: 0.93
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_22"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_67 (Dense)            (None, 128)               100480

 dense_68 (Dense)            (None, 128)               16512
 dense_69 (Dense)            (None, 128)               16512

 dense_70 (Dense)            (None, 10)                1290

=================================================================
Total params: 134,794
Trainable params: 134,794
Non-trainable params: 0
_________________________________________________________________
Epoch 1/30
422/422 [==============================] - 4s 7ms/step - loss: 2.2797
- accuracy: 0.1716 - val_loss: 2.2490 - val_accuracy: 0.2208
Epoch 2/30
422/422 [==============================] - 4s 8ms/step - loss: 2.2181
- accuracy: 0.3469 - val_loss: 2.1788 - val_accuracy: 0.4187
Epoch 3/30
422/422 [==============================] - 3s 6ms/step - loss: 2.1345
- accuracy: 0.4814 - val_loss: 2.0752 - val_accuracy: 0.5597
Epoch 4/30
422/422 [==============================] - 3s 6ms/step - loss: 2.0056
- accuracy: 0.5523 - val_loss: 1.9135 - val_accuracy: 0.5998
Epoch 5/30
422/422 [==============================] - 3s 6ms/step - loss: 1.8217
- accuracy: 0.6001 - val_loss: 1.7031 - val_accuracy: 0.6442
Epoch 6/30
422/422 [==============================] - 4s 9ms/step - loss: 1.6030
- accuracy: 0.6490 - val_loss: 1.4707 - val_accuracy: 0.7138
Epoch 7/30
422/422 [==============================] - 3s 7ms/step - loss: 1.3866
- accuracy: 0.7004 - val_loss: 1.2581 - val_accuracy: 0.7580
Epoch 8/30
422/422 [==============================] - 3s 6ms/step - loss: 1.1985
- accuracy: 0.7434 - val_loss: 1.0813 - val_accuracy: 0.7913
Epoch 9/30
422/422 [==============================] - 3s 6ms/step - loss: 1.0435
- accuracy: 0.7790 - val_loss: 0.9357 - val_accuracy: 0.8217
Epoch 10/30
422/422 [==============================] - 3s 8ms/step - loss: 0.9156
- accuracy: 0.8091 - val_loss: 0.8177 - val_accuracy: 0.8580
Epoch 11/30
422/422 [==============================] - 3s 7ms/step - loss: 0.8105
- accuracy: 0.8346 - val_loss: 0.7195 - val_accuracy: 0.8700
Epoch 12/30
422/422 [==============================] - 3s 7ms/step - loss: 0.7219
- accuracy: 0.8521 - val_loss: 0.6385 - val_accuracy: 0.8815
Epoch 13/30
422/422 [==============================] - 3s 6ms/step - loss: 0.6474
- accuracy: 0.8660 - val_loss: 0.5738 - val_accuracy: 0.8918
Epoch 14/30
```

```
422/422 [==============================] - 3s 6ms/step - loss: 0.5867
- accuracy: 0.8757 - val_loss: 0.5146 - val_accuracy: 0.8982
Epoch 15/30
422/422 [==============================] - 4s 9ms/step - loss: 0.5364
- accuracy: 0.8831 - val_loss: 0.4709 - val_accuracy: 0.9062
Epoch 16/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4952
- accuracy: 0.8891 - val_loss: 0.4354 - val_accuracy: 0.9087
Epoch 17/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4624
- accuracy: 0.8942 - val_loss: 0.4042 - val_accuracy: 0.9127
Epoch 18/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4326
- accuracy: 0.8997 - val_loss: 0.3803 - val_accuracy: 0.9140
Epoch 19/30
422/422 [==============================] - 4s 9ms/step - loss: 0.4083
- accuracy: 0.9039 - val_loss: 0.3591 - val_accuracy: 0.9173
Epoch 20/30
422/422 [==============================] - 3s 6ms/step - loss: 0.3885
- accuracy: 0.9070 - val_loss: 0.3413 - val_accuracy: 0.9200
Epoch 21/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3694
- accuracy: 0.9099 - val_loss: 0.3252 - val_accuracy: 0.9222
Epoch 22/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3528
- accuracy: 0.9127 - val_loss: 0.3130 - val_accuracy: 0.9225
Epoch 23/30
422/422 [==============================] - 4s 9ms/step - loss: 0.3387
- accuracy: 0.9153 - val_loss: 0.2979 - val_accuracy: 0.9282
Epoch 24/30
422/422 [==============================] - 3s 6ms/step - loss: 0.3269
- accuracy: 0.9172 - val_loss: 0.2885 - val_accuracy: 0.9277
Epoch 25/30
422/422 [==============================] - 3s 6ms/step - loss: 0.3142
- accuracy: 0.9198 - val_loss: 0.2776 - val_accuracy: 0.9292
Epoch 26/30
422/422 [==============================] - 3s 6ms/step - loss: 0.3029
- accuracy: 0.9218 - val_loss: 0.2717 - val_accuracy: 0.9323
Epoch 27/30
422/422 [==============================] - 4s 8ms/step - loss: 0.2949
- accuracy: 0.9235 - val_loss: 0.2620 - val_accuracy: 0.9328
Epoch 28/30
422/422 [==============================] - 3s 8ms/step - loss: 0.2856
- accuracy: 0.9256 - val_loss: 0.2535 - val_accuracy: 0.9345
Epoch 29/30
422/422 [==============================] - 3s 7ms/step - loss: 0.2780
- accuracy: 0.9273 - val_loss: 0.2449 - val_accuracy: 0.9377
Epoch 30/30
422/422 [==============================] - 3s 6ms/step - loss: 0.2698
- accuracy: 0.9290 - val_loss: 0.2431 - val_accuracy: 0.9400
```

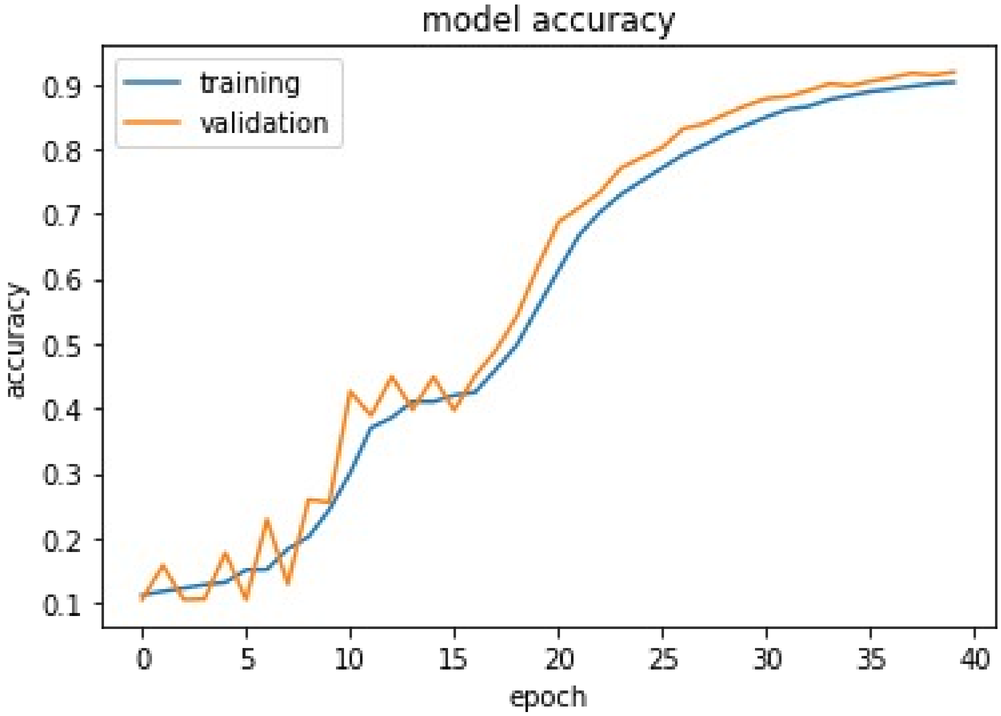


```
Test loss: 0.279
Test accuracy: 0.928
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_23"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_71 (Dense)            (None, 128)               100480

 dense_72 (Dense)            (None, 128)               16512

 dense_73 (Dense)            (None, 128)               16512

 dense_74 (Dense)            (None, 128)               16512

 dense_75 (Dense)            (None, 10)                1290

=================================================================
Total params: 151,306
Trainable params: 151,306
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 5s 10ms/step - loss: 2.3051
- accuracy: 0.1120 - val_loss: 2.3011 - val_accuracy: 0.1050
```

```
Epoch 2/40
422/422 [==============================] - 3s 8ms/step - loss: 2.2987
- accuracy: 0.1178 - val_loss: 2.2963 - val_accuracy: 0.1578
Epoch 3/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2950
- accuracy: 0.1228 - val_loss: 2.2932 - val_accuracy: 0.1050
Epoch 4/40
422/422 [==============================] - 3s 8ms/step - loss: 2.2912
- accuracy: 0.1281 - val_loss: 2.2889 - val_accuracy: 0.1060
Epoch 5/40
422/422 [==============================] - 3s 8ms/step - loss: 2.2869
- accuracy: 0.1317 - val_loss: 2.2840 - val_accuracy: 0.1773
Epoch 6/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2818
- accuracy: 0.1509 - val_loss: 2.2793 - val_accuracy: 0.1050
Epoch 7/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2759
- accuracy: 0.1514 - val_loss: 2.2718 - val_accuracy: 0.2287
Epoch 8/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2683
- accuracy: 0.1831 - val_loss: 2.2637 - val_accuracy: 0.1288
Epoch 9/40
422/422 [==============================] - 4s 10ms/step - loss: 2.2585
- accuracy: 0.2019 - val_loss: 2.2519 - val_accuracy: 0.2592
Epoch 10/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2449
- accuracy: 0.2439 - val_loss: 2.2363 - val_accuracy: 0.2553
Epoch 11/40
422/422 [==============================] - 3s 7ms/step - loss: 2.2264
- accuracy: 0.3005 - val_loss: 2.2132 - val_accuracy: 0.4260
Epoch 12/40
422/422 [==============================] - 3s 8ms/step - loss: 2.1986
- accuracy: 0.3699 - val_loss: 2.1799 - val_accuracy: 0.3888
Epoch 13/40
422/422 [==============================] - 4s 8ms/step - loss: 2.1560
- accuracy: 0.3855 - val_loss: 2.1277 - val_accuracy: 0.4488
Epoch 14/40
422/422 [==============================] - 3s 7ms/step - loss: 2.0917
- accuracy: 0.4107 - val_loss: 2.0499 - val_accuracy: 0.3980
Epoch 15/40
422/422 [==============================] - 3s 7ms/step - loss: 2.0010
- accuracy: 0.4103 - val_loss: 1.9491 - val_accuracy: 0.4483
Epoch 16/40
422/422 [==============================] - 3s 8ms/step - loss: 1.8908
- accuracy: 0.4201 - val_loss: 1.8307 - val_accuracy: 0.3970
Epoch 17/40
422/422 [==============================] - 4s 9ms/step - loss: 1.7669
- accuracy: 0.4247 - val_loss: 1.6984 - val_accuracy: 0.4502
Epoch 18/40
422/422 [==============================] - 3s 7ms/step - loss: 1.6371
```

```
- accuracy: 0.4597 - val_loss: 1.5693 - val_accuracy: 0.4893
Epoch 19/40
422/422 [==============================] - 3s 7ms/step - loss: 1.5172
- accuracy: 0.4978 - val_loss: 1.4514 - val_accuracy: 0.5420
Epoch 20/40
422/422 [==============================] - 3s 8ms/step - loss: 1.4098
- accuracy: 0.5554 - val_loss: 1.3433 - val_accuracy: 0.6178
Epoch 21/40
422/422 [==============================] - 4s 8ms/step - loss: 1.3076
- accuracy: 0.6126 - val_loss: 1.2362 - val_accuracy: 0.6870
Epoch 22/40
422/422 [==============================] - 3s 7ms/step - loss: 1.2064
- accuracy: 0.6673 - val_loss: 1.1326 - val_accuracy: 0.7093
Epoch 23/40
422/422 [==============================] - 3s 7ms/step - loss: 1.1088
- accuracy: 0.7023 - val_loss: 1.0324 - val_accuracy: 0.7332
Epoch 24/40
422/422 [==============================] - 4s 8ms/step - loss: 1.0198
- accuracy: 0.7297 - val_loss: 0.9479 - val_accuracy: 0.7705
Epoch 25/40
422/422 [==============================] - 3s 8ms/step - loss: 0.9437
- accuracy: 0.7505 - val_loss: 0.8758 - val_accuracy: 0.7865
Epoch 26/40
422/422 [==============================] - 3s 7ms/step - loss: 0.8776
- accuracy: 0.7710 - val_loss: 0.8095 - val_accuracy: 0.8022
Epoch 27/40
422/422 [==============================] - 3s 7ms/step - loss: 0.8179
- accuracy: 0.7909 - val_loss: 0.7524 - val_accuracy: 0.8312
Epoch 28/40
422/422 [==============================] - 3s 8ms/step - loss: 0.7684
- accuracy: 0.8060 - val_loss: 0.7036 - val_accuracy: 0.8385
Epoch 29/40
422/422 [==============================] - 4s 9ms/step - loss: 0.7207
- accuracy: 0.8227 - val_loss: 0.6621 - val_accuracy: 0.8530
Epoch 30/40
422/422 [==============================] - 3s 7ms/step - loss: 0.6779
- accuracy: 0.8362 - val_loss: 0.6172 - val_accuracy: 0.8667
Epoch 31/40
422/422 [==============================] - 4s 8ms/step - loss: 0.6394
- accuracy: 0.8494 - val_loss: 0.5804 - val_accuracy: 0.8777
Epoch 32/40
422/422 [==============================] - 5s 11ms/step - loss: 0.6028
- accuracy: 0.8607 - val_loss: 0.5489 - val_accuracy: 0.8807
Epoch 33/40
422/422 [==============================] - 3s 7ms/step - loss: 0.5753
- accuracy: 0.8655 - val_loss: 0.5187 - val_accuracy: 0.8898
Epoch 34/40
422/422 [==============================] - 3s 7ms/step - loss: 0.5423
- accuracy: 0.8759 - val_loss: 0.4873 - val_accuracy: 0.9003
Epoch 35/40
```

```
422/422 [==============================] - 3s 7ms/step - loss: 0.5138
- accuracy: 0.8823 - val_loss: 0.4650 - val_accuracy: 0.8973
Epoch 36/40
422/422 [==============================] - 4s 10ms/step - loss: 0.4877
- accuracy: 0.8885 - val_loss: 0.4412 - val_accuracy: 0.9040
Epoch 37/40
422/422 [==============================] - 3s 7ms/step - loss: 0.4666
- accuracy: 0.8924 - val_loss: 0.4219 - val_accuracy: 0.9097
Epoch 38/40
422/422 [==============================] - 3s 7ms/step - loss: 0.4464
- accuracy: 0.8966 - val_loss: 0.4004 - val_accuracy: 0.9165
Epoch 39/40
422/422 [==============================] - 3s 6ms/step - loss: 0.4259
- accuracy: 0.9008 - val_loss: 0.3913 - val_accuracy: 0.9137
Epoch 40/40
422/422 [==============================] - 4s 10ms/step - loss: 0.4109
- accuracy: 0.9030 - val_loss: 0.3792 - val_accuracy: 0.9183
```

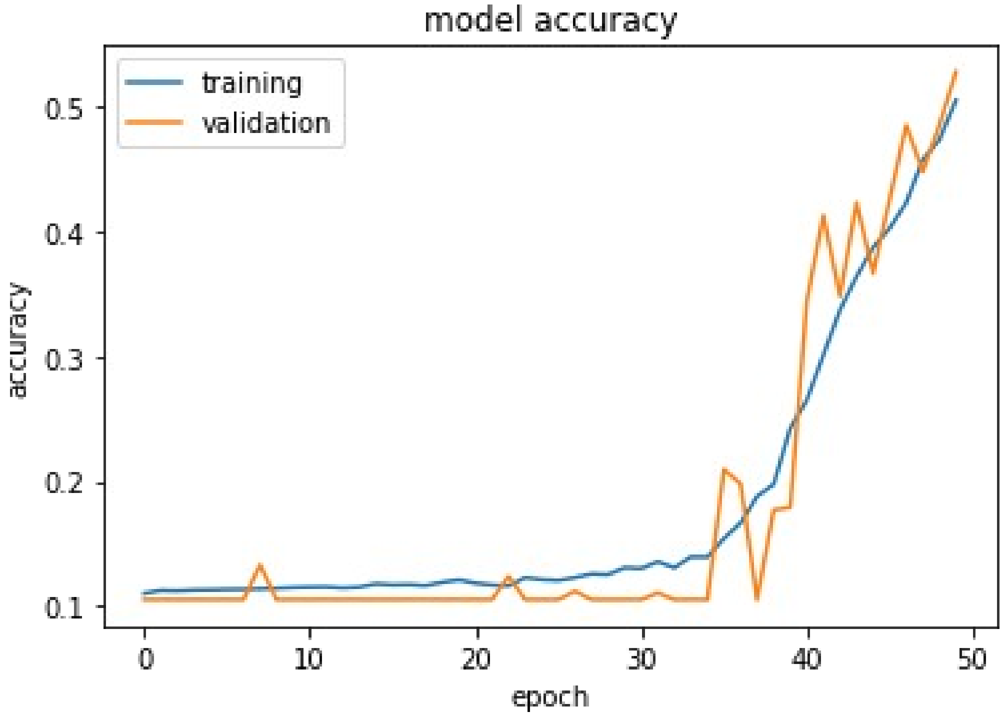


```
Test loss: 0.422
Test accuracy: 0.9
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_24"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_76 (Dense)            (None, 128)               100480

 dense_77 (Dense)            (None, 128)               16512

 dense_78 (Dense)            (None, 128)               16512

 dense_79 (Dense)            (None, 128)               16512

 dense_80 (Dense)            (None, 128)               16512

 dense_81 (Dense)            (None, 10)                1290

=================================================================
Total params: 167,818
Trainable params: 167,818
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
422/422 [==============================] - 4s 8ms/step - loss: 2.3062
- accuracy: 0.1100 - val_loss: 2.3022 - val_accuracy: 0.1050
Epoch 2/50
422/422 [==============================] - 3s 8ms/step - loss: 2.3013
- accuracy: 0.1124 - val_loss: 2.3045 - val_accuracy: 0.1050
Epoch 3/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3012
- accuracy: 0.1121 - val_loss: 2.3020 - val_accuracy: 0.1050
Epoch 4/50
422/422 [==============================] - 4s 8ms/step - loss: 2.3010
- accuracy: 0.1127 - val_loss: 2.3030 - val_accuracy: 0.1050
Epoch 5/50
422/422 [==============================] - 3s 7ms/step - loss: 2.3008
- accuracy: 0.1129 - val_loss: 2.3015 - val_accuracy: 0.1050
Epoch 6/50
422/422 [==============================] - 3s 8ms/step - loss: 2.3006
- accuracy: 0.1133 - val_loss: 2.3010 - val_accuracy: 0.1050
Epoch 7/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3004
- accuracy: 0.1133 - val_loss: 2.3002 - val_accuracy: 0.1050
Epoch 8/50
422/422 [==============================] - 3s 7ms/step - loss: 2.3001
- accuracy: 0.1138 - val_loss: 2.2999 - val_accuracy: 0.1328
Epoch 9/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2998
- accuracy: 0.1141 - val_loss: 2.3010 - val_accuracy: 0.1050
Epoch 10/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2994
- accuracy: 0.1148 - val_loss: 2.2997 - val_accuracy: 0.1050
Epoch 11/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2994
- accuracy: 0.1151 - val_loss: 2.2987 - val_accuracy: 0.1050
```

```
Epoch 12/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2990
- accuracy: 0.1152 - val_loss: 2.2995 - val_accuracy: 0.1050
Epoch 13/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2988
- accuracy: 0.1144 - val_loss: 2.3000 - val_accuracy: 0.1050
Epoch 14/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2984
- accuracy: 0.1151 - val_loss: 2.2983 - val_accuracy: 0.1050
Epoch 15/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2979
- accuracy: 0.1176 - val_loss: 2.2998 - val_accuracy: 0.1050
Epoch 16/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2977
- accuracy: 0.1168 - val_loss: 2.2984 - val_accuracy: 0.1050
Epoch 17/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2973
- accuracy: 0.1171 - val_loss: 2.2975 - val_accuracy: 0.1050
Epoch 18/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2969
- accuracy: 0.1162 - val_loss: 2.2971 - val_accuracy: 0.1050
Epoch 19/50
422/422 [==============================] - 4s 9ms/step - loss: 2.2965
- accuracy: 0.1189 - val_loss: 2.2966 - val_accuracy: 0.1050
Epoch 20/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2961
- accuracy: 0.1209 - val_loss: 2.2962 - val_accuracy: 0.1050
Epoch 21/50
422/422 [==============================] - 3s 7ms/step - loss: 2.2955
- accuracy: 0.1182 - val_loss: 2.2965 - val_accuracy: 0.1050
Epoch 22/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2950
- accuracy: 0.1167 - val_loss: 2.2951 - val_accuracy: 0.1050
Epoch 23/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2944
- accuracy: 0.1161 - val_loss: 2.2947 - val_accuracy: 0.1237
Epoch 24/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2936
- accuracy: 0.1227 - val_loss: 2.2942 - val_accuracy: 0.1050
Epoch 25/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2929
- accuracy: 0.1211 - val_loss: 2.2929 - val_accuracy: 0.1050
Epoch 26/50
422/422 [==============================] - 5s 11ms/step - loss: 2.2922
- accuracy: 0.1204 - val_loss: 2.2915 - val_accuracy: 0.1050
Epoch 27/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2911
- accuracy: 0.1226 - val_loss: 2.2907 - val_accuracy: 0.1118
Epoch 28/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2899
```

```
- accuracy: 0.1256 - val_loss: 2.2900 - val_accuracy: 0.1050
Epoch 29/50
422/422 [==============================] - 5s 11ms/step - loss: 2.2888
- accuracy: 0.1251 - val_loss: 2.2886 - val_accuracy: 0.1050
Epoch 30/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2874
- accuracy: 0.1306 - val_loss: 2.2869 - val_accuracy: 0.1050
Epoch 31/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2857
- accuracy: 0.1300 - val_loss: 2.2854 - val_accuracy: 0.1050
Epoch 32/50
422/422 [==============================] - 4s 9ms/step - loss: 2.2837
- accuracy: 0.1354 - val_loss: 2.2834 - val_accuracy: 0.1105
Epoch 33/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2817
- accuracy: 0.1306 - val_loss: 2.2813 - val_accuracy: 0.1050
Epoch 34/50
422/422 [==============================] - 4s 8ms/step - loss: 2.2788
- accuracy: 0.1393 - val_loss: 2.2780 - val_accuracy: 0.1050
Epoch 35/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2759
- accuracy: 0.1392 - val_loss: 2.2741 - val_accuracy: 0.1050
Epoch 36/50
422/422 [==============================] - 5s 11ms/step - loss: 2.2718
- accuracy: 0.1542 - val_loss: 2.2686 - val_accuracy: 0.2095
Epoch 37/50
422/422 [==============================] - 4s 8ms/step - loss: 2.2669
- accuracy: 0.1666 - val_loss: 2.2640 - val_accuracy: 0.1980
Epoch 38/50
422/422 [==============================] - 4s 9ms/step - loss: 2.2605
- accuracy: 0.1880 - val_loss: 2.2576 - val_accuracy: 0.1050
Epoch 39/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2524
- accuracy: 0.1976 - val_loss: 2.2472 - val_accuracy: 0.1770
Epoch 40/50
422/422 [==============================] - 4s 9ms/step - loss: 2.2412
- accuracy: 0.2425 - val_loss: 2.2348 - val_accuracy: 0.1790
Epoch 41/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2256
- accuracy: 0.2651 - val_loss: 2.2161 - val_accuracy: 0.3448
Epoch 42/50
422/422 [==============================] - 3s 8ms/step - loss: 2.2030
- accuracy: 0.3013 - val_loss: 2.1888 - val_accuracy: 0.4135
Epoch 43/50
422/422 [==============================] - 5s 11ms/step - loss: 2.1692
- accuracy: 0.3374 - val_loss: 2.1476 - val_accuracy: 0.3483
Epoch 44/50
422/422 [==============================] - 3s 8ms/step - loss: 2.1178
- accuracy: 0.3643 - val_loss: 2.0846 - val_accuracy: 0.4235
Epoch 45/50
```

```
422/422 [==============================] - 3s 8ms/step - loss: 2.0423
- accuracy: 0.3875 - val_loss: 1.9973 - val_accuracy: 0.3670
Epoch 46/50
422/422 [==============================] - 4s 10ms/step - loss: 1.9433
- accuracy: 0.4030 - val_loss: 1.8849 - val_accuracy: 0.4263
Epoch 47/50
422/422 [==============================] - 4s 10ms/step - loss: 1.8292
- accuracy: 0.4234 - val_loss: 1.7621 - val_accuracy: 0.4857
Epoch 48/50
422/422 [==============================] - 4s 9ms/step - loss: 1.7109
- accuracy: 0.4581 - val_loss: 1.6411 - val_accuracy: 0.4485
Epoch 49/50
422/422 [==============================] - 4s 10ms/step - loss: 1.6004
- accuracy: 0.4739 - val_loss: 1.5319 - val_accuracy: 0.4857
Epoch 50/50
422/422 [==============================] - 4s 9ms/step - loss: 1.5020
- accuracy: 0.5057 - val_loss: 1.4345 - val_accuracy: 0.5283
```



```
Test loss: 1.45
Test accuracy: 0.521
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

Dicusión

Test loss: 0.246 Test accuracy: 0.932

Test loss: 0.252 Test accuracy: 0.93

Test loss: 0.279 Test accuracy: 0.928

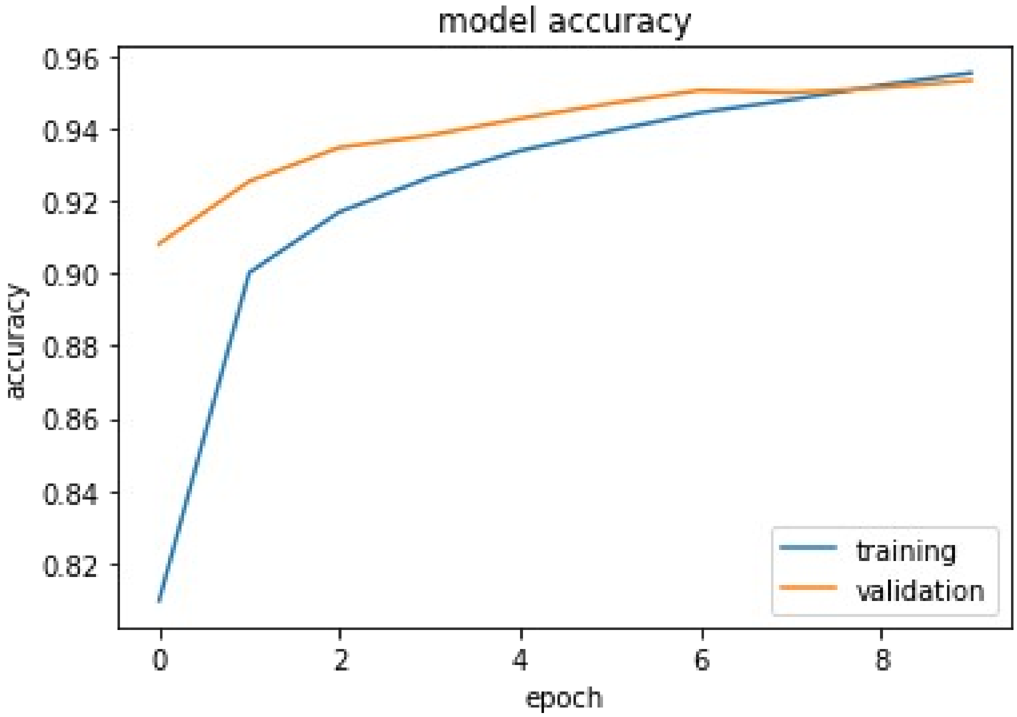
Test loss: 0.422 Test accuracy: 0.9

Test loss: 1.45 Test accuracy: 0.521

```
#PARTE 4.3 ESCRIBA SU CÓDIGO AQUÍ.
for layers in [1, 2, 3, 4, 5]:
    model = create_dense([512] * layers)
    evaluate(model, batch_size=128, epochs=10*layers, verbose=True)
#verbose por defecto es false
```

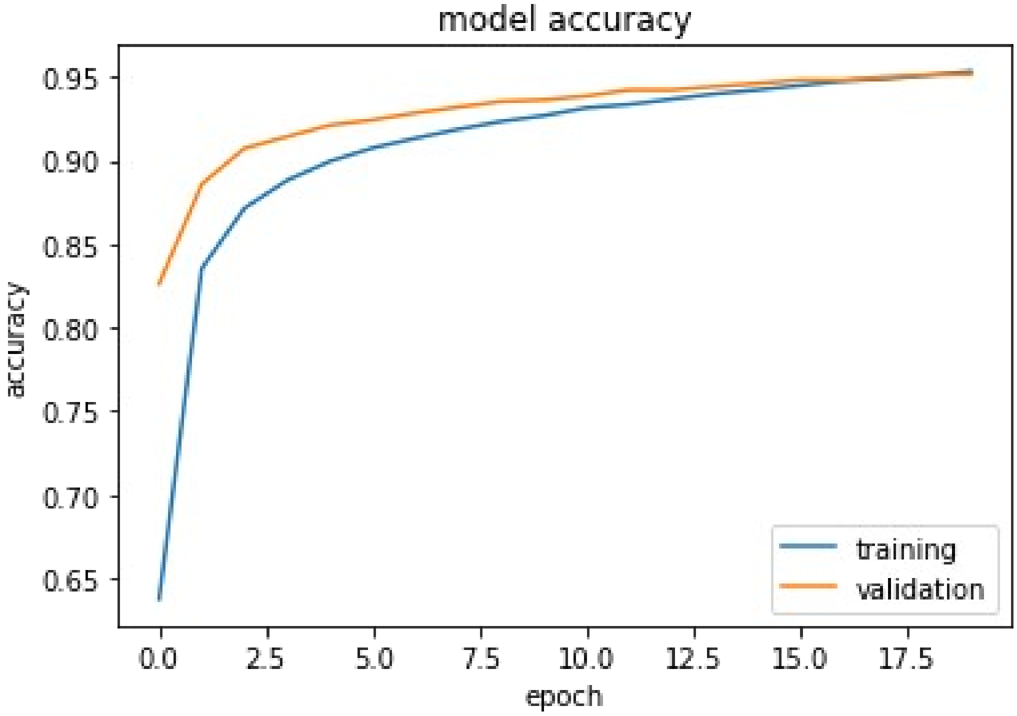
```
Model: "sequential_26"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_84 (Dense)            (None, 512)               401920

 dense_85 (Dense)            (None, 10)                5130

=================================================================
Total params: 407,050
Trainable params: 407,050
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 5s 11ms/step - loss: 0.7327
- accuracy: 0.8096 - val_loss: 0.3734 - val_accuracy: 0.9082
Epoch 2/10
422/422 [==============================] - 4s 9ms/step - loss: 0.3783
- accuracy: 0.9002 - val_loss: 0.2888 - val_accuracy: 0.9255
Epoch 3/10
422/422 [==============================] - 4s 10ms/step - loss: 0.3097
- accuracy: 0.9170 - val_loss: 0.2503 - val_accuracy: 0.9348
Epoch 4/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2711
- accuracy: 0.9265 - val_loss: 0.2246 - val_accuracy: 0.9382
Epoch 5/10
422/422 [==============================] - 4s 9ms/step - loss: 0.2435
- accuracy: 0.9339 - val_loss: 0.2078 - val_accuracy: 0.9428
Epoch 6/10
422/422 [==============================] - 4s 9ms/step - loss: 0.2224
- accuracy: 0.9394 - val_loss: 0.1974 - val_accuracy: 0.9470
Epoch 7/10
422/422 [==============================] - 5s 11ms/step - loss: 0.2067
```

```
- accuracy: 0.9445 - val_loss: 0.1858 - val_accuracy: 0.9507
Epoch 8/10
422/422 [==============================] - 4s 9ms/step - loss: 0.1929
- accuracy: 0.9480 - val_loss: 0.1801 - val_accuracy: 0.9500
Epoch 9/10
422/422 [==============================] - 4s 9ms/step - loss: 0.1807
- accuracy: 0.9520 - val_loss: 0.1745 - val_accuracy: 0.9513
Epoch 10/10
422/422 [==============================] - 5s 11ms/step - loss: 0.1699
- accuracy: 0.9554 - val_loss: 0.1669 - val_accuracy: 0.9533
```



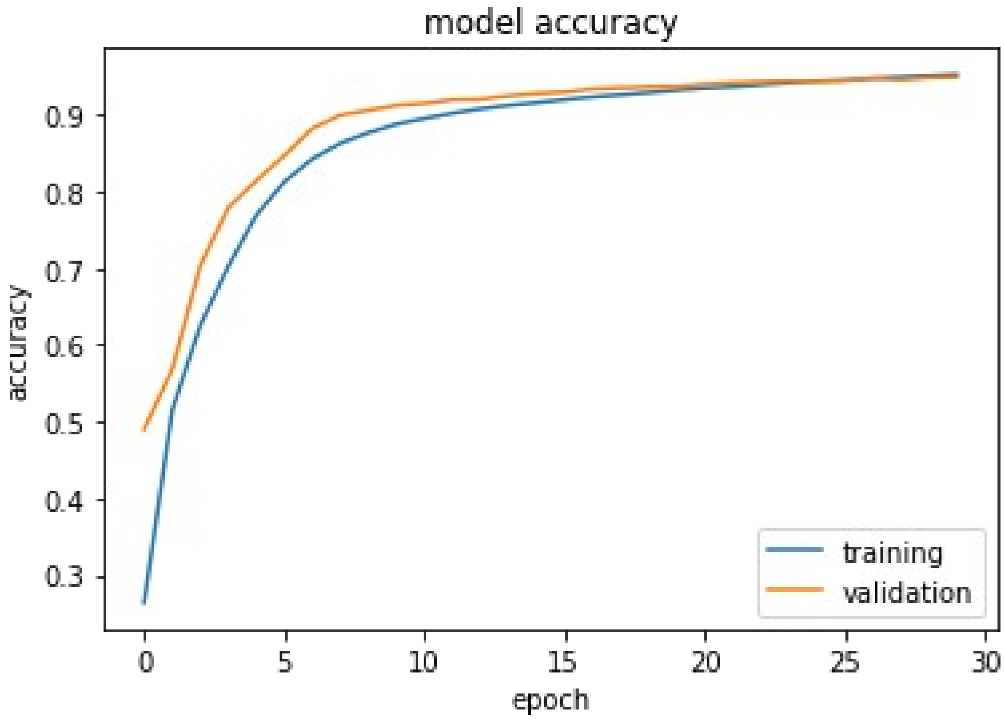
```
Test loss: 0.192
Test accuracy: 0.945
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_27"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_86 (Dense)            (None, 512)               401920

 dense_87 (Dense)            (None, 512)               262656

 dense_88 (Dense)            (None, 10)                5130

=================================================================
Total params: 669,706
Trainable params: 669,706
Non-trainable params: 0
_________________________________________________________________
Epoch 1/20
422/422 [==============================] - 7s 17ms/step - loss: 1.6736
- accuracy: 0.6373 - val_loss: 1.1118 - val_accuracy: 0.8262
Epoch 2/20
422/422 [==============================] - 7s 16ms/step - loss: 0.8927
- accuracy: 0.8354 - val_loss: 0.6423 - val_accuracy: 0.8858
Epoch 3/20
422/422 [==============================] - 7s 16ms/step - loss: 0.6051
- accuracy: 0.8715 - val_loss: 0.4641 - val_accuracy: 0.9073
Epoch 4/20
422/422 [==============================] - 6s 15ms/step - loss: 0.4780
- accuracy: 0.8882 - val_loss: 0.3789 - val_accuracy: 0.9143
Epoch 5/20
422/422 [==============================] - 7s 17ms/step - loss: 0.4076
- accuracy: 0.8995 - val_loss: 0.3292 - val_accuracy: 0.9212
Epoch 6/20
422/422 [==============================] - 6s 15ms/step - loss: 0.3617
- accuracy: 0.9075 - val_loss: 0.2990 - val_accuracy: 0.9243
Epoch 7/20
422/422 [==============================] - 7s 17ms/step - loss: 0.3304
- accuracy: 0.9133 - val_loss: 0.2763 - val_accuracy: 0.9285
Epoch 8/20
422/422 [==============================] - 6s 14ms/step - loss: 0.3056
- accuracy: 0.9187 - val_loss: 0.2563 - val_accuracy: 0.9320
Epoch 9/20
422/422 [==============================] - 7s 17ms/step - loss: 0.2854
- accuracy: 0.9234 - val_loss: 0.2435 - val_accuracy: 0.9353
Epoch 10/20
422/422 [==============================] - 6s 15ms/step - loss: 0.2690
- accuracy: 0.9269 - val_loss: 0.2313 - val_accuracy: 0.9363
Epoch 11/20
422/422 [==============================] - 7s 17ms/step - loss: 0.2544
- accuracy: 0.9316 - val_loss: 0.2236 - val_accuracy: 0.9387
Epoch 12/20
422/422 [==============================] - 6s 14ms/step - loss: 0.2421
- accuracy: 0.9336 - val_loss: 0.2133 - val_accuracy: 0.9423
Epoch 13/20
422/422 [==============================] - 7s 17ms/step - loss: 0.2310
- accuracy: 0.9368 - val_loss: 0.2067 - val_accuracy: 0.9423
Epoch 14/20
422/422 [==============================] - 6s 14ms/step - loss: 0.2198
- accuracy: 0.9400 - val_loss: 0.2001 - val_accuracy: 0.9443
Epoch 15/20
422/422 [==============================] - 7s 17ms/step - loss: 0.2109
- accuracy: 0.9423 - val_loss: 0.1932 - val_accuracy: 0.9462
Epoch 16/20
```

```
422/422 [==============================] - 6s 14ms/step - loss: 0.2028
- accuracy: 0.9450 - val_loss: 0.1897 - val_accuracy: 0.9482
Epoch 17/20
422/422 [==============================] - 7s 17ms/step - loss: 0.1942
- accuracy: 0.9477 - val_loss: 0.1859 - val_accuracy: 0.9480
Epoch 18/20
422/422 [==============================] - 6s 14ms/step - loss: 0.1877
- accuracy: 0.9489 - val_loss: 0.1801 - val_accuracy: 0.9500
Epoch 19/20
422/422 [==============================] - 9s 20ms/step - loss: 0.1807
- accuracy: 0.9510 - val_loss: 0.1774 - val_accuracy: 0.9512
Epoch 20/20
422/422 [==============================] - 6s 14ms/step - loss: 0.1743
- accuracy: 0.9533 - val_loss: 0.1736 - val_accuracy: 0.9515
```



```
Test loss: 0.203
Test accuracy: 0.943
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_28"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_89 (Dense)            (None, 512)               401920

 dense_90 (Dense)            (None, 512)               262656
 dense_91 (Dense)            (None, 512)               262656

 dense_92 (Dense)            (None, 10)                5130

=================================================================
Total params: 932,362
Trainable params: 932,362
Non-trainable params: 0
_________________________________________________________________
Epoch 1/30
422/422 [==============================] - 11s 24ms/step - loss:
2.2465 - accuracy: 0.2641 - val_loss: 2.1699 - val_accuracy: 0.4898
Epoch 2/30
422/422 [==============================] - 10s 23ms/step - loss:
2.0720 - accuracy: 0.5149 - val_loss: 1.9383 - val_accuracy: 0.5680
Epoch 3/30
422/422 [==============================] - 9s 21ms/step - loss: 1.7745
- accuracy: 0.6252 - val_loss: 1.5617 - val_accuracy: 0.7033
Epoch 4/30
422/422 [==============================] - 10s 23ms/step - loss:
1.3886 - accuracy: 0.7023 - val_loss: 1.1734 - val_accuracy: 0.7787
Epoch 5/30
422/422 [==============================] - 10s 23ms/step - loss:
1.0653 - accuracy: 0.7689 - val_loss: 0.8956 - val_accuracy: 0.8140
Epoch 6/30
422/422 [==============================] - 10s 23ms/step - loss:
0.8472 - accuracy: 0.8124 - val_loss: 0.7132 - val_accuracy: 0.8467
Epoch 7/30
422/422 [==============================] - 9s 21ms/step - loss: 0.6993
- accuracy: 0.8422 - val_loss: 0.5849 - val_accuracy: 0.8820
Epoch 8/30
422/422 [==============================] - 10s 24ms/step - loss:
0.5938 - accuracy: 0.8627 - val_loss: 0.4957 - val_accuracy: 0.8995
Epoch 9/30
422/422 [==============================] - 10s 23ms/step - loss:
0.5169 - accuracy: 0.8764 - val_loss: 0.4323 - val_accuracy: 0.9055
Epoch 10/30
422/422 [==============================] - 10s 23ms/step - loss:
0.4601 - accuracy: 0.8876 - val_loss: 0.3877 - val_accuracy: 0.9122
Epoch 11/30
422/422 [==============================] - 9s 21ms/step - loss: 0.4177
- accuracy: 0.8951 - val_loss: 0.3527 - val_accuracy: 0.9147
Epoch 12/30
422/422 [==============================] - 10s 24ms/step - loss:
0.3837 - accuracy: 0.9018 - val_loss: 0.3245 - val_accuracy: 0.9193
Epoch 13/30
422/422 [==============================] - 10s 23ms/step - loss:
0.3574 - accuracy: 0.9076 - val_loss: 0.3067 - val_accuracy: 0.9202
Epoch 14/30
```

```
422/422 [==============================] - 10s 23ms/step - loss:
0.3347 - accuracy: 0.9122 - val_loss: 0.2887 - val_accuracy: 0.9245
Epoch 15/30
422/422 [==============================] - 9s 20ms/step - loss: 0.3164
- accuracy: 0.9156 - val_loss: 0.2740 - val_accuracy: 0.9275
Epoch 16/30
422/422 [==============================] - 10s 23ms/step - loss:
0.3003 - accuracy: 0.9195 - val_loss: 0.2621 - val_accuracy: 0.9293
Epoch 17/30
422/422 [==============================] - 10s 23ms/step - loss:
0.2857 - accuracy: 0.9231 - val_loss: 0.2521 - val_accuracy: 0.9335
Epoch 18/30
422/422 [==============================] - 10s 23ms/step - loss:
0.2732 - accuracy: 0.9259 - val_loss: 0.2435 - val_accuracy: 0.9348
Epoch 19/30
422/422 [==============================] - 9s 21ms/step - loss: 0.2615
- accuracy: 0.9288 - val_loss: 0.2339 - val_accuracy: 0.9362
Epoch 20/30
422/422 [==============================] - 10s 24ms/step - loss:
0.2510 - accuracy: 0.9320 - val_loss: 0.2286 - val_accuracy: 0.9365
Epoch 21/30
422/422 [==============================] - 11s 25ms/step - loss:
0.2413 - accuracy: 0.9346 - val_loss: 0.2218 - val_accuracy: 0.9392
Epoch 22/30
422/422 [==============================] - 10s 23ms/step - loss:
0.2329 - accuracy: 0.9364 - val_loss: 0.2154 - val_accuracy: 0.9413
Epoch 23/30
422/422 [==============================] - 9s 21ms/step - loss: 0.2241
- accuracy: 0.9391 - val_loss: 0.2104 - val_accuracy: 0.9427
Epoch 24/30
422/422 [==============================] - 10s 23ms/step - loss:
0.2163 - accuracy: 0.9415 - val_loss: 0.2051 - val_accuracy: 0.9437
Epoch 25/30
422/422 [==============================] - 10s 23ms/step - loss:
0.2092 - accuracy: 0.9436 - val_loss: 0.2008 - val_accuracy: 0.9430
Epoch 26/30
422/422 [==============================] - 9s 22ms/step - loss: 0.2025
- accuracy: 0.9445 - val_loss: 0.1960 - val_accuracy: 0.9443
Epoch 27/30
422/422 [==============================] - 9s 21ms/step - loss: 0.1951
- accuracy: 0.9474 - val_loss: 0.1922 - val_accuracy: 0.9472
Epoch 28/30
422/422 [==============================] - 10s 23ms/step - loss:
0.1894 - accuracy: 0.9489 - val_loss: 0.1894 - val_accuracy: 0.9452
Epoch 29/30
422/422 [==============================] - 10s 23ms/step - loss:
0.1835 - accuracy: 0.9505 - val_loss: 0.1859 - val_accuracy: 0.9478
Epoch 30/30
422/422 [==============================] - 9s 21ms/step - loss: 0.1780
- accuracy: 0.9521 - val_loss: 0.1844 - val_accuracy: 0.9485
```

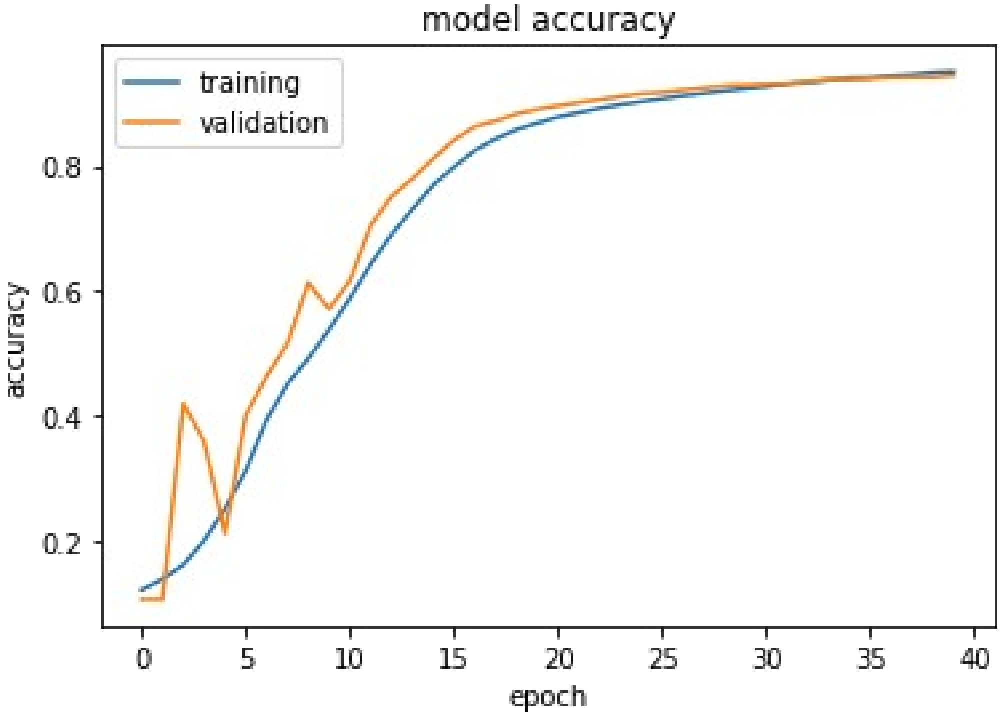


```
Test loss: 0.218
Test accuracy: 0.935
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_29"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_93 (Dense)            (None, 512)               401920    
                                                                 
 dense_94 (Dense)            (None, 512)               262656    
                                                                 
 dense_95 (Dense)            (None, 512)               262656    
                                                                 
 dense_96 (Dense)            (None, 512)               262656    
                                                                 
 dense_97 (Dense)            (None, 10)                5130      
                                                                 
=================================================================
Total params: 1,195,018
Trainable params: 1,195,018
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 13s 29ms/step - loss:
2.2995 - accuracy: 0.1204 - val_loss: 2.2961 - val_accuracy: 0.1050
```

```
Epoch 2/40
422/422 [==============================] - 12s 28ms/step - loss:
2.2901 - accuracy: 0.1378 - val_loss: 2.2854 - val_accuracy: 0.1050
Epoch 3/40
422/422 [==============================] - 11s 26ms/step - loss:
2.2797 - accuracy: 0.1607 - val_loss: 2.2716 - val_accuracy: 0.4190
Epoch 4/40
422/422 [==============================] - 11s 27ms/step - loss:
2.2657 - accuracy: 0.1998 - val_loss: 2.2600 - val_accuracy: 0.3578
Epoch 5/40
422/422 [==============================] - 12s 28ms/step - loss:
2.2465 - accuracy: 0.2508 - val_loss: 2.2324 - val_accuracy: 0.2105
Epoch 6/40
422/422 [==============================] - 12s 29ms/step - loss:
2.2165 - accuracy: 0.3120 - val_loss: 2.1926 - val_accuracy: 0.4002
Epoch 7/40
422/422 [==============================] - 12s 29ms/step - loss:
2.1654 - accuracy: 0.3932 - val_loss: 2.1234 - val_accuracy: 0.4628
Epoch 8/40
422/422 [==============================] - 12s 29ms/step - loss:
2.0712 - accuracy: 0.4504 - val_loss: 1.9973 - val_accuracy: 0.5155
Epoch 9/40
422/422 [==============================] - 12s 29ms/step - loss:
1.9043 - accuracy: 0.4912 - val_loss: 1.7864 - val_accuracy: 0.6125
Epoch 10/40
422/422 [==============================] - 12s 28ms/step - loss:
1.6746 - accuracy: 0.5375 - val_loss: 1.5415 - val_accuracy: 0.5708
Epoch 11/40
422/422 [==============================] - 12s 29ms/step - loss:
1.4538 - accuracy: 0.5879 - val_loss: 1.3372 - val_accuracy: 0.6162
Epoch 12/40
422/422 [==============================] - 12s 29ms/step - loss:
1.2797 - accuracy: 0.6428 - val_loss: 1.1779 - val_accuracy: 0.7043
Epoch 13/40
422/422 [==============================] - 12s 29ms/step - loss:
1.1311 - accuracy: 0.6901 - val_loss: 1.0269 - val_accuracy: 0.7517
Epoch 14/40
422/422 [==============================] - 12s 29ms/step - loss:
0.9926 - accuracy: 0.7305 - val_loss: 0.8859 - val_accuracy: 0.7797
Epoch 15/40
422/422 [==============================] - 12s 28ms/step - loss:
0.8672 - accuracy: 0.7696 - val_loss: 0.7638 - val_accuracy: 0.8120
Epoch 16/40
422/422 [==============================] - 12s 28ms/step - loss:
0.7639 - accuracy: 0.7984 - val_loss: 0.6690 - val_accuracy: 0.8420
Epoch 17/40
422/422 [==============================] - 12s 28ms/step - loss:
0.6825 - accuracy: 0.8250 - val_loss: 0.5948 - val_accuracy: 0.8638
Epoch 18/40
422/422 [==============================] - 11s 27ms/step - loss:
```

```
0.6186 - accuracy: 0.8439 - val_loss: 0.5398 - val_accuracy: 0.8727
Epoch 19/40
422/422 [==============================] - 12s 27ms/step - loss:
0.5665 - accuracy: 0.8588 - val_loss: 0.4947 - val_accuracy: 0.8838
Epoch 20/40
422/422 [==============================] - 12s 28ms/step - loss:
0.5227 - accuracy: 0.8693 - val_loss: 0.4557 - val_accuracy: 0.8913
Epoch 21/40
422/422 [==============================] - 12s 28ms/step - loss:
0.4860 - accuracy: 0.8790 - val_loss: 0.4261 - val_accuracy: 0.8968
Epoch 22/40
422/422 [==============================] - 12s 29ms/step - loss:
0.4554 - accuracy: 0.8859 - val_loss: 0.3986 - val_accuracy: 0.9022
Epoch 23/40
422/422 [==============================] - 12s 28ms/step - loss:
0.4283 - accuracy: 0.8929 - val_loss: 0.3785 - val_accuracy: 0.9068
Epoch 24/40
422/422 [==============================] - 12s 29ms/step - loss:
0.4045 - accuracy: 0.8987 - val_loss: 0.3617 - val_accuracy: 0.9118
Epoch 25/40
422/422 [==============================] - 12s 29ms/step - loss:
0.3833 - accuracy: 0.9036 - val_loss: 0.3426 - val_accuracy: 0.9163
Epoch 26/40
422/422 [==============================] - 13s 31ms/step - loss:
0.3628 - accuracy: 0.9083 - val_loss: 0.3268 - val_accuracy: 0.9183
Epoch 27/40
422/422 [==============================] - 12s 29ms/step - loss:
0.3457 - accuracy: 0.9131 - val_loss: 0.3141 - val_accuracy: 0.9218
Epoch 28/40
422/422 [==============================] - 12s 29ms/step - loss:
0.3298 - accuracy: 0.9170 - val_loss: 0.3019 - val_accuracy: 0.9253
Epoch 29/40
422/422 [==============================] - 12s 28ms/step - loss:
0.3137 - accuracy: 0.9210 - val_loss: 0.2876 - val_accuracy: 0.9287
Epoch 30/40
422/422 [==============================] - 12s 28ms/step - loss:
0.3000 - accuracy: 0.9244 - val_loss: 0.2792 - val_accuracy: 0.9302
Epoch 31/40
422/422 [==============================] - 11s 26ms/step - loss:
0.2867 - accuracy: 0.9276 - val_loss: 0.2703 - val_accuracy: 0.9313
Epoch 32/40
422/422 [==============================] - 11s 27ms/step - loss:
0.2749 - accuracy: 0.9316 - val_loss: 0.2611 - val_accuracy: 0.9325
Epoch 33/40
422/422 [==============================] - 12s 29ms/step - loss:
0.2633 - accuracy: 0.9345 - val_loss: 0.2541 - val_accuracy: 0.9355
Epoch 34/40
422/422 [==============================] - 12s 29ms/step - loss:
0.2513 - accuracy: 0.9379 - val_loss: 0.2486 - val_accuracy: 0.9388
Epoch 35/40
```

```
422/422 [==============================] - 12s 28ms/step - loss: 
0.2417 - accuracy: 0.9411 - val_loss: 0.2416 - val_accuracy: 0.9387
Epoch 36/40
422/422 [==============================] - 12s 28ms/step - loss: 
0.2324 - accuracy: 0.9417 - val_loss: 0.2351 - val_accuracy: 0.9403
Epoch 37/40
422/422 [==============================] - 12s 28ms/step - loss: 
0.2232 - accuracy: 0.9448 - val_loss: 0.2334 - val_accuracy: 0.9425
Epoch 38/40
422/422 [==============================] - 12s 29ms/step - loss: 
0.2150 - accuracy: 0.9466 - val_loss: 0.2253 - val_accuracy: 0.9427
Epoch 39/40
422/422 [==============================] - 12s 28ms/step - loss: 
0.2059 - accuracy: 0.9496 - val_loss: 0.2246 - val_accuracy: 0.9427
Epoch 40/40
422/422 [==============================] - 12s 28ms/step - loss: 
0.1986 - accuracy: 0.9511 - val_loss: 0.2164 - val_accuracy: 0.9450
```

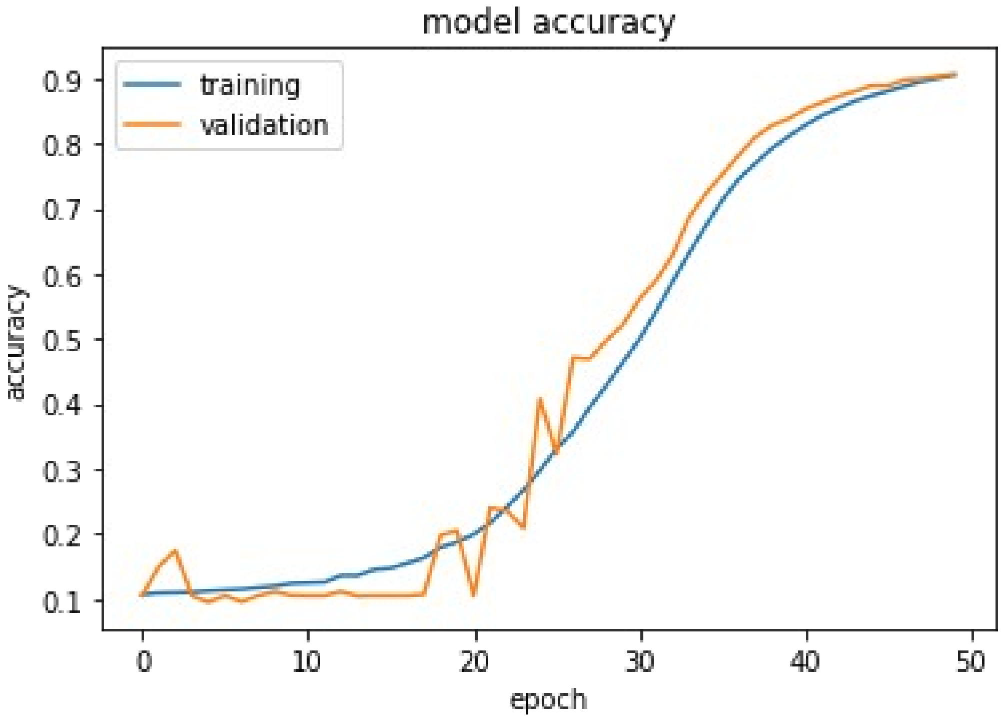


```
Test loss: 0.253
Test accuracy: 0.933
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.    ]
Model: "sequential_30"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_98 (Dense)            (None, 512)               401920

 dense_99 (Dense)            (None, 512)               262656

 dense_100 (Dense)           (None, 512)               262656

 dense_101 (Dense)           (None, 512)               262656

 dense_102 (Dense)           (None, 512)               262656

 dense_103 (Dense)           (None, 10)                5130

=================================================================
Total params: 1,457,674
Trainable params: 1,457,674
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
422/422 [==============================] - 16s 37ms/step - loss:
2.3056 - accuracy: 0.1069 - val_loss: 2.3021 - val_accuracy: 0.1050
Epoch 2/50
422/422 [==============================] - 15s 35ms/step - loss:
2.3032 - accuracy: 0.1093 - val_loss: 2.3024 - val_accuracy: 0.1497
Epoch 3/50
422/422 [==============================] - 15s 35ms/step - loss:
2.3023 - accuracy: 0.1097 - val_loss: 2.3015 - val_accuracy: 0.1745
Epoch 4/50
422/422 [==============================] - 14s 34ms/step - loss:
2.3020 - accuracy: 0.1105 - val_loss: 2.3019 - val_accuracy: 0.1050
Epoch 5/50
422/422 [==============================] - 14s 34ms/step - loss:
2.3016 - accuracy: 0.1124 - val_loss: 2.3013 - val_accuracy: 0.0952
Epoch 6/50
422/422 [==============================] - 15s 35ms/step - loss:
2.3009 - accuracy: 0.1139 - val_loss: 2.2994 - val_accuracy: 0.1050
Epoch 7/50
422/422 [==============================] - 14s 34ms/step - loss:
2.2997 - accuracy: 0.1150 - val_loss: 2.2994 - val_accuracy: 0.0960
Epoch 8/50
422/422 [==============================] - 14s 34ms/step - loss:
2.2995 - accuracy: 0.1174 - val_loss: 2.2995 - val_accuracy: 0.1050
Epoch 9/50
422/422 [==============================] - 15s 35ms/step - loss:
2.2986 - accuracy: 0.1198 - val_loss: 2.3049 - val_accuracy: 0.1113
Epoch 10/50
422/422 [==============================] - 16s 39ms/step - loss:
2.2977 - accuracy: 0.1239 - val_loss: 2.2979 - val_accuracy: 0.1055
Epoch 11/50
422/422 [==============================] - 17s 40ms/step - loss:
2.2965 - accuracy: 0.1249 - val_loss: 2.2953 - val_accuracy: 0.1050
```

```
Epoch 12/50
422/422 [==============================] - 22s 53ms/step - loss:
2.2951 - accuracy: 0.1257 - val_loss: 2.2975 - val_accuracy: 0.1050
Epoch 13/50
422/422 [==============================] - 15s 34ms/step - loss:
2.2939 - accuracy: 0.1359 - val_loss: 2.2906 - val_accuracy: 0.1113
Epoch 14/50
422/422 [==============================] - 14s 34ms/step - loss:
2.2926 - accuracy: 0.1360 - val_loss: 2.2904 - val_accuracy: 0.1045
Epoch 15/50
422/422 [==============================] - 16s 39ms/step - loss:
2.2912 - accuracy: 0.1451 - val_loss: 2.2927 - val_accuracy: 0.1050
Epoch 16/50
422/422 [==============================] - 15s 35ms/step - loss:
2.2887 - accuracy: 0.1474 - val_loss: 2.2930 - val_accuracy: 0.1050
Epoch 17/50
422/422 [==============================] - 16s 37ms/step - loss:
2.2862 - accuracy: 0.1550 - val_loss: 2.2855 - val_accuracy: 0.1050
Epoch 18/50
422/422 [==============================] - 16s 37ms/step - loss:
2.2832 - accuracy: 0.1632 - val_loss: 2.2818 - val_accuracy: 0.1070
Epoch 19/50
422/422 [==============================] - 15s 36ms/step - loss:
2.2797 - accuracy: 0.1791 - val_loss: 2.2768 - val_accuracy: 0.1980
Epoch 20/50
422/422 [==============================] - 16s 38ms/step - loss:
2.2749 - accuracy: 0.1875 - val_loss: 2.2716 - val_accuracy: 0.2048
Epoch 21/50
422/422 [==============================] - 15s 36ms/step - loss:
2.2686 - accuracy: 0.1994 - val_loss: 2.2651 - val_accuracy: 0.1050
Epoch 22/50
422/422 [==============================] - 15s 35ms/step - loss:
2.2599 - accuracy: 0.2170 - val_loss: 2.2547 - val_accuracy: 0.2402
Epoch 23/50
422/422 [==============================] - 16s 39ms/step - loss:
2.2474 - accuracy: 0.2411 - val_loss: 2.2408 - val_accuracy: 0.2355
Epoch 24/50
422/422 [==============================] - 15s 35ms/step - loss:
2.2287 - accuracy: 0.2671 - val_loss: 2.2153 - val_accuracy: 0.2092
Epoch 25/50
422/422 [==============================] - 15s 35ms/step - loss:
2.1973 - accuracy: 0.2976 - val_loss: 2.1741 - val_accuracy: 0.4073
Epoch 26/50
422/422 [==============================] - 15s 36ms/step - loss:
2.1429 - accuracy: 0.3309 - val_loss: 2.1025 - val_accuracy: 0.3227
Epoch 27/50
422/422 [==============================] - 15s 35ms/step - loss:
2.0481 - accuracy: 0.3567 - val_loss: 1.9826 - val_accuracy: 0.4713
Epoch 28/50
422/422 [==============================] - 16s 38ms/step - loss:
```

```
1.9105 - accuracy: 0.3944 - val_loss: 1.8298 - val_accuracy: 0.4688
Epoch 29/50
422/422 [==============================] - 17s 39ms/step - loss:
1.7504 - accuracy: 0.4278 - val_loss: 1.6567 - val_accuracy: 0.4972
Epoch 30/50
422/422 [==============================] - 15s 35ms/step - loss:
1.5953 - accuracy: 0.4636 - val_loss: 1.5063 - val_accuracy: 0.5215
Epoch 31/50
422/422 [==============================] - 15s 36ms/step - loss:
1.4645 - accuracy: 0.4991 - val_loss: 1.3812 - val_accuracy: 0.5608
Epoch 32/50
422/422 [==============================] - 15s 35ms/step - loss:
1.3474 - accuracy: 0.5416 - val_loss: 1.2626 - val_accuracy: 0.5895
Epoch 33/50
422/422 [==============================] - 15s 36ms/step - loss:
1.2387 - accuracy: 0.5878 - val_loss: 1.1584 - val_accuracy: 0.6278
Epoch 34/50
422/422 [==============================] - 16s 37ms/step - loss:
1.1373 - accuracy: 0.6313 - val_loss: 1.0523 - val_accuracy: 0.6862
Epoch 35/50
422/422 [==============================] - 15s 36ms/step - loss:
1.0418 - accuracy: 0.6732 - val_loss: 0.9577 - val_accuracy: 0.7227
Epoch 36/50
422/422 [==============================] - 15s 35ms/step - loss:
0.9561 - accuracy: 0.7127 - val_loss: 0.8729 - val_accuracy: 0.7522
Epoch 37/50
422/422 [==============================] - 15s 36ms/step - loss:
0.8794 - accuracy: 0.7453 - val_loss: 0.8006 - val_accuracy: 0.7822
Epoch 38/50
422/422 [==============================] - 15s 35ms/step - loss:
0.8139 - accuracy: 0.7694 - val_loss: 0.7412 - val_accuracy: 0.8098
Epoch 39/50
422/422 [==============================] - 15s 35ms/step - loss:
0.7557 - accuracy: 0.7925 - val_loss: 0.6878 - val_accuracy: 0.8277
Epoch 40/50
422/422 [==============================] - 16s 37ms/step - loss:
0.7047 - accuracy: 0.8109 - val_loss: 0.6434 - val_accuracy: 0.8378
Epoch 41/50
422/422 [==============================] - 15s 37ms/step - loss:
0.6587 - accuracy: 0.8278 - val_loss: 0.6040 - val_accuracy: 0.8525
Epoch 42/50
422/422 [==============================] - 15s 35ms/step - loss:
0.6181 - accuracy: 0.8427 - val_loss: 0.5678 - val_accuracy: 0.8628
Epoch 43/50
422/422 [==============================] - 15s 34ms/step - loss:
0.5802 - accuracy: 0.8538 - val_loss: 0.5365 - val_accuracy: 0.8725
Epoch 44/50
422/422 [==============================] - 16s 39ms/step - loss:
0.5462 - accuracy: 0.8651 - val_loss: 0.5111 - val_accuracy: 0.8802
Epoch 45/50
```

```
422/422 [==============================] - 15s 36ms/step - loss:
0.5161 - accuracy: 0.8734 - val_loss: 0.4851 - val_accuracy: 0.8887
Epoch 46/50
422/422 [==============================] - 15s 35ms/step - loss:
0.4888 - accuracy: 0.8805 - val_loss: 0.4653 - val_accuracy: 0.8887
Epoch 47/50
422/422 [==============================] - 16s 37ms/step - loss:
0.4658 - accuracy: 0.8879 - val_loss: 0.4451 - val_accuracy: 0.8975
Epoch 48/50
422/422 [==============================] - 15s 36ms/step - loss:
0.4428 - accuracy: 0.8949 - val_loss: 0.4284 - val_accuracy: 0.8990
Epoch 49/50
422/422 [==============================] - 14s 34ms/step - loss:
0.4221 - accuracy: 0.9000 - val_loss: 0.4167 - val_accuracy: 0.9030
Epoch 50/50
422/422 [==============================] - 15s 35ms/step - loss:
0.4044 - accuracy: 0.9047 - val_loss: 0.4009 - val_accuracy: 0.9063
```



```
Test loss: 0.439
Test accuracy: 0.895
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

#Discusión

Test loss: 0.191 Test accuracy: 0.945

Test loss: 0.201 Test accuracy: 0.942

Test loss: 0.217 Test accuracy: 0.939

Test loss: 0.255 Test accuracy: 0.932

Test loss: 0.422 Test accuracy: 0.898

5. Mas capas, más entrenamiento, Batch más pequeño (4 puntos)

A veces, los modelos con varias capas necesitan no solo entrenarse durante más tiempo, sino que también necesitan más "correcciones" por época. Al disminuir el tamaño del *batch*, podemos aumentar el número de "correcciones" que obtiene un modelo para mejorar su desempeño. También nos aseguramos de que obtenga información más detallada ajustando el error en un *batch* más pequeño.

En este caso, podemos forzar un modelo que no aprendió bien, como el modelo de la sección anterior con 5 capas ocultas de 32 nodos, para lograr una precisión moderadamente respetable. Aunque dicho rendimiento aún no sea excelente, vale la pena mencionar que con paciencia y potencia computacional podemos hacer que un modelo que parezca malo tenga un rendimiento decente.

Con este objetivo, Cree un modelo que tenga 5 capas de 32 nodos (i.e., similar al de la sección anterior que tuvo rendimiento pobre) y entrénelo durante 50 épocas pero esta vez con un tamaño de batch de 16 (batch_size=16).Discuta sus resultados y compare con el modelo de 5 capas ocultas de 32 nodos de la sección anterior.

Considere que este experimento puede demorar en ejecutar

```
for layers in [5]:
    print(layers*[32])
    model = create_dense([32] * layers)
    evaluate(model, batch_size=16, epochs=50, verbose=True)
```

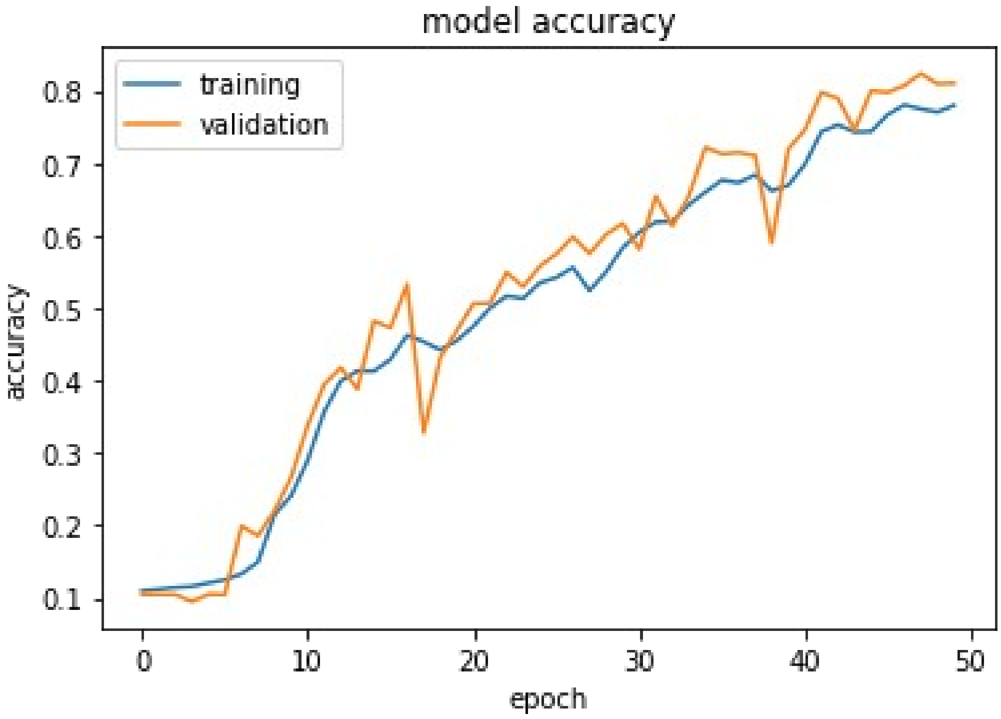
```
[32, 32, 32, 32, 32]
Model: "sequential_44"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_142 (Dense)           (None, 32)                25120     
                                                                 
 dense_143 (Dense)           (None, 32)                1056      
                                                                 
 dense_144 (Dense)           (None, 32)                1056      
                                                                 
 dense_145 (Dense)           (None, 32)                1056      
                                                                 
 dense_146 (Dense)           (None, 32)                1056      
                                                                 
 dense_147 (Dense)           (None, 10)                330       
```

```
=================================================================
Total params: 29,674
Trainable params: 29,674
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
3375/3375 [==============================] - 12s 3ms/step - loss:
2.3320 - accuracy: 0.1100 - val_loss: 2.3270 - val_accuracy: 0.1050
Epoch 2/50
3375/3375 [==============================] - 11s 3ms/step - loss:
2.3263 - accuracy: 0.1121 - val_loss: 2.3253 - val_accuracy: 0.1050
Epoch 3/50
3375/3375 [==============================] - 9s 3ms/step - loss:
2.3251 - accuracy: 0.1146 - val_loss: 2.3252 - val_accuracy: 0.1050
Epoch 4/50
3375/3375 [==============================] - 9s 3ms/step - loss:
2.3240 - accuracy: 0.1160 - val_loss: 2.3236 - val_accuracy: 0.0952
Epoch 5/50
3375/3375 [==============================] - 9s 3ms/step - loss:
2.3226 - accuracy: 0.1208 - val_loss: 2.3221 - val_accuracy: 0.1050
Epoch 6/50
3375/3375 [==============================] - 12s 4ms/step - loss:
2.3209 - accuracy: 0.1253 - val_loss: 2.3203 - val_accuracy: 0.1050
Epoch 7/50
3375/3375 [==============================] - 10s 3ms/step - loss:
2.3170 - accuracy: 0.1332 - val_loss: 2.3164 - val_accuracy: 0.1992
Epoch 8/50
3375/3375 [==============================] - 9s 3ms/step - loss:
2.3062 - accuracy: 0.1499 - val_loss: 2.2930 - val_accuracy: 0.1860
Epoch 9/50
3375/3375 [==============================] - 12s 4ms/step - loss:
2.2199 - accuracy: 0.2154 - val_loss: 2.0578 - val_accuracy: 0.2208
Epoch 10/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.9148 - accuracy: 0.2412 - val_loss: 1.8412 - val_accuracy: 0.2665
Epoch 11/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.8050 - accuracy: 0.2903 - val_loss: 1.7340 - val_accuracy: 0.3383
Epoch 12/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.6706 - accuracy: 0.3573 - val_loss: 1.5644 - val_accuracy: 0.3950
Epoch 13/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.5947 - accuracy: 0.3994 - val_loss: 1.5356 - val_accuracy: 0.4177
Epoch 14/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.5861 - accuracy: 0.4132 - val_loss: 1.5869 - val_accuracy: 0.3883
Epoch 15/50
3375/3375 [==============================] - 12s 3ms/step - loss:
1.5938 - accuracy: 0.4129 - val_loss: 1.4891 - val_accuracy: 0.4820
```

```
Epoch 16/50
3375/3375 [==============================] - 14s 4ms/step - loss:
1.5817 - accuracy: 0.4293 - val_loss: 1.4413 - val_accuracy: 0.4727
Epoch 17/50
3375/3375 [==============================] - 14s 4ms/step - loss:
1.5060 - accuracy: 0.4625 - val_loss: 1.4381 - val_accuracy: 0.5330
Epoch 18/50
3375/3375 [==============================] - 15s 5ms/step - loss:
1.5390 - accuracy: 0.4535 - val_loss: 1.6138 - val_accuracy: 0.3277
Epoch 19/50
3375/3375 [==============================] - 12s 4ms/step - loss:
1.5096 - accuracy: 0.4422 - val_loss: 1.4517 - val_accuracy: 0.4317
Epoch 20/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.5228 - accuracy: 0.4549 - val_loss: 1.4493 - val_accuracy: 0.4690
Epoch 21/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.4769 - accuracy: 0.4749 - val_loss: 1.3822 - val_accuracy: 0.5060
Epoch 22/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.4418 - accuracy: 0.4999 - val_loss: 1.3527 - val_accuracy: 0.5067
Epoch 23/50
3375/3375 [==============================] - 8s 3ms/step - loss:
1.4127 - accuracy: 0.5165 - val_loss: 1.3509 - val_accuracy: 0.5490
Epoch 24/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.4259 - accuracy: 0.5134 - val_loss: 1.3104 - val_accuracy: 0.5295
Epoch 25/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.3755 - accuracy: 0.5347 - val_loss: 1.3255 - val_accuracy: 0.5575
Epoch 26/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.3529 - accuracy: 0.5421 - val_loss: 1.2586 - val_accuracy: 0.5745
Epoch 27/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.3204 - accuracy: 0.5560 - val_loss: 1.2197 - val_accuracy: 0.5980
Epoch 28/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.3722 - accuracy: 0.5240 - val_loss: 1.2304 - val_accuracy: 0.5750
Epoch 29/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.3242 - accuracy: 0.5498 - val_loss: 1.1991 - val_accuracy: 0.6012
Epoch 30/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.2486 - accuracy: 0.5827 - val_loss: 1.1644 - val_accuracy: 0.6167
Epoch 31/50
3375/3375 [==============================] - 13s 4ms/step - loss:
1.2292 - accuracy: 0.6044 - val_loss: 1.1998 - val_accuracy: 0.5808
Epoch 32/50
3375/3375 [==============================] - 8s 2ms/step - loss:
```

```
1.2054 - accuracy: 0.6184 - val_loss: 1.0917 - val_accuracy: 0.6538
Epoch 33/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.1886 - accuracy: 0.6194 - val_loss: 1.1250 - val_accuracy: 0.6130
Epoch 34/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.1532 - accuracy: 0.6424 - val_loss: 1.1310 - val_accuracy: 0.6547
Epoch 35/50
3375/3375 [==============================] - 11s 3ms/step - loss:
1.1415 - accuracy: 0.6598 - val_loss: 1.0106 - val_accuracy: 0.7213
Epoch 36/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.1025 - accuracy: 0.6763 - val_loss: 1.0140 - val_accuracy: 0.7125
Epoch 37/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.1055 - accuracy: 0.6727 - val_loss: 1.0301 - val_accuracy: 0.7143
Epoch 38/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.1005 - accuracy: 0.6836 - val_loss: 1.0128 - val_accuracy: 0.7103
Epoch 39/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.1321 - accuracy: 0.6616 - val_loss: 1.2771 - val_accuracy: 0.5893
Epoch 40/50
3375/3375 [==============================] - 12s 3ms/step - loss:
1.1231 - accuracy: 0.6690 - val_loss: 1.0468 - val_accuracy: 0.7190
Epoch 41/50
3375/3375 [==============================] - 9s 3ms/step - loss:
1.0894 - accuracy: 0.6979 - val_loss: 0.9517 - val_accuracy: 0.7457
Epoch 42/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.0323 - accuracy: 0.7432 - val_loss: 0.8804 - val_accuracy: 0.7973
Epoch 43/50
3375/3375 [==============================] - 10s 3ms/step - loss:
1.0009 - accuracy: 0.7525 - val_loss: 0.8986 - val_accuracy: 0.7890
Epoch 44/50
3375/3375 [==============================] - 12s 3ms/step - loss:
1.0060 - accuracy: 0.7428 - val_loss: 0.9417 - val_accuracy: 0.7452
Epoch 45/50
3375/3375 [==============================] - 12s 4ms/step - loss:
0.9874 - accuracy: 0.7433 - val_loss: 0.8523 - val_accuracy: 0.7995
Epoch 46/50
3375/3375 [==============================] - 11s 3ms/step - loss:
0.9390 - accuracy: 0.7666 - val_loss: 0.8404 - val_accuracy: 0.7975
Epoch 47/50
3375/3375 [==============================] - 9s 3ms/step - loss:
0.9214 - accuracy: 0.7801 - val_loss: 0.8391 - val_accuracy: 0.8067
Epoch 48/50
3375/3375 [==============================] - 10s 3ms/step - loss:
0.9254 - accuracy: 0.7746 - val_loss: 0.7995 - val_accuracy: 0.8232
Epoch 49/50
```

```
3375/3375 [==============================] - 10s 3ms/step - loss:
0.9442 - accuracy: 0.7702 - val_loss: 0.8354 - val_accuracy: 0.8092
Epoch 50/50
3375/3375 [==============================] - 10s 3ms/step - loss:
0.9211 - accuracy: 0.7794 - val_loss: 0.8023 - val_accuracy: 0.8100
```



```
Test loss: 0.873
Test accuracy: 0.794
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.    ]
```

#Discusión

Test loss: 0.873 Test accuracy: 0.794

6. Regularización en Redes Neuronales (4 puntos)

Como se estudió en clase, la regularización permite obtener modelos que pueden generalizar de manera más precisa en un conjunto de test.

1. Investigue cómo añadir un término de regularización igual al estudiado en clase.
2. Modifique la función `create_dense`para que todas sus capas (i.e., inclusive la capa de salida) incluyan este término de regularización. Observe que Keras requiere que se especifique en cada capa dicho término. Fije el valor del parámetro de regularización en 1e-4.

3. Repita la sección 2, 3 y 4.2 usando el término de regularización

Discuta sus resultados

Tip: al definir el modelo use el parámetro `kernel_regularizer` (variable lambda en la teoría). Ver ayuda en aquí

PARTE 1. Solo se necesita conocer estas capas que exponen 3 argumentos de palabras clave:

kernel_regularizer: Regularizador para aplicar una penalización en el núcleo de la capa

bias_regularizer: Regularizador para aplicar una penalización en el sesgo de la capa

activity_regularizer: Regularizador para aplicar una penalización en la salida de la capa

```
#PARTE 2 Modifique la función create_dense para que todas sus capas
(i.e., inclusive la capa de salida) incluyan este término de
regularización.
#Observe que Keras requiere que se especifique en cada capa dicho
término. Fije el valor del parámetro de regularización en 1e-4.
def create_dense(layer_sizes):
    model = Sequential()
    from tensorflow.keras import regularizers
    kernel_regularizer=regularizers.L2(1e-4)
    model.add(Dense(layer_sizes[0], activation='sigmoid',
input_shape=(image_size,),kernel_regularizer=regularizers.L2(1e-4),
                        bias_regularizer=regularizers.L2(1e-4),
                        activity_regularizer=regularizers.L2(1e-
4)))#aqui añadir kernel_regularizer

    for s in layer_sizes[1:]:
        model.add(Dense(units = s, activation = 'sigmoid',
kernel_regularizer=regularizers.L2(1e-4),
                        bias_regularizer=regularizers.L2(1e-4),
                        activity_regularizer=regularizers.L2(1e-4)))
#aqui añadir kernel_regularizer

    model.add(Dense(units=num_classes, activation='softmax',
kernel_regularizer=regularizers.L2(1e-4),
                        bias_regularizer=regularizers.L2(1e-4),
                        activity_regularizer=regularizers.L2(1e-4)))
    plot_model(model, to_file='model_plot.png', show_shapes=True,
show_layer_names=True)
    return model

#PARTE 3 ESCRIBA SU CÓDIGO AQUÍ.
#2
model = create_dense([32] * 3) #Aquí ya se aplica la regularización
por la definicion de create_dense
```

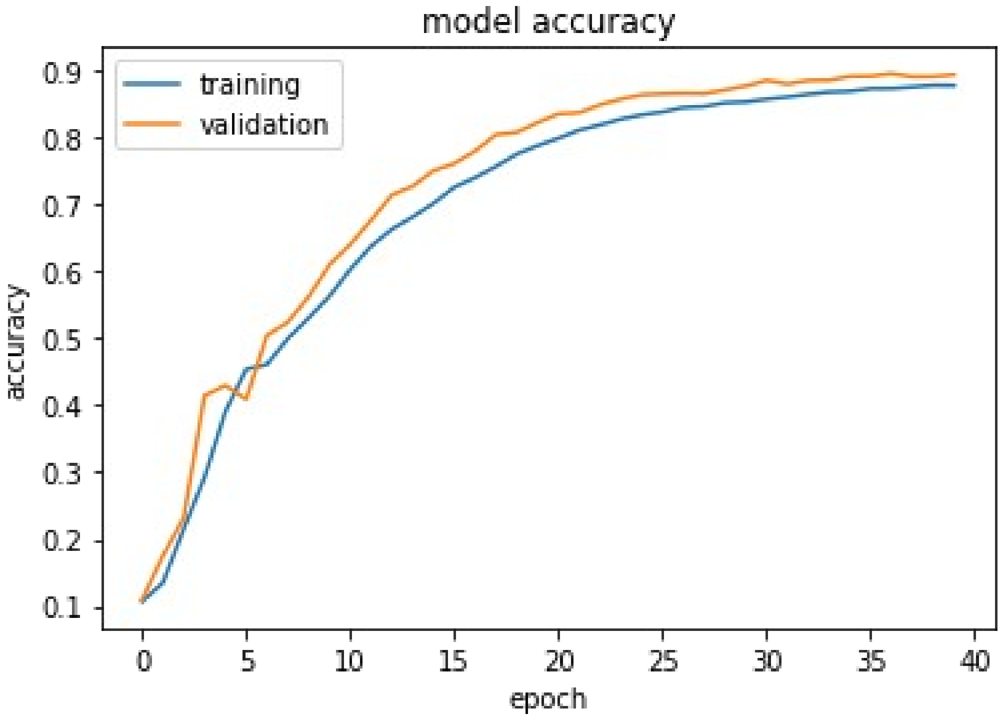
```
evaluate(model, batch_size=128, epochs=40, verbose=True) #verbose por
defecto es false

#3
for nodes in [32, 64, 128, 256, 512, 1024, 2048]:
    #print(i)
    model = create_dense([nodes] * 1) #Aquí ya se aplica la
regularización por la definicion de create_dense
    evaluate(model, batch_size=128, epochs=10, verbose=True) #verbose
por defecto es false

#4.2
for layers in [1, 2, 3, 4, 5]:
    model = create_dense([128] * layers) #Aquí ya se aplica la
regularización por la definicion de create_dense
    evaluate(model, batch_size=128, epochs=10*layers, verbose=True)
#verbose por defecto es false
```

```
Model: "sequential_31"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_104 (Dense)           (None, 32)                25120     
                                                                 
 dense_105 (Dense)           (None, 32)                1056      
                                                                 
 dense_106 (Dense)           (None, 32)                1056      
                                                                 
 dense_107 (Dense)           (None, 10)                330       
                                                                 
=================================================================
Total params: 27,562
Trainable params: 27,562
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 4s 7ms/step - loss: 2.3529
- accuracy: 0.1068 - val_loss: 2.3001 - val_accuracy: 0.1097
Epoch 2/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2883
- accuracy: 0.1348 - val_loss: 2.2782 - val_accuracy: 0.1750
Epoch 3/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2667
- accuracy: 0.2153 - val_loss: 2.2541 - val_accuracy: 0.2320
Epoch 4/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2411
- accuracy: 0.2923 - val_loss: 2.2244 - val_accuracy: 0.4142
Epoch 5/40
422/422 [==============================] - 2s 4ms/step - loss: 2.2082
- accuracy: 0.3902 - val_loss: 2.1860 - val_accuracy: 0.4293
```

```
Epoch 6/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1649
- accuracy: 0.4536 - val_loss: 2.1346 - val_accuracy: 0.4088
Epoch 7/40
422/422 [==============================] - 2s 4ms/step - loss: 2.1072
- accuracy: 0.4607 - val_loss: 2.0660 - val_accuracy: 0.5042
Epoch 8/40
422/422 [==============================] - 2s 6ms/step - loss: 2.0312
- accuracy: 0.4994 - val_loss: 1.9774 - val_accuracy: 0.5240
Epoch 9/40
422/422 [==============================] - 2s 4ms/step - loss: 1.9374
- accuracy: 0.5302 - val_loss: 1.8735 - val_accuracy: 0.5618
Epoch 10/40
422/422 [==============================] - 1s 3ms/step - loss: 1.8304
- accuracy: 0.5627 - val_loss: 1.7589 - val_accuracy: 0.6095
Epoch 11/40
422/422 [==============================] - 2s 4ms/step - loss: 1.7164
- accuracy: 0.6027 - val_loss: 1.6398 - val_accuracy: 0.6398
Epoch 12/40
422/422 [==============================] - 2s 4ms/step - loss: 1.6000
- accuracy: 0.6371 - val_loss: 1.5220 - val_accuracy: 0.6760
Epoch 13/40
422/422 [==============================] - 2s 4ms/step - loss: 1.4862
- accuracy: 0.6626 - val_loss: 1.4082 - val_accuracy: 0.7138
Epoch 14/40
422/422 [==============================] - 2s 4ms/step - loss: 1.3793
- accuracy: 0.6810 - val_loss: 1.3042 - val_accuracy: 0.7268
Epoch 15/40
422/422 [==============================] - 2s 5ms/step - loss: 1.2834
- accuracy: 0.7012 - val_loss: 1.2093 - val_accuracy: 0.7502
Epoch 16/40
422/422 [==============================] - 2s 5ms/step - loss: 1.1991
- accuracy: 0.7254 - val_loss: 1.1266 - val_accuracy: 0.7608
Epoch 17/40
422/422 [==============================] - 1s 4ms/step - loss: 1.1274
- accuracy: 0.7397 - val_loss: 1.0618 - val_accuracy: 0.7793
Epoch 18/40
422/422 [==============================] - 2s 4ms/step - loss: 1.0658
- accuracy: 0.7563 - val_loss: 1.0004 - val_accuracy: 0.8040
Epoch 19/40
422/422 [==============================] - 2s 4ms/step - loss: 1.0125
- accuracy: 0.7750 - val_loss: 0.9496 - val_accuracy: 0.8073
Epoch 20/40
422/422 [==============================] - 2s 4ms/step - loss: 0.9647
- accuracy: 0.7876 - val_loss: 0.9043 - val_accuracy: 0.8222
Epoch 21/40
422/422 [==============================] - 2s 4ms/step - loss: 0.9258
- accuracy: 0.7985 - val_loss: 0.8609 - val_accuracy: 0.8350
Epoch 22/40
422/422 [==============================] - 1s 3ms/step - loss: 0.8886
```

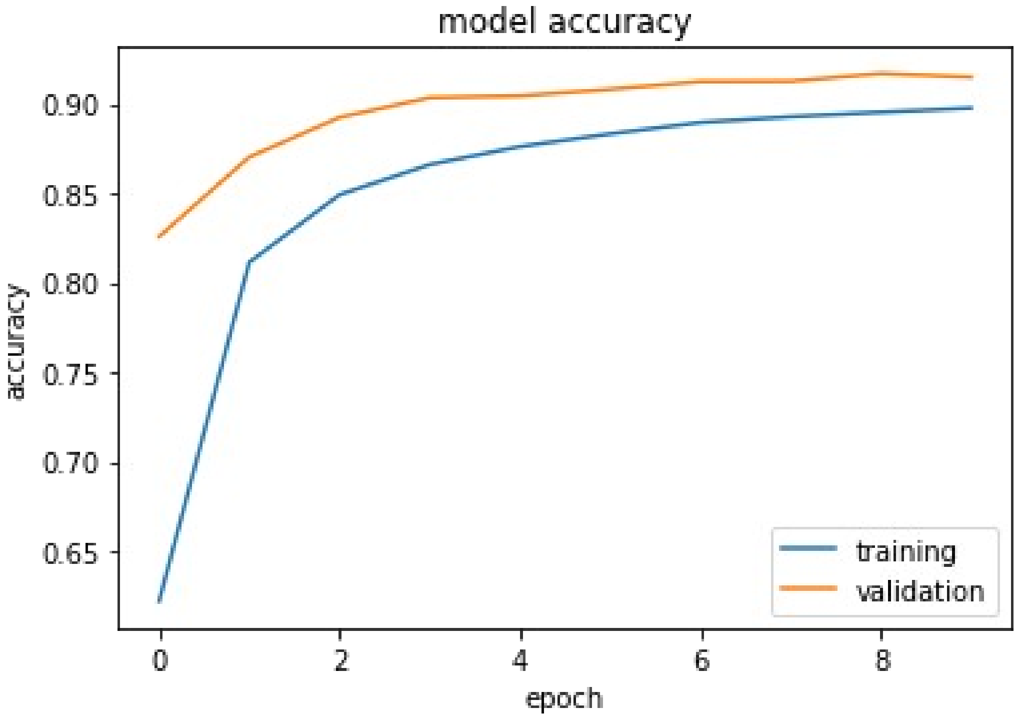
```
- accuracy: 0.8106 - val_loss: 0.8292 - val_accuracy: 0.8367
Epoch 23/40
422/422 [==============================] - 3s 7ms/step - loss: 0.8560
- accuracy: 0.8186 - val_loss: 0.7957 - val_accuracy: 0.8485
Epoch 24/40
422/422 [==============================] - 2s 4ms/step - loss: 0.8243
- accuracy: 0.8271 - val_loss: 0.7597 - val_accuracy: 0.8572
Epoch 25/40
422/422 [==============================] - 2s 4ms/step - loss: 0.7967
- accuracy: 0.8334 - val_loss: 0.7292 - val_accuracy: 0.8635
Epoch 26/40
422/422 [==============================] - 2s 4ms/step - loss: 0.7706
- accuracy: 0.8379 - val_loss: 0.7078 - val_accuracy: 0.8648
Epoch 27/40
422/422 [==============================] - 2s 4ms/step - loss: 0.7443
- accuracy: 0.8445 - val_loss: 0.6806 - val_accuracy: 0.8657
Epoch 28/40
422/422 [==============================] - 2s 4ms/step - loss: 0.7255
- accuracy: 0.8460 - val_loss: 0.6677 - val_accuracy: 0.8652
Epoch 29/40
422/422 [==============================] - 2s 4ms/step - loss: 0.7041
- accuracy: 0.8516 - val_loss: 0.6451 - val_accuracy: 0.8707
Epoch 30/40
422/422 [==============================] - 2s 6ms/step - loss: 0.6838
- accuracy: 0.8528 - val_loss: 0.6276 - val_accuracy: 0.8767
Epoch 31/40
422/422 [==============================] - 2s 5ms/step - loss: 0.6679
- accuracy: 0.8570 - val_loss: 0.6006 - val_accuracy: 0.8847
Epoch 32/40
422/422 [==============================] - 1s 3ms/step - loss: 0.6529
- accuracy: 0.8602 - val_loss: 0.5979 - val_accuracy: 0.8797
Epoch 33/40
422/422 [==============================] - 1s 3ms/step - loss: 0.6334
- accuracy: 0.8642 - val_loss: 0.5793 - val_accuracy: 0.8845
Epoch 34/40
422/422 [==============================] - 1s 3ms/step - loss: 0.6193
- accuracy: 0.8676 - val_loss: 0.5656 - val_accuracy: 0.8850
Epoch 35/40
422/422 [==============================] - 1s 4ms/step - loss: 0.6070
- accuracy: 0.8688 - val_loss: 0.5466 - val_accuracy: 0.8908
Epoch 36/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5922
- accuracy: 0.8727 - val_loss: 0.5383 - val_accuracy: 0.8910
Epoch 37/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5865
- accuracy: 0.8729 - val_loss: 0.5330 - val_accuracy: 0.8950
Epoch 38/40
422/422 [==============================] - 2s 6ms/step - loss: 0.5724
- accuracy: 0.8751 - val_loss: 0.5276 - val_accuracy: 0.8902
Epoch 39/40
```

```
422/422 [==============================] - 1s 3ms/step - loss: 0.5643
- accuracy: 0.8776 - val_loss: 0.5128 - val_accuracy: 0.8905
Epoch 40/40
422/422 [==============================] - 2s 4ms/step - loss: 0.5572
- accuracy: 0.8774 - val_loss: 0.5045 - val_accuracy: 0.8933
```

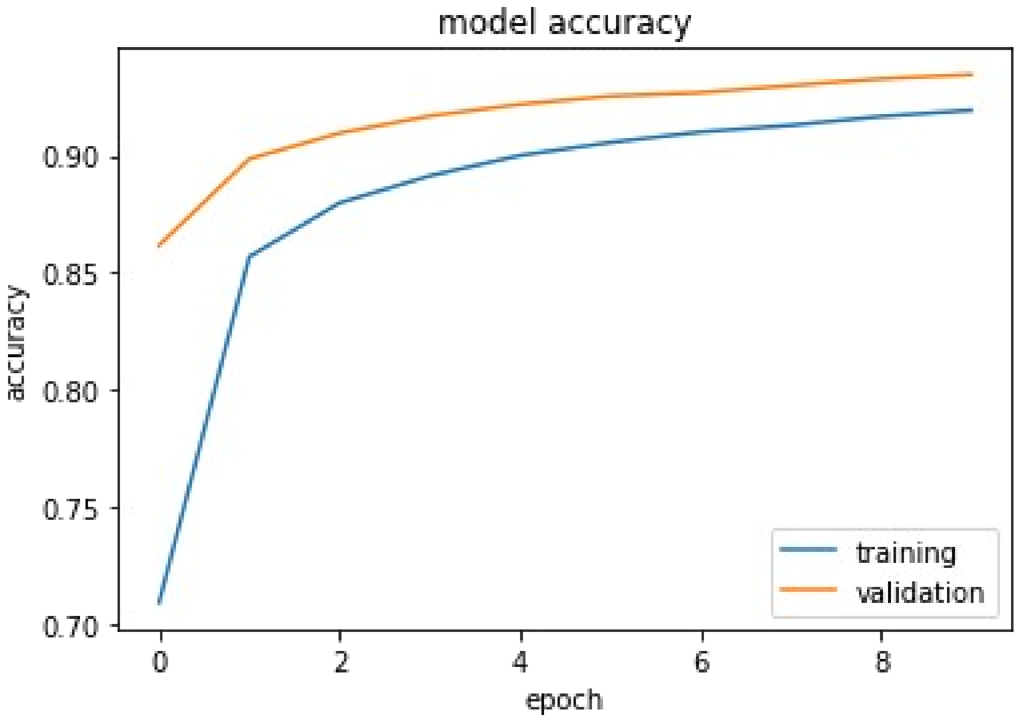


```
Test loss: 0.567
Test accuracy: 0.87
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_32"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_108 (Dense)           (None, 32)                25120

 dense_109 (Dense)           (None, 10)                330

=================================================================
Total params: 25,450
Trainable params: 25,450
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 5ms/step - loss: 1.4904
- accuracy: 0.6220 - val_loss: 1.0338 - val_accuracy: 0.8257
```

```
Epoch 2/10
422/422 [==============================] - 2s 5ms/step - loss: 0.9417
- accuracy: 0.8113 - val_loss: 0.7486 - val_accuracy: 0.8702
Epoch 3/10
422/422 [==============================] - 3s 7ms/step - loss: 0.7302
- accuracy: 0.8491 - val_loss: 0.5916 - val_accuracy: 0.8923
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 0.6172
- accuracy: 0.8660 - val_loss: 0.5086 - val_accuracy: 0.9033
Epoch 5/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5527
- accuracy: 0.8760 - val_loss: 0.4627 - val_accuracy: 0.9043
Epoch 6/10
422/422 [==============================] - 2s 4ms/step - loss: 0.5055
- accuracy: 0.8831 - val_loss: 0.4363 - val_accuracy: 0.9080
Epoch 7/10
422/422 [==============================] - 2s 4ms/step - loss: 0.4702
- accuracy: 0.8895 - val_loss: 0.3984 - val_accuracy: 0.9123
Epoch 8/10
422/422 [==============================] - 2s 4ms/step - loss: 0.4461
- accuracy: 0.8927 - val_loss: 0.3896 - val_accuracy: 0.9125
Epoch 9/10
422/422 [==============================] - 2s 4ms/step - loss: 0.4284
- accuracy: 0.8952 - val_loss: 0.3671 - val_accuracy: 0.9168
Epoch 10/10
422/422 [==============================] - 3s 6ms/step - loss: 0.4119
- accuracy: 0.8975 - val_loss: 0.3665 - val_accuracy: 0.9148
```

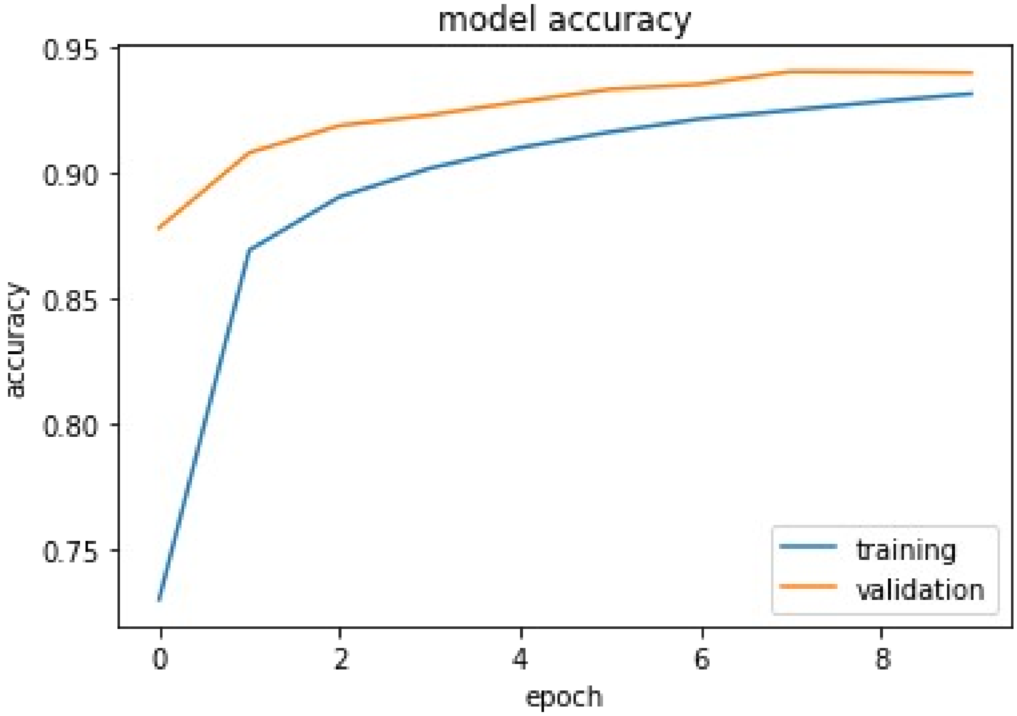


```
Test loss: 0.399
Test accuracy: 0.899
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_33"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_110 (Dense)           (None, 64)                50240     
                                                                 
 dense_111 (Dense)           (None, 10)                650       
                                                                 
=================================================================
Total params: 50,890
Trainable params: 50,890
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 5ms/step - loss: 1.1577
- accuracy: 0.7090 - val_loss: 0.7049 - val_accuracy: 0.8617
Epoch 2/10
422/422 [==============================] - 2s 5ms/step - loss: 0.6578
- accuracy: 0.8568 - val_loss: 0.5070 - val_accuracy: 0.8987
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 0.5283
- accuracy: 0.8798 - val_loss: 0.4239 - val_accuracy: 0.9097
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4597
- accuracy: 0.8914 - val_loss: 0.3762 - val_accuracy: 0.9170
Epoch 5/10
422/422 [==============================] - 3s 6ms/step - loss: 0.4180
- accuracy: 0.9001 - val_loss: 0.3513 - val_accuracy: 0.9220
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3881
- accuracy: 0.9057 - val_loss: 0.3236 - val_accuracy: 0.9255
Epoch 7/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3656
- accuracy: 0.9102 - val_loss: 0.3080 - val_accuracy: 0.9270
Epoch 8/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3502
- accuracy: 0.9130 - val_loss: 0.2945 - val_accuracy: 0.9300
Epoch 9/10
422/422 [==============================] - 2s 4ms/step - loss: 0.3346
- accuracy: 0.9168 - val_loss: 0.2872 - val_accuracy: 0.9328
Epoch 10/10
422/422 [==============================] - 2s 6ms/step - loss: 0.3215
- accuracy: 0.9194 - val_loss: 0.2753 - val_accuracy: 0.9347
```



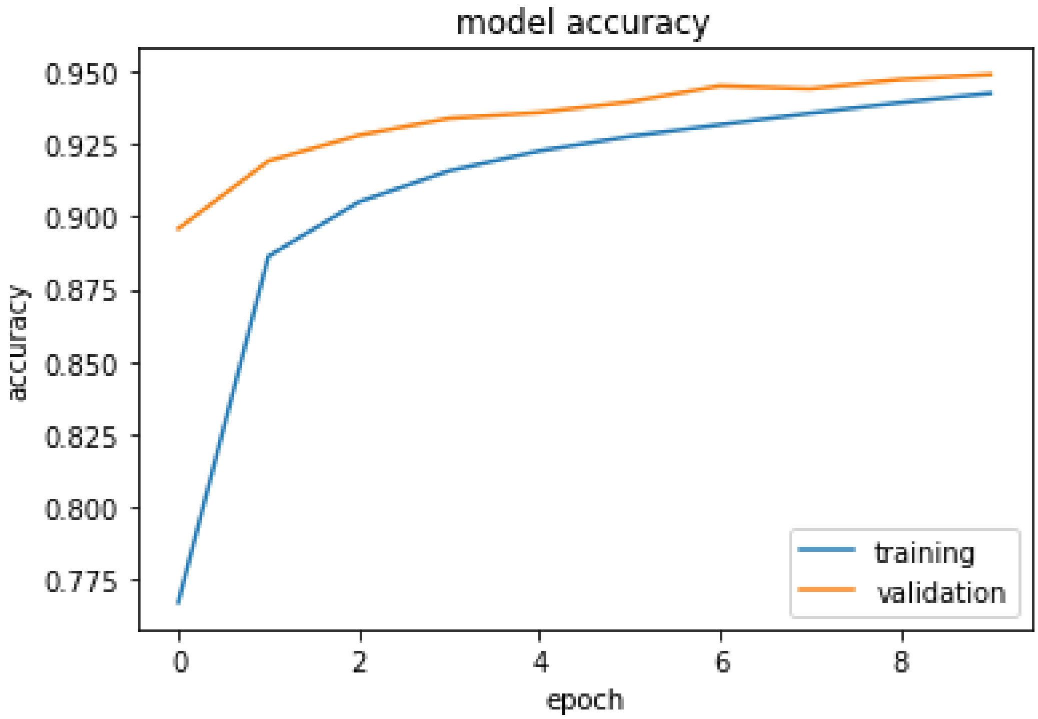
```
Test loss: 0.309
Test accuracy: 0.925
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_34"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_112 (Dense)           (None, 128)               100480

 dense_113 (Dense)           (None, 10)                1290

=================================================================
Total params: 101,770
Trainable params: 101,770
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 5ms/step - loss: 1.0705
- accuracy: 0.7300 - val_loss: 0.5904 - val_accuracy: 0.8782
Epoch 2/10
422/422 [==============================] - 2s 5ms/step - loss: 0.5679
- accuracy: 0.8693 - val_loss: 0.4328 - val_accuracy: 0.9082
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4598
- accuracy: 0.8906 - val_loss: 0.3708 - val_accuracy: 0.9190
```

```
Epoch 4/10
422/422 [==============================] - 3s 6ms/step - loss: 0.4059
- accuracy: 0.9019 - val_loss: 0.3361 - val_accuracy: 0.9232
Epoch 5/10
422/422 [==============================] - 3s 7ms/step - loss: 0.3711
- accuracy: 0.9102 - val_loss: 0.3092 - val_accuracy: 0.9285
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3451
- accuracy: 0.9164 - val_loss: 0.2912 - val_accuracy: 0.9335
Epoch 7/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3261
- accuracy: 0.9217 - val_loss: 0.2784 - val_accuracy: 0.9355
Epoch 8/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3116
- accuracy: 0.9251 - val_loss: 0.2633 - val_accuracy: 0.9405
Epoch 9/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2964
- accuracy: 0.9286 - val_loss: 0.2589 - val_accuracy: 0.9403
Epoch 10/10
422/422 [==============================] - 3s 8ms/step - loss: 0.2830
- accuracy: 0.9317 - val_loss: 0.2532 - val_accuracy: 0.9400
```



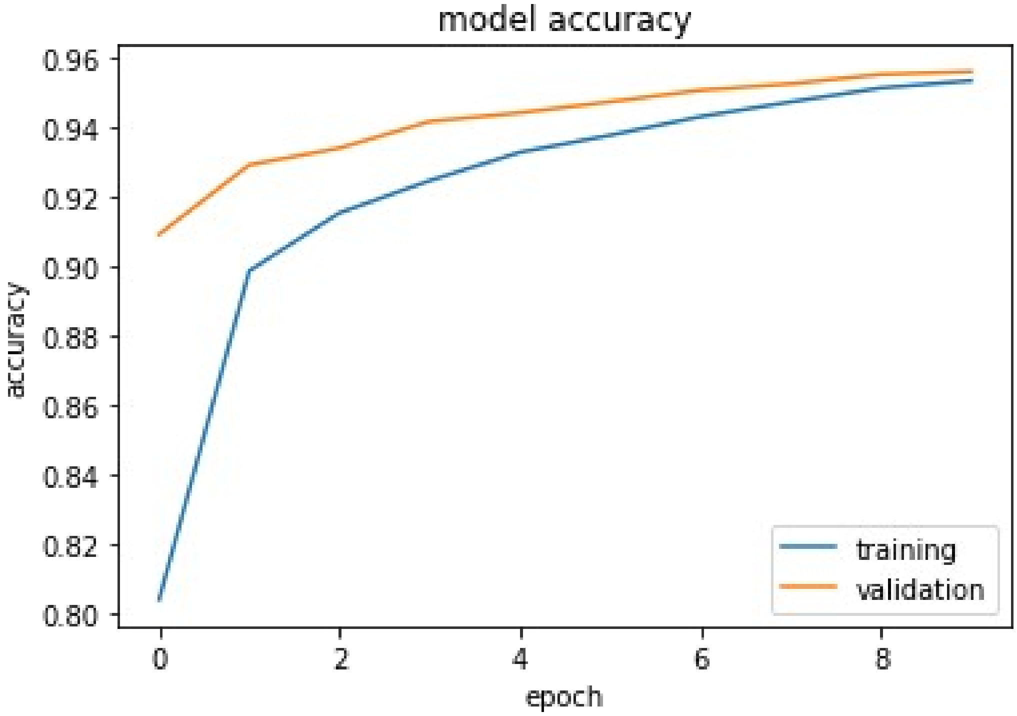
```
Test loss: 0.288
Test accuracy: 0.928
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

```
Model: "sequential_35"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_114 (Dense)           (None, 256)               200960

 dense_115 (Dense)           (None, 10)                2570

=================================================================
Total params: 203,530
Trainable params: 203,530
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 4s 8ms/step - loss: 0.9430
- accuracy: 0.7673 - val_loss: 0.5196 - val_accuracy: 0.8958
Epoch 2/10
422/422 [==============================] - 4s 9ms/step - loss: 0.5056
- accuracy: 0.8864 - val_loss: 0.3951 - val_accuracy: 0.9192
Epoch 3/10
422/422 [==============================] - 3s 7ms/step - loss: 0.4189
- accuracy: 0.9050 - val_loss: 0.3436 - val_accuracy: 0.9280
Epoch 4/10
422/422 [==============================] - 3s 7ms/step - loss: 0.3740
- accuracy: 0.9157 - val_loss: 0.3139 - val_accuracy: 0.9338
Epoch 5/10
422/422 [==============================] - 3s 7ms/step - loss: 0.3432
- accuracy: 0.9226 - val_loss: 0.2985 - val_accuracy: 0.9358
Epoch 6/10
422/422 [==============================] - 4s 9ms/step - loss: 0.3217
- accuracy: 0.9276 - val_loss: 0.2817 - val_accuracy: 0.9395
Epoch 7/10
422/422 [==============================] - 3s 6ms/step - loss: 0.3043
- accuracy: 0.9316 - val_loss: 0.2673 - val_accuracy: 0.9450
Epoch 8/10
422/422 [==============================] - 3s 7ms/step - loss: 0.2899
- accuracy: 0.9356 - val_loss: 0.2596 - val_accuracy: 0.9440
Epoch 9/10
422/422 [==============================] - 3s 7ms/step - loss: 0.2769
- accuracy: 0.9392 - val_loss: 0.2539 - val_accuracy: 0.9472
Epoch 10/10
422/422 [==============================] - 4s 9ms/step - loss: 0.2662
- accuracy: 0.9425 - val_loss: 0.2440 - val_accuracy: 0.9488
```



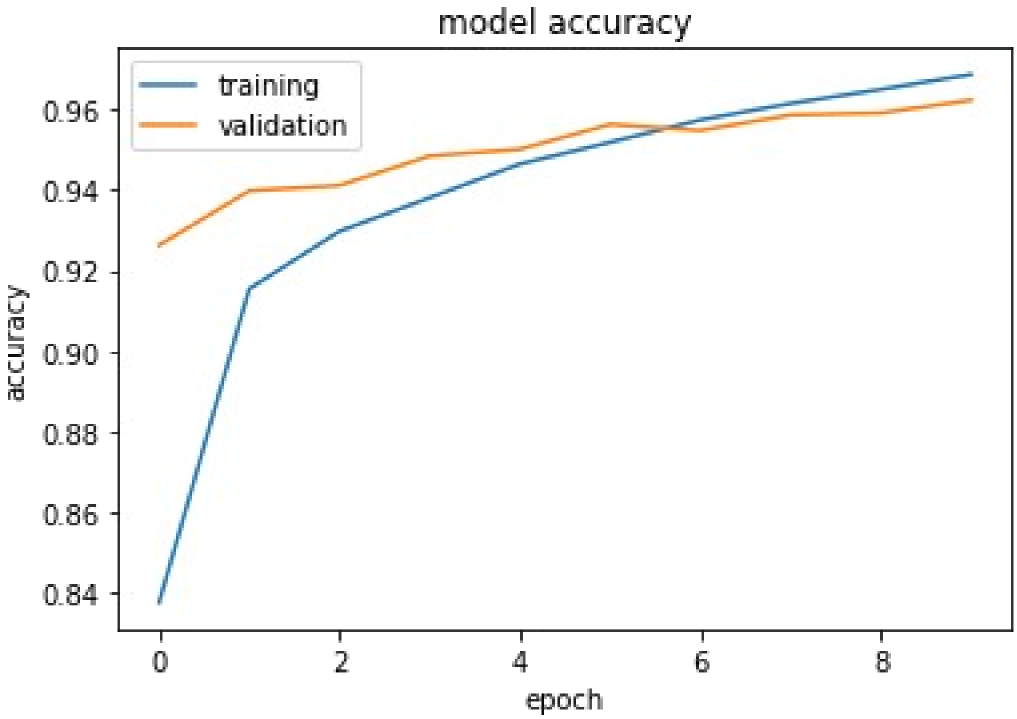
```
Test loss: 0.27
Test accuracy: 0.941
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_36"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_116 (Dense)           (None, 512)               401920

 dense_117 (Dense)           (None, 10)                5130

=================================================================
Total params: 407,050
Trainable params: 407,050
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 5s 10ms/step - loss: 0.8413
- accuracy: 0.8043 - val_loss: 0.4671 - val_accuracy: 0.9093
Epoch 2/10
422/422 [==============================] - 5s 11ms/step - loss: 0.4739
- accuracy: 0.8988 - val_loss: 0.3785 - val_accuracy: 0.9293
Epoch 3/10
422/422 [==============================] - 4s 11ms/step - loss: 0.4033
- accuracy: 0.9155 - val_loss: 0.3419 - val_accuracy: 0.9342
```

```
Epoch 4/10
422/422 [==============================] - 4s 9ms/step - loss: 0.3639
- accuracy: 0.9248 - val_loss: 0.3161 - val_accuracy: 0.9418
Epoch 5/10
422/422 [==============================] - 5s 12ms/step - loss: 0.3360
- accuracy: 0.9330 - val_loss: 0.2987 - val_accuracy: 0.9443
Epoch 6/10
422/422 [==============================] - 4s 11ms/step - loss: 0.3158
- accuracy: 0.9379 - val_loss: 0.2882 - val_accuracy: 0.9475
Epoch 7/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2980
- accuracy: 0.9432 - val_loss: 0.2737 - val_accuracy: 0.9508
Epoch 8/10
422/422 [==============================] - 5s 12ms/step - loss: 0.2829
- accuracy: 0.9475 - val_loss: 0.2661 - val_accuracy: 0.9527
Epoch 9/10
422/422 [==============================] - 4s 10ms/step - loss: 0.2714
- accuracy: 0.9515 - val_loss: 0.2616 - val_accuracy: 0.9553
Epoch 10/10
422/422 [==============================] - 4s 9ms/step - loss: 0.2615
- accuracy: 0.9536 - val_loss: 0.2564 - val_accuracy: 0.9562
```



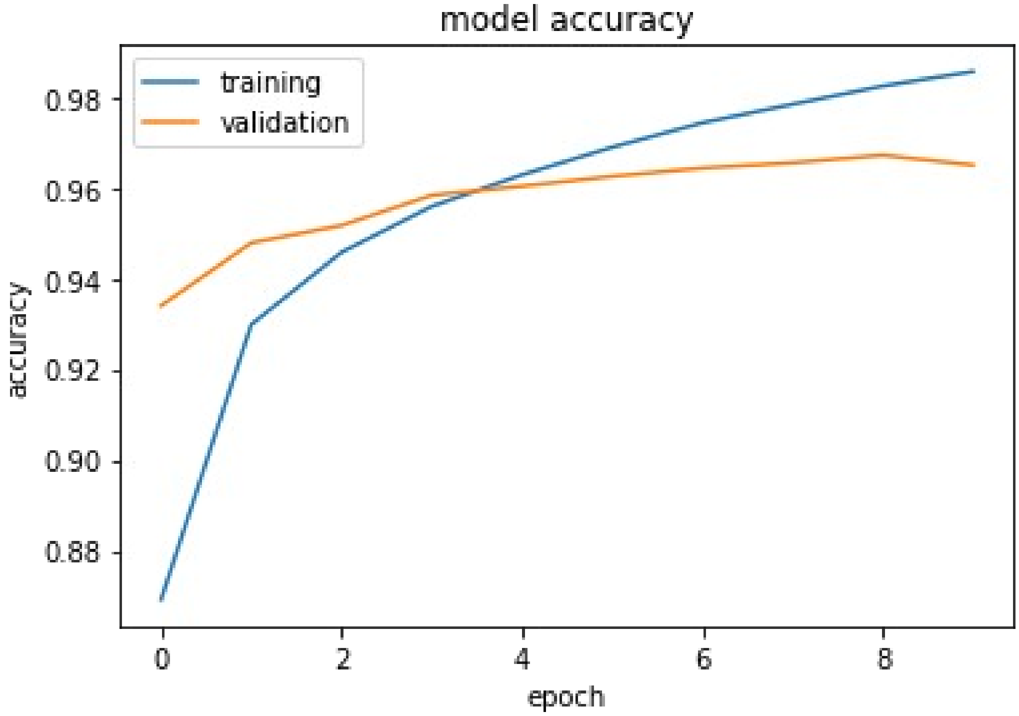
```
Test loss: 0.278
Test accuracy: 0.947
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

```
Model: "sequential_37"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_118 (Dense)           (None, 1024)              803840    
                                                                 
 dense_119 (Dense)           (None, 10)                10250     
                                                                 
=================================================================
Total params: 814,090
Trainable params: 814,090
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 9s 19ms/step - loss: 0.7586
- accuracy: 0.8376 - val_loss: 0.4485 - val_accuracy: 0.9263
Epoch 2/10
422/422 [==============================] - 8s 19ms/step - loss: 0.4578
- accuracy: 0.9155 - val_loss: 0.3819 - val_accuracy: 0.9398
Epoch 3/10
422/422 [==============================] - 7s 17ms/step - loss: 0.3998
- accuracy: 0.9298 - val_loss: 0.3505 - val_accuracy: 0.9412
Epoch 4/10
422/422 [==============================] - 8s 19ms/step - loss: 0.3657
- accuracy: 0.9381 - val_loss: 0.3300 - val_accuracy: 0.9485
Epoch 5/10
422/422 [==============================] - 7s 17ms/step - loss: 0.3392
- accuracy: 0.9465 - val_loss: 0.3172 - val_accuracy: 0.9502
Epoch 6/10
422/422 [==============================] - 8s 18ms/step - loss: 0.3193
- accuracy: 0.9519 - val_loss: 0.3059 - val_accuracy: 0.9563
Epoch 7/10
422/422 [==============================] - 8s 19ms/step - loss: 0.3024
- accuracy: 0.9574 - val_loss: 0.3002 - val_accuracy: 0.9548
Epoch 8/10
422/422 [==============================] - 8s 19ms/step - loss: 0.2879
- accuracy: 0.9615 - val_loss: 0.2922 - val_accuracy: 0.9587
Epoch 9/10
422/422 [==============================] - 8s 19ms/step - loss: 0.2753
- accuracy: 0.9650 - val_loss: 0.2865 - val_accuracy: 0.9592
Epoch 10/10
422/422 [==============================] - 7s 17ms/step - loss: 0.2648
- accuracy: 0.9687 - val_loss: 0.2818 - val_accuracy: 0.9623
```



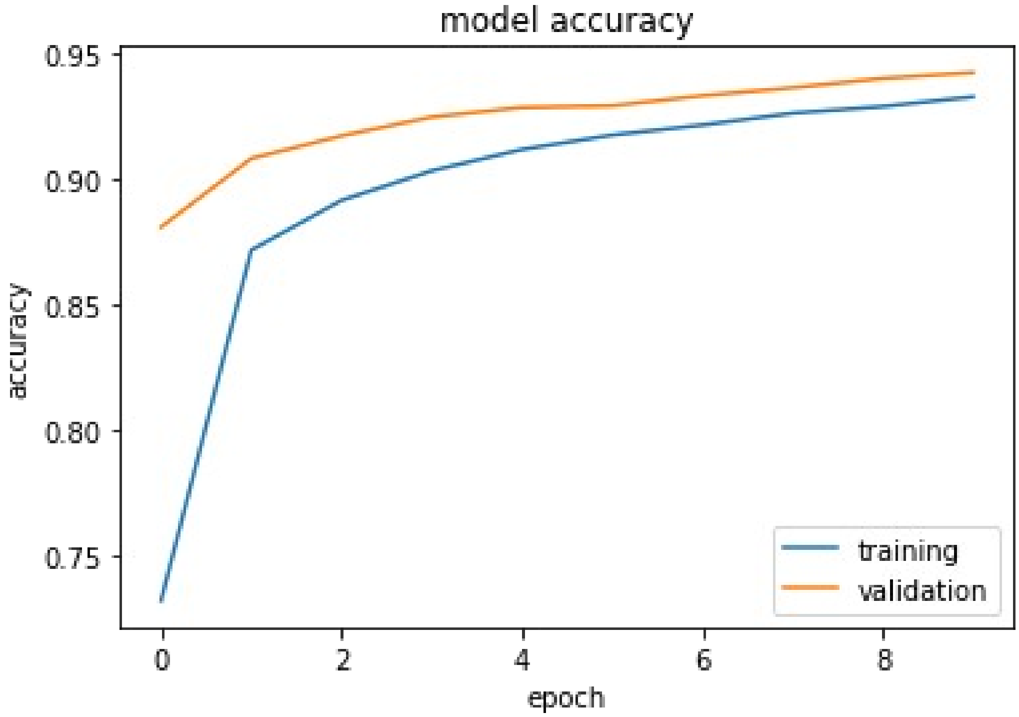
```
Test loss: 0.306
Test accuracy: 0.95
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_38"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_120 (Dense)           (None, 2048)              1607680

 dense_121 (Dense)           (None, 10)                20490

=================================================================
Total params: 1,628,170
Trainable params: 1,628,170
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 15s 33ms/step - loss:
0.7029 - accuracy: 0.8692 - val_loss: 0.4618 - val_accuracy: 0.9340
Epoch 2/10
422/422 [==============================] - 14s 33ms/step - loss:
0.4682 - accuracy: 0.9299 - val_loss: 0.4069 - val_accuracy: 0.9478
Epoch 3/10
422/422 [==============================] - 14s 33ms/step - loss:
0.4128 - accuracy: 0.9457 - val_loss: 0.3847 - val_accuracy: 0.9517
```

```
Epoch 4/10
422/422 [==============================] - 14s 34ms/step - loss:
0.3779 - accuracy: 0.9559 - val_loss: 0.3665 - val_accuracy: 0.9583
Epoch 5/10
422/422 [==============================] - 15s 35ms/step - loss:
0.3526 - accuracy: 0.9628 - val_loss: 0.3543 - val_accuracy: 0.9603
Epoch 6/10
422/422 [==============================] - 14s 33ms/step - loss:
0.3320 - accuracy: 0.9689 - val_loss: 0.3463 - val_accuracy: 0.9625
Epoch 7/10
422/422 [==============================] - 14s 33ms/step - loss:
0.3157 - accuracy: 0.9743 - val_loss: 0.3398 - val_accuracy: 0.9643
Epoch 8/10
422/422 [==============================] - 14s 33ms/step - loss:
0.3017 - accuracy: 0.9784 - val_loss: 0.3319 - val_accuracy: 0.9655
Epoch 9/10
422/422 [==============================] - 14s 33ms/step - loss:
0.2898 - accuracy: 0.9824 - val_loss: 0.3279 - val_accuracy: 0.9672
Epoch 10/10
422/422 [==============================] - 14s 33ms/step - loss:
0.2795 - accuracy: 0.9856 - val_loss: 0.3245 - val_accuracy: 0.9650
```



```
Test loss: 0.346
Test accuracy: 0.958
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

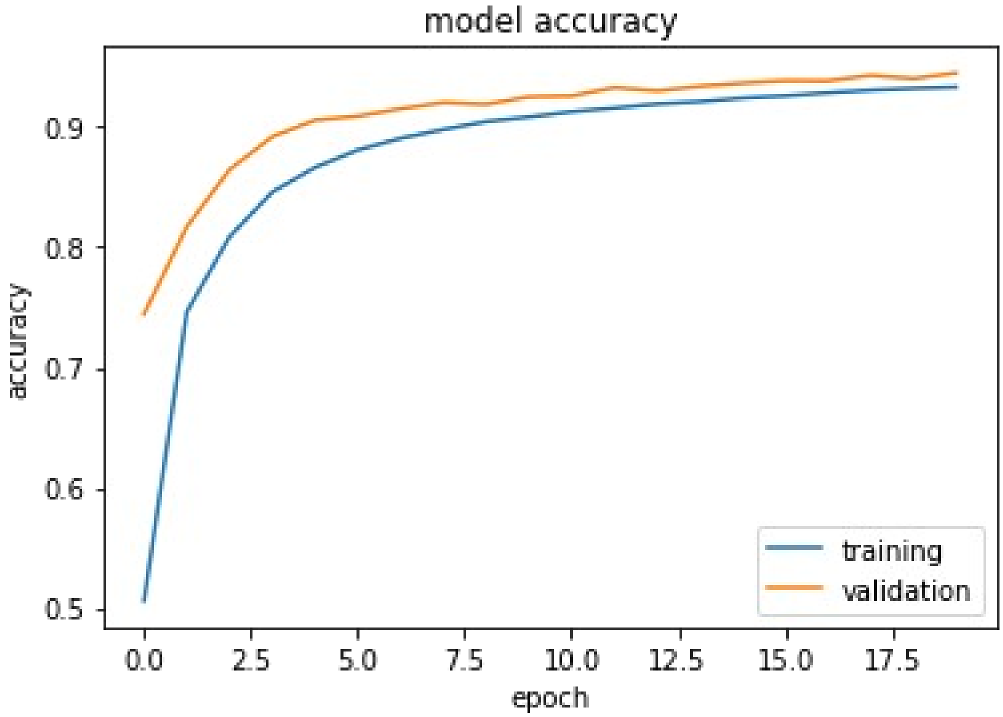
```
Model: "sequential_39"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_122 (Dense)           (None, 128)               100480

 dense_123 (Dense)           (None, 10)                1290

=================================================================
Total params: 101,770
Trainable params: 101,770
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
422/422 [==============================] - 3s 6ms/step - loss: 1.0652
- accuracy: 0.7320 - val_loss: 0.5862 - val_accuracy: 0.8807
Epoch 2/10
422/422 [==============================] - 3s 8ms/step - loss: 0.5635
- accuracy: 0.8715 - val_loss: 0.4266 - val_accuracy: 0.9080
Epoch 3/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4559
- accuracy: 0.8913 - val_loss: 0.3682 - val_accuracy: 0.9170
Epoch 4/10
422/422 [==============================] - 2s 5ms/step - loss: 0.4025
- accuracy: 0.9031 - val_loss: 0.3332 - val_accuracy: 0.9245
Epoch 5/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3672
- accuracy: 0.9116 - val_loss: 0.3077 - val_accuracy: 0.9283
Epoch 6/10
422/422 [==============================] - 2s 5ms/step - loss: 0.3422
- accuracy: 0.9173 - val_loss: 0.2977 - val_accuracy: 0.9290
Epoch 7/10
422/422 [==============================] - 3s 7ms/step - loss: 0.3253
- accuracy: 0.9213 - val_loss: 0.2761 - val_accuracy: 0.9330
Epoch 8/10
422/422 [==============================] - 3s 6ms/step - loss: 0.3085
- accuracy: 0.9260 - val_loss: 0.2652 - val_accuracy: 0.9362
Epoch 9/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2950
- accuracy: 0.9287 - val_loss: 0.2602 - val_accuracy: 0.9398
Epoch 10/10
422/422 [==============================] - 2s 5ms/step - loss: 0.2826
- accuracy: 0.9326 - val_loss: 0.2500 - val_accuracy: 0.9422
```



```
Test loss: 0.284
Test accuracy: 0.93
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_40"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_124 (Dense)           (None, 128)               100480

 dense_125 (Dense)           (None, 128)               16512

 dense_126 (Dense)           (None, 10)                1290

=================================================================
Total params: 118,282
Trainable params: 118,282
Non-trainable params: 0
_________________________________________________________________
Epoch 1/20
422/422 [==============================] - 4s 9ms/step - loss: 2.0601
- accuracy: 0.5061 - val_loss: 1.7351 - val_accuracy: 0.7447
Epoch 2/20
422/422 [==============================] - 3s 7ms/step - loss: 1.5083
- accuracy: 0.7463 - val_loss: 1.2363 - val_accuracy: 0.8170
Epoch 3/20
```

```
422/422 [==============================] - 3s 6ms/step - loss: 1.1144
- accuracy: 0.8086 - val_loss: 0.9190 - val_accuracy: 0.8645
Epoch 4/20
422/422 [==============================] - 2s 6ms/step - loss: 0.8689
- accuracy: 0.8459 - val_loss: 0.7229 - val_accuracy: 0.8920
Epoch 5/20
422/422 [==============================] - 3s 7ms/step - loss: 0.7142
- accuracy: 0.8663 - val_loss: 0.5973 - val_accuracy: 0.9057
Epoch 6/20
422/422 [==============================] - 3s 8ms/step - loss: 0.6118
- accuracy: 0.8808 - val_loss: 0.5168 - val_accuracy: 0.9090
Epoch 7/20
422/422 [==============================] - 2s 6ms/step - loss: 0.5415
- accuracy: 0.8906 - val_loss: 0.4615 - val_accuracy: 0.9152
Epoch 8/20
422/422 [==============================] - 4s 9ms/step - loss: 0.4915
- accuracy: 0.8979 - val_loss: 0.4185 - val_accuracy: 0.9203
Epoch 9/20
422/422 [==============================] - 3s 7ms/step - loss: 0.4539
- accuracy: 0.9045 - val_loss: 0.3917 - val_accuracy: 0.9188
Epoch 10/20
422/422 [==============================] - 4s 9ms/step - loss: 0.4274
- accuracy: 0.9083 - val_loss: 0.3657 - val_accuracy: 0.9252
Epoch 11/20
422/422 [==============================] - 3s 6ms/step - loss: 0.4030
- accuracy: 0.9127 - val_loss: 0.3488 - val_accuracy: 0.9258
Epoch 12/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3845
- accuracy: 0.9158 - val_loss: 0.3310 - val_accuracy: 0.9328
Epoch 13/20
422/422 [==============================] - 3s 6ms/step - loss: 0.3686
- accuracy: 0.9191 - val_loss: 0.3245 - val_accuracy: 0.9302
Epoch 14/20
422/422 [==============================] - 3s 8ms/step - loss: 0.3565
- accuracy: 0.9210 - val_loss: 0.3110 - val_accuracy: 0.9338
Epoch 15/20
422/422 [==============================] - 3s 7ms/step - loss: 0.3449
- accuracy: 0.9241 - val_loss: 0.3042 - val_accuracy: 0.9365
Epoch 16/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3338
- accuracy: 0.9259 - val_loss: 0.2940 - val_accuracy: 0.9385
Epoch 17/20
422/422 [==============================] - 3s 6ms/step - loss: 0.3248
- accuracy: 0.9284 - val_loss: 0.2879 - val_accuracy: 0.9383
Epoch 18/20
422/422 [==============================] - 3s 6ms/step - loss: 0.3152
- accuracy: 0.9307 - val_loss: 0.2798 - val_accuracy: 0.9428
Epoch 19/20
422/422 [==============================] - 4s 9ms/step - loss: 0.3083
- accuracy: 0.9321 - val_loss: 0.2804 - val_accuracy: 0.9402
```

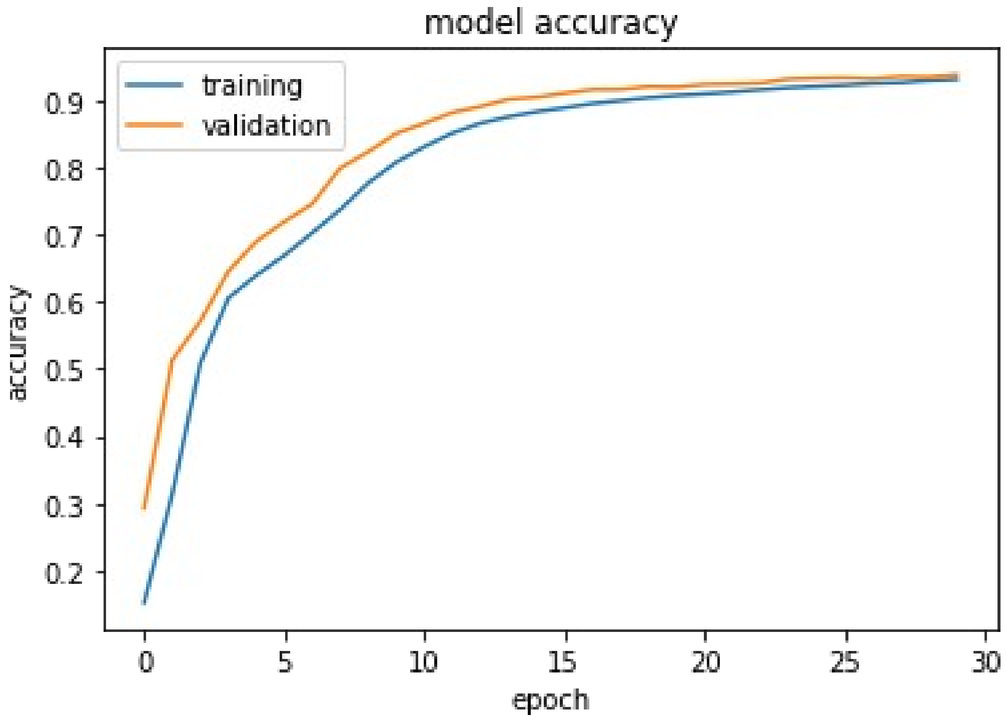
```
Epoch 20/20
422/422 [==============================] - 2s 6ms/step - loss: 0.3005
- accuracy: 0.9332 - val_loss: 0.2704 - val_accuracy: 0.9448
```



```
Test loss: 0.304
Test accuracy: 0.932
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_41"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_127 (Dense)           (None, 128)               100480

 dense_128 (Dense)           (None, 128)               16512

 dense_129 (Dense)           (None, 128)               16512

 dense_130 (Dense)           (None, 10)                1290

=================================================================
Total params: 134,794
Trainable params: 134,794
Non-trainable params: 0
_________________________________________________________________
Epoch 1/30
```

```
422/422 [==============================] - 4s 8ms/step - loss: 2.3553
- accuracy: 0.1519 - val_loss: 2.3209 - val_accuracy: 0.2930
Epoch 2/30
422/422 [==============================] - 4s 8ms/step - loss: 2.2944
- accuracy: 0.3139 - val_loss: 2.2612 - val_accuracy: 0.5132
Epoch 3/30
422/422 [==============================] - 4s 9ms/step - loss: 2.2223
- accuracy: 0.5079 - val_loss: 2.1697 - val_accuracy: 0.5708
Epoch 4/30
422/422 [==============================] - 3s 7ms/step - loss: 2.1090
- accuracy: 0.6053 - val_loss: 2.0247 - val_accuracy: 0.6452
Epoch 5/30
422/422 [==============================] - 3s 7ms/step - loss: 1.9363
- accuracy: 0.6392 - val_loss: 1.8174 - val_accuracy: 0.6893
Epoch 6/30
422/422 [==============================] - 3s 8ms/step - loss: 1.7126
- accuracy: 0.6685 - val_loss: 1.5747 - val_accuracy: 0.7188
Epoch 7/30
422/422 [==============================] - 3s 8ms/step - loss: 1.4797
- accuracy: 0.7024 - val_loss: 1.3452 - val_accuracy: 0.7455
Epoch 8/30
422/422 [==============================] - 3s 7ms/step - loss: 1.2727
- accuracy: 0.7369 - val_loss: 1.1508 - val_accuracy: 0.7988
Epoch 9/30
422/422 [==============================] - 3s 7ms/step - loss: 1.1049
- accuracy: 0.7767 - val_loss: 0.9960 - val_accuracy: 0.8230
Epoch 10/30
422/422 [==============================] - 3s 8ms/step - loss: 0.9720
- accuracy: 0.8075 - val_loss: 0.8745 - val_accuracy: 0.8503
Epoch 11/30
422/422 [==============================] - 3s 8ms/step - loss: 0.8655
- accuracy: 0.8306 - val_loss: 0.7758 - val_accuracy: 0.8655
Epoch 12/30
422/422 [==============================] - 3s 7ms/step - loss: 0.7768
- accuracy: 0.8508 - val_loss: 0.6935 - val_accuracy: 0.8815
Epoch 13/30
422/422 [==============================] - 3s 7ms/step - loss: 0.7024
- accuracy: 0.8655 - val_loss: 0.6256 - val_accuracy: 0.8905
Epoch 14/30
422/422 [==============================] - 4s 9ms/step - loss: 0.6411
- accuracy: 0.8752 - val_loss: 0.5695 - val_accuracy: 0.9012
Epoch 15/30
422/422 [==============================] - 3s 8ms/step - loss: 0.5915
- accuracy: 0.8827 - val_loss: 0.5262 - val_accuracy: 0.9043
Epoch 16/30
422/422 [==============================] - 3s 7ms/step - loss: 0.5520
- accuracy: 0.8885 - val_loss: 0.4912 - val_accuracy: 0.9103
Epoch 17/30
422/422 [==============================] - 3s 7ms/step - loss: 0.5184
- accuracy: 0.8949 - val_loss: 0.4603 - val_accuracy: 0.9157
```

```
Epoch 18/30
422/422 [==============================] - 4s 9ms/step - loss: 0.4927
- accuracy: 0.8993 - val_loss: 0.4410 - val_accuracy: 0.9158
Epoch 19/30
422/422 [==============================] - 3s 8ms/step - loss: 0.4700
- accuracy: 0.9035 - val_loss: 0.4218 - val_accuracy: 0.9202
Epoch 20/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4517
- accuracy: 0.9069 - val_loss: 0.4118 - val_accuracy: 0.9195
Epoch 21/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4359
- accuracy: 0.9093 - val_loss: 0.3947 - val_accuracy: 0.9233
Epoch 22/30
422/422 [==============================] - 4s 10ms/step - loss: 0.4220
- accuracy: 0.9121 - val_loss: 0.3812 - val_accuracy: 0.9240
Epoch 23/30
422/422 [==============================] - 3s 7ms/step - loss: 0.4093
- accuracy: 0.9155 - val_loss: 0.3701 - val_accuracy: 0.9252
Epoch 24/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3974
- accuracy: 0.9184 - val_loss: 0.3596 - val_accuracy: 0.9312
Epoch 25/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3867
- accuracy: 0.9204 - val_loss: 0.3546 - val_accuracy: 0.9323
Epoch 26/30
422/422 [==============================] - 4s 10ms/step - loss: 0.3787
- accuracy: 0.9224 - val_loss: 0.3454 - val_accuracy: 0.9328
Epoch 27/30
422/422 [==============================] - 3s 8ms/step - loss: 0.3678
- accuracy: 0.9248 - val_loss: 0.3381 - val_accuracy: 0.9315
Epoch 28/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3610
- accuracy: 0.9260 - val_loss: 0.3312 - val_accuracy: 0.9352
Epoch 29/30
422/422 [==============================] - 3s 7ms/step - loss: 0.3525
- accuracy: 0.9291 - val_loss: 0.3264 - val_accuracy: 0.9343
Epoch 30/30
422/422 [==============================] - 4s 10ms/step - loss: 0.3447
- accuracy: 0.9309 - val_loss: 0.3228 - val_accuracy: 0.9378
```

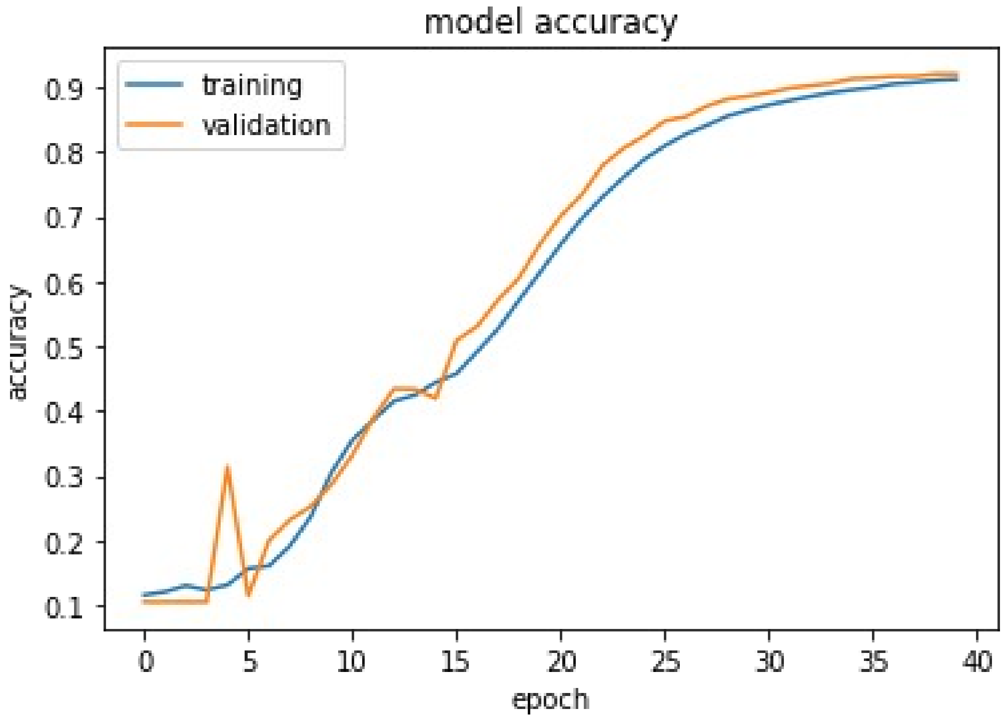


```
Test loss: 0.352
Test accuracy: 0.927
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_42"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 dense_131 (Dense)           (None, 128)               100480    
                                                                 
 dense_132 (Dense)           (None, 128)               16512     
                                                                 
 dense_133 (Dense)           (None, 128)               16512     
                                                                 
 dense_134 (Dense)           (None, 128)               16512     
                                                                 
 dense_135 (Dense)           (None, 10)                1290      
                                                                 
=================================================================
Total params: 151,306
Trainable params: 151,306
Non-trainable params: 0
_________________________________________________________________
Epoch 1/40
422/422 [==============================] - 6s 11ms/step - loss: 2.3808
- accuracy: 0.1159 - val_loss: 2.3763 - val_accuracy: 0.1050
```

```
Epoch 2/40
422/422 [==============================] - 3s 8ms/step - loss: 2.3738
- accuracy: 0.1208 - val_loss: 2.3724 - val_accuracy: 0.1050
Epoch 3/40
422/422 [==============================] - 3s 8ms/step - loss: 2.3702
- accuracy: 0.1303 - val_loss: 2.3686 - val_accuracy: 0.1050
Epoch 4/40
422/422 [==============================] - 5s 11ms/step - loss: 2.3664
- accuracy: 0.1236 - val_loss: 2.3642 - val_accuracy: 0.1050
Epoch 5/40
422/422 [==============================] - 5s 11ms/step - loss: 2.3620
- accuracy: 0.1314 - val_loss: 2.3592 - val_accuracy: 0.3142
Epoch 6/40
422/422 [==============================] - 3s 8ms/step - loss: 2.3568
- accuracy: 0.1568 - val_loss: 2.3534 - val_accuracy: 0.1147
Epoch 7/40
422/422 [==============================] - 5s 11ms/step - loss: 2.3504
- accuracy: 0.1609 - val_loss: 2.3461 - val_accuracy: 0.2002
Epoch 8/40
422/422 [==============================] - 3s 8ms/step - loss: 2.3425
- accuracy: 0.1917 - val_loss: 2.3378 - val_accuracy: 0.2320
Epoch 9/40
422/422 [==============================] - 3s 8ms/step - loss: 2.3324
- accuracy: 0.2368 - val_loss: 2.3253 - val_accuracy: 0.2527
Epoch 10/40
422/422 [==============================] - 4s 9ms/step - loss: 2.3191
- accuracy: 0.3054 - val_loss: 2.3103 - val_accuracy: 0.2868
Epoch 11/40
422/422 [==============================] - 4s 10ms/step - loss: 2.3006
- accuracy: 0.3550 - val_loss: 2.2871 - val_accuracy: 0.3318
Epoch 12/40
422/422 [==============================] - 3s 8ms/step - loss: 2.2735
- accuracy: 0.3857 - val_loss: 2.2543 - val_accuracy: 0.3885
Epoch 13/40
422/422 [==============================] - 3s 8ms/step - loss: 2.2327
- accuracy: 0.4151 - val_loss: 2.2049 - val_accuracy: 0.4342
Epoch 14/40
422/422 [==============================] - 5s 11ms/step - loss: 2.1709
- accuracy: 0.4244 - val_loss: 2.1282 - val_accuracy: 0.4335
Epoch 15/40
422/422 [==============================] - 4s 9ms/step - loss: 2.0820
- accuracy: 0.4440 - val_loss: 2.0249 - val_accuracy: 0.4193
Epoch 16/40
422/422 [==============================] - 4s 8ms/step - loss: 1.9702
- accuracy: 0.4576 - val_loss: 1.9014 - val_accuracy: 0.5087
Epoch 17/40
422/422 [==============================] - 5s 11ms/step - loss: 1.8436
- accuracy: 0.4917 - val_loss: 1.7635 - val_accuracy: 0.5308
Epoch 18/40
422/422 [==============================] - 4s 9ms/step - loss: 1.7051
```

```
- accuracy: 0.5268 - val_loss: 1.6167 - val_accuracy: 0.5715
Epoch 19/40
422/422 [==============================] - 3s 8ms/step - loss: 1.5635
- accuracy: 0.5713 - val_loss: 1.4748 - val_accuracy: 0.6057
Epoch 20/40
422/422 [==============================] - 3s 8ms/step - loss: 1.4361
- accuracy: 0.6137 - val_loss: 1.3537 - val_accuracy: 0.6573
Epoch 21/40
422/422 [==============================] - 4s 10ms/step - loss: 1.3301
- accuracy: 0.6569 - val_loss: 1.2538 - val_accuracy: 0.7007
Epoch 22/40
422/422 [==============================] - 3s 8ms/step - loss: 1.2367
- accuracy: 0.6964 - val_loss: 1.1610 - val_accuracy: 0.7333
Epoch 23/40
422/422 [==============================] - 3s 8ms/step - loss: 1.1491
- accuracy: 0.7301 - val_loss: 1.0725 - val_accuracy: 0.7787
Epoch 24/40
422/422 [==============================] - 4s 9ms/step - loss: 1.0656
- accuracy: 0.7601 - val_loss: 0.9887 - val_accuracy: 0.8052
Epoch 25/40
422/422 [==============================] - 4s 10ms/step - loss: 0.9884
- accuracy: 0.7880 - val_loss: 0.9128 - val_accuracy: 0.8245
Epoch 26/40
422/422 [==============================] - 3s 8ms/step - loss: 0.9175
- accuracy: 0.8094 - val_loss: 0.8446 - val_accuracy: 0.8477
Epoch 27/40
422/422 [==============================] - 3s 8ms/step - loss: 0.8556
- accuracy: 0.8272 - val_loss: 0.7870 - val_accuracy: 0.8542
Epoch 28/40
422/422 [==============================] - 5s 11ms/step - loss: 0.8010
- accuracy: 0.8406 - val_loss: 0.7322 - val_accuracy: 0.8703
Epoch 29/40
422/422 [==============================] - 3s 8ms/step - loss: 0.7513
- accuracy: 0.8551 - val_loss: 0.6882 - val_accuracy: 0.8813
Epoch 30/40
422/422 [==============================] - 3s 8ms/step - loss: 0.7059
- accuracy: 0.8644 - val_loss: 0.6435 - val_accuracy: 0.8860
Epoch 31/40
422/422 [==============================] - 4s 9ms/step - loss: 0.6665
- accuracy: 0.8724 - val_loss: 0.6071 - val_accuracy: 0.8915
Epoch 32/40
422/422 [==============================] - 4s 9ms/step - loss: 0.6312
- accuracy: 0.8793 - val_loss: 0.5748 - val_accuracy: 0.8983
Epoch 33/40
422/422 [==============================] - 3s 7ms/step - loss: 0.6016
- accuracy: 0.8856 - val_loss: 0.5463 - val_accuracy: 0.9022
Epoch 34/40
422/422 [==============================] - 4s 8ms/step - loss: 0.5734
- accuracy: 0.8913 - val_loss: 0.5200 - val_accuracy: 0.9057
Epoch 35/40
```

```
422/422 [==============================] - 4s 10ms/step - loss: 0.5503
- accuracy: 0.8957 - val_loss: 0.4980 - val_accuracy: 0.9125
Epoch 36/40
422/422 [==============================] - 3s 8ms/step - loss: 0.5307
- accuracy: 0.8992 - val_loss: 0.4825 - val_accuracy: 0.9138
Epoch 37/40
422/422 [==============================] - 3s 8ms/step - loss: 0.5123
- accuracy: 0.9049 - val_loss: 0.4695 - val_accuracy: 0.9162
Epoch 38/40
422/422 [==============================] - 3s 8ms/step - loss: 0.4962
- accuracy: 0.9071 - val_loss: 0.4566 - val_accuracy: 0.9163
Epoch 39/40
422/422 [==============================] - 4s 10ms/step - loss: 0.4839
- accuracy: 0.9104 - val_loss: 0.4461 - val_accuracy: 0.9198
Epoch 40/40
422/422 [==============================] - 3s 8ms/step - loss: 0.4691
- accuracy: 0.9119 - val_loss: 0.4317 - val_accuracy: 0.9198
```

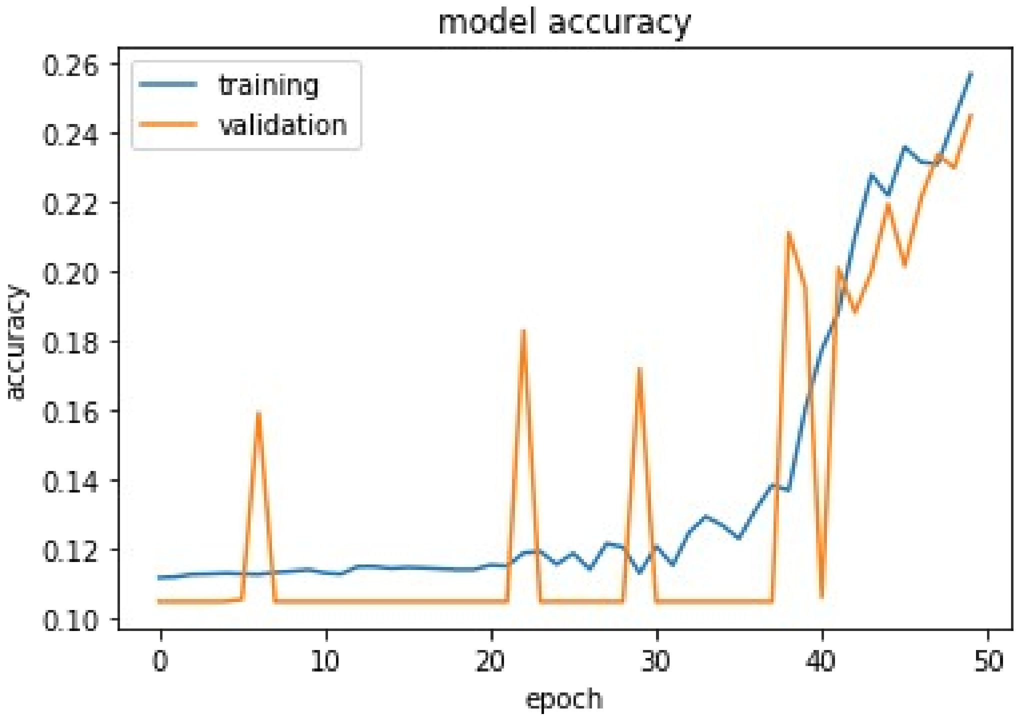


```
Test loss: 0.472
Test accuracy: 0.911
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
Model: "sequential_43"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_136 (Dense)           (None, 128)               100480

 dense_137 (Dense)           (None, 128)               16512

 dense_138 (Dense)           (None, 128)               16512

 dense_139 (Dense)           (None, 128)               16512

 dense_140 (Dense)           (None, 128)               16512

 dense_141 (Dense)           (None, 10)                1290

=================================================================
Total params: 167,818
Trainable params: 167,818
Non-trainable params: 0
_________________________________________________________________
Epoch 1/50
422/422 [==============================] - 7s 13ms/step - loss: 2.3989
- accuracy: 0.1119 - val_loss: 2.3971 - val_accuracy: 0.1050
Epoch 2/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3962
- accuracy: 0.1121 - val_loss: 2.3966 - val_accuracy: 0.1050
Epoch 3/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3956
- accuracy: 0.1129 - val_loss: 2.3964 - val_accuracy: 0.1050
Epoch 4/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3950
- accuracy: 0.1130 - val_loss: 2.3957 - val_accuracy: 0.1050
Epoch 5/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3945
- accuracy: 0.1132 - val_loss: 2.3963 - val_accuracy: 0.1050
Epoch 6/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3940
- accuracy: 0.1130 - val_loss: 2.3945 - val_accuracy: 0.1057
Epoch 7/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3933
- accuracy: 0.1129 - val_loss: 2.3935 - val_accuracy: 0.1593
Epoch 8/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3928
- accuracy: 0.1134 - val_loss: 2.3933 - val_accuracy: 0.1050
Epoch 9/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3922
- accuracy: 0.1137 - val_loss: 2.3921 - val_accuracy: 0.1050
Epoch 10/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3916
- accuracy: 0.1142 - val_loss: 2.3922 - val_accuracy: 0.1050
Epoch 11/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3912
- accuracy: 0.1133 - val_loss: 2.3917 - val_accuracy: 0.1050
```

```
Epoch 12/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3908
- accuracy: 0.1129 - val_loss: 2.3919 - val_accuracy: 0.1050
Epoch 13/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3903
- accuracy: 0.1151 - val_loss: 2.3910 - val_accuracy: 0.1050
Epoch 14/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3899
- accuracy: 0.1151 - val_loss: 2.3903 - val_accuracy: 0.1050
Epoch 15/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3893
- accuracy: 0.1146 - val_loss: 2.3895 - val_accuracy: 0.1050
Epoch 16/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3888
- accuracy: 0.1149 - val_loss: 2.3890 - val_accuracy: 0.1050
Epoch 17/50
422/422 [==============================] - 6s 14ms/step - loss: 2.3884
- accuracy: 0.1147 - val_loss: 2.3891 - val_accuracy: 0.1050
Epoch 18/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3879
- accuracy: 0.1144 - val_loss: 2.3880 - val_accuracy: 0.1050
Epoch 19/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3874
- accuracy: 0.1142 - val_loss: 2.3873 - val_accuracy: 0.1050
Epoch 20/50
422/422 [==============================] - 7s 16ms/step - loss: 2.3869
- accuracy: 0.1142 - val_loss: 2.3868 - val_accuracy: 0.1050
Epoch 21/50
422/422 [==============================] - 6s 14ms/step - loss: 2.3863
- accuracy: 0.1156 - val_loss: 2.3867 - val_accuracy: 0.1050
Epoch 22/50
422/422 [==============================] - 6s 15ms/step - loss: 2.3859
- accuracy: 0.1153 - val_loss: 2.3865 - val_accuracy: 0.1050
Epoch 23/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3852
- accuracy: 0.1190 - val_loss: 2.3857 - val_accuracy: 0.1830
Epoch 24/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3848
- accuracy: 0.1195 - val_loss: 2.3855 - val_accuracy: 0.1050
Epoch 25/50
422/422 [==============================] - 6s 15ms/step - loss: 2.3842
- accuracy: 0.1157 - val_loss: 2.3846 - val_accuracy: 0.1050
Epoch 26/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3834
- accuracy: 0.1189 - val_loss: 2.3838 - val_accuracy: 0.1050
Epoch 27/50
422/422 [==============================] - 7s 18ms/step - loss: 2.3830
- accuracy: 0.1143 - val_loss: 2.3844 - val_accuracy: 0.1050
Epoch 28/50
422/422 [==============================] - 6s 13ms/step - loss: 2.3824
```

```
- accuracy: 0.1217 - val_loss: 2.3833 - val_accuracy: 0.1050
Epoch 29/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3819
- accuracy: 0.1207 - val_loss: 2.3819 - val_accuracy: 0.1050
Epoch 30/50
422/422 [==============================] - 7s 17ms/step - loss: 2.3811
- accuracy: 0.1132 - val_loss: 2.3809 - val_accuracy: 0.1720
Epoch 31/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3801
- accuracy: 0.1210 - val_loss: 2.3810 - val_accuracy: 0.1050
Epoch 32/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3794
- accuracy: 0.1155 - val_loss: 2.3794 - val_accuracy: 0.1050
Epoch 33/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3785
- accuracy: 0.1250 - val_loss: 2.3786 - val_accuracy: 0.1050
Epoch 34/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3774
- accuracy: 0.1295 - val_loss: 2.3778 - val_accuracy: 0.1050
Epoch 35/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3763
- accuracy: 0.1269 - val_loss: 2.3770 - val_accuracy: 0.1050
Epoch 36/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3750
- accuracy: 0.1232 - val_loss: 2.3754 - val_accuracy: 0.1050
Epoch 37/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3735
- accuracy: 0.1314 - val_loss: 2.3728 - val_accuracy: 0.1050
Epoch 38/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3716
- accuracy: 0.1386 - val_loss: 2.3717 - val_accuracy: 0.1050
Epoch 39/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3694
- accuracy: 0.1371 - val_loss: 2.3683 - val_accuracy: 0.2112
Epoch 40/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3667
- accuracy: 0.1604 - val_loss: 2.3663 - val_accuracy: 0.1953
Epoch 41/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3637
- accuracy: 0.1775 - val_loss: 2.3629 - val_accuracy: 0.1062
Epoch 42/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3596
- accuracy: 0.1883 - val_loss: 2.3580 - val_accuracy: 0.2010
Epoch 43/50
422/422 [==============================] - 4s 9ms/step - loss: 2.3542
- accuracy: 0.2100 - val_loss: 2.3526 - val_accuracy: 0.1882
Epoch 44/50
422/422 [==============================] - 5s 13ms/step - loss: 2.3469
- accuracy: 0.2278 - val_loss: 2.3441 - val_accuracy: 0.2000
Epoch 45/50
```

```
422/422 [==============================] - 4s 10ms/step - loss: 2.3369
- accuracy: 0.2218 - val_loss: 2.3325 - val_accuracy: 0.2193
Epoch 46/50
422/422 [==============================] - 4s 10ms/step - loss: 2.3223
- accuracy: 0.2358 - val_loss: 2.3168 - val_accuracy: 0.2017
Epoch 47/50
422/422 [==============================] - 5s 12ms/step - loss: 2.3002
- accuracy: 0.2314 - val_loss: 2.2885 - val_accuracy: 0.2212
Epoch 48/50
422/422 [==============================] - 4s 10ms/step - loss: 2.2663
- accuracy: 0.2309 - val_loss: 2.2471 - val_accuracy: 0.2337
Epoch 49/50
422/422 [==============================] - 6s 13ms/step - loss: 2.2143
- accuracy: 0.2439 - val_loss: 2.1862 - val_accuracy: 0.2298
Epoch 50/50
422/422 [==============================] - 4s 10ms/step - loss: 2.1432
- accuracy: 0.2568 - val_loss: 2.1066 - val_accuracy: 0.2448
```



```
Test loss: 2.1
Test accuracy: 0.25
Shape of my predictions (test set): (10000, 10)
First prediction for number 2, probabilities: [0.004 0.046 0.885 0.003
0.    0.013 0.048 0.    0.002 0.   ]
```

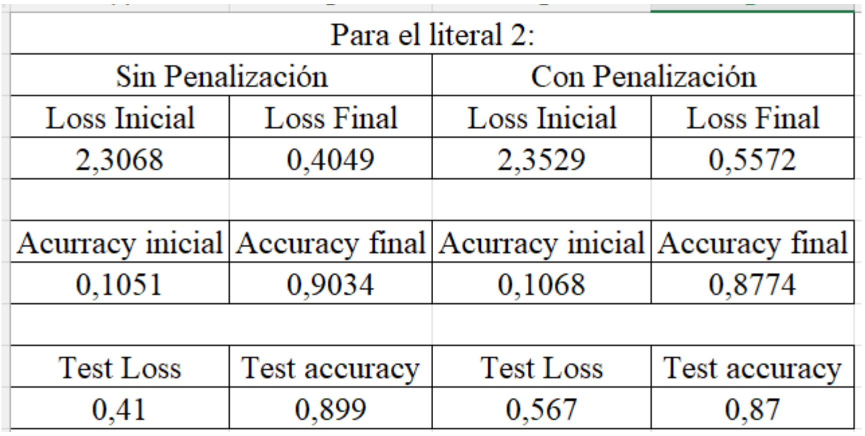
7. Comparación (2 puntos)

Resuma todos sus resultados en una tabla donde se verifique la precisión en el conjunto de entrenamiento, validación y test para las diferentes arquitecturas entrenadas con los hiperparámetros modificados (i.e., # de epochs, batch size, # de nodos por capa oculta, # capas ocultas, con y sin regularización).

Discuta sus resultados.

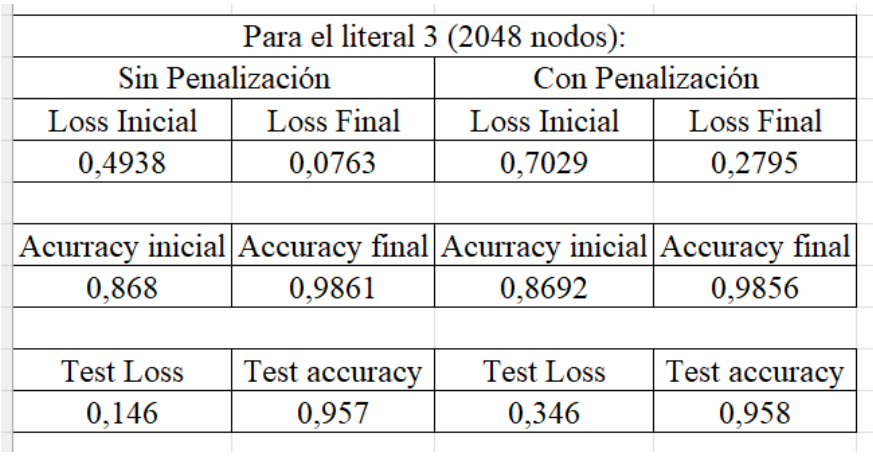
Todos los datos fueron los mismos, sin embargo podemos observar el cambio solo agregandole una penalización L2.

Para el literal 2



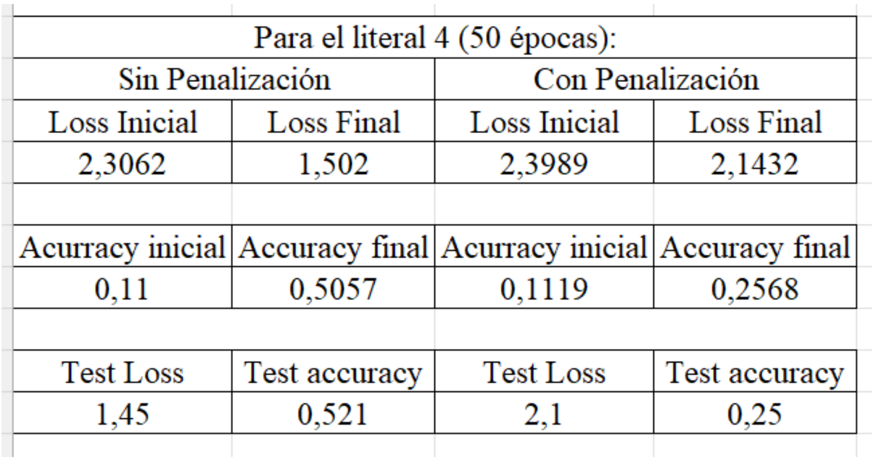
Para el literal 2:			
Sin Penalización		Con Penalización	
Loss Inicial	Loss Final	Loss Inicial	Loss Final
2,3068	0,4049	2,3529	0,5572
Acurracy inicial	Accuracy final	Acurracy inicial	Accuracy final
0,1051	0,9034	0,1068	0,8774
Test Loss	Test accuracy	Test Loss	Test accuracy
0,41	0,899	0,567	0,87

Para el literal 3:



Para el literal 3 (2048 nodos):			
Sin Penalización		Con Penalización	
Loss Inicial	Loss Final	Loss Inicial	Loss Final
0,4938	0,0763	0,7029	0,2795
Acurracy inicial	Accuracy final	Acurracy inicial	Accuracy final
0,868	0,9861	0,8692	0,9856
Test Loss	Test accuracy	Test Loss	Test accuracy
0,146	0,957	0,346	0,958

Para el literal 4.2:



Para el literal 4 (50 épocas):			
Sin Penalización		Con Penalización	
Loss Inicial	Loss Final	Loss Inicial	Loss Final
2,3062	1,502	2,3989	2,1432
Acurracy inicial	Accuracy final	Acurracy inicial	Accuracy final
0,11	0,5057	0,1119	0,2568
Test Loss	Test accuracy	Test Loss	Test accuracy
1,45	0,521	2,1	0,25

8. Conclusiones

El uso de redes neuronales es la mejor opcion para este tipo de problemas donde tenemos que clasificar, se combina todo lo que hemos visto en clases.

Del mismo modo, todo proceso se puede optimizar y es lo que hemos presentado. primero agregando mas capas a nuestra red, luego agregando funciones para generalizar en cualquier modelo que querramos usar a futuro.

La decisión de aplicar o no una penalización en el entrenamiento de redes neuronales depende del problema específico que se esté tratando de resolver y del conjunto de datos que se esté utilizando. En el caso de este deber, usamos la regularización L2. Hay que evitar el overfitting haciendo esto, aunque depende del conjunto de datos, cuando tenemos grandes conjuntos de datos como el que tenemos en este ejemplo vemos que no cambió mucho agregando la penalización (no es tan necesario).

9. Bibliografía

En caso de ser necesario, incluya la bibliografía utilizada en formato IEEE. No olvide citar en el texto sus referencias donde sea pertinente.

- ciberseg1922 (2021) ¿Qué es epoch en machine learning?, Ciberseguridad. Available at: https://ciberseguridad.com/guias/nuevas-tecnologias/machine-learning/epoch/ (Accessed: March 27, 2023).
- Furnieles, G. (2022) Sigmoid and Softmax functions in 5 minutes, Medium. Towards Data Science. Available at: https://towardsdatascience.com/sigmoid-and-softmax-functions-in-5-minutes-f516c80ea1f9 (Accessed: March 27, 2023).
- Patrikar, S. (2019) Batch, Mini Batch & stochastic gradient descent, Medium. Towards Data Science. Available at: https://towardsdatascience.com/batch-mini-batch-stochastic-gradient-descent-7a62ecba642a (Accessed: March 27, 2023).
- Tf.keras.metrics.categorical_crossentropy : tensorflow V2.12.0 (no date) TensorFlow. Available at:

https://www.tensorflow.org/api_docs/python/tf/keras/metrics/categorical_crossentropy (Accessed: March 27, 2023).

- ¿Cuál es el tamaño del lote en la red neuronal? (no date) QA Stack. Available at: https://qastack.mx/stats/153531/what-is-batch-size-in-neural-network (Accessed: March 27, 2023).