final-2

November 17, 2023

[1]:
```
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
import numpy as np
import random
```

```
C:\Users\HP\anaconda3\lib\site-packages\scipy\__init__.py:146: UserWarning: A
NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy
(detected version 1.26.2
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

[2]:
```
data = keras.datasets.mnist
```

[3]:
```
(x_train,y_train),(x_test,y_test) = data.load_data()
```

[4]:
```
x_train.shape
```

[4]:
```
(60000, 28, 28)
```

[5]:
```
x_test.shape
```

[5]:
```
(10000, 28, 28)
```

[6]:
```
x_train, x_test = x_train/255, x_test/255
```

[7]:
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Flatten, Dense
```

[8]:
```
model = Sequential([
    Flatten(input_shape=(28,28)),
    Dense(150,activation='relu'),
    Dense(10, activation='softmax')
])
```

[9]:
```
model.compile(optimizer='SGD',loss='sparse_categorical_crossentropy',␣
 ↪metrics=['accuracy'])
```

[10]:
```
model.fit(x_train,y_train,epochs=5)
```

```
Epoch 1/5
1875/1875 [==============================] - 4s 2ms/step - loss: 0.6377 -
accuracy: 0.8386
Epoch 2/5
1875/1875 [==============================] - 4s 2ms/step - loss: 0.3317 -
accuracy: 0.9071
Epoch 3/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2825 -
accuracy: 0.9194
Epoch 4/5
1875/1875 [==============================] - 4s 2ms/step - loss: 0.2517 -
accuracy: 0.9293
Epoch 5/5
1875/1875 [==============================] - 4s 2ms/step - loss: 0.2279 -
accuracy: 0.9366
```

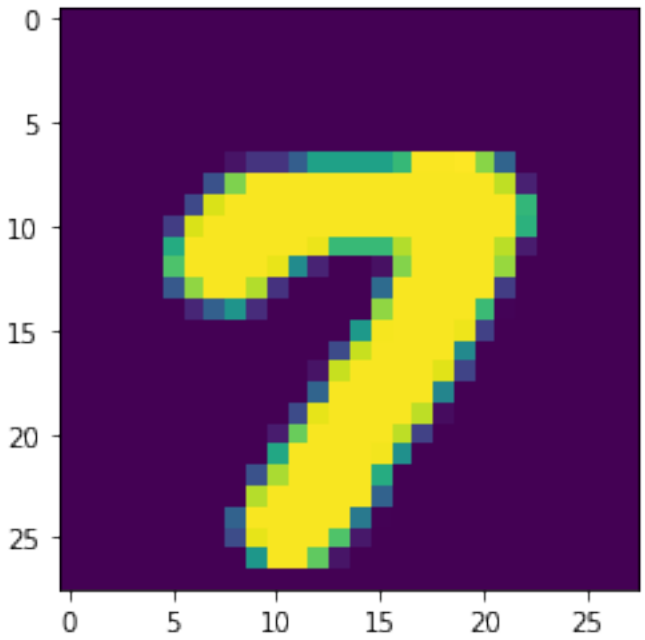
[10]:
```
<keras.src.callbacks.History at 0x265e611ba60>
```

[11]:
```
model.evaluate(x_test,y_test)
```

```
313/313 [==============================] - 1s 1ms/step - loss: 0.2126 -
accuracy: 0.9377
```

[11]:
```
[0.2125556617975235, 0.9376999735832214]
```

[15]:
```
idx = random.randint(0,len(x_test))
plt.imshow(x_test[idx])
plt.show()
```



[16]:
```
prediction = model.predict(x_test)
print("Model says it is ==> ",np.argmax(prediction[idx]))
```

```
313/313 [==============================] - 0s 860us/step
Model says it is ==>  7
```

[ ]:
```

```