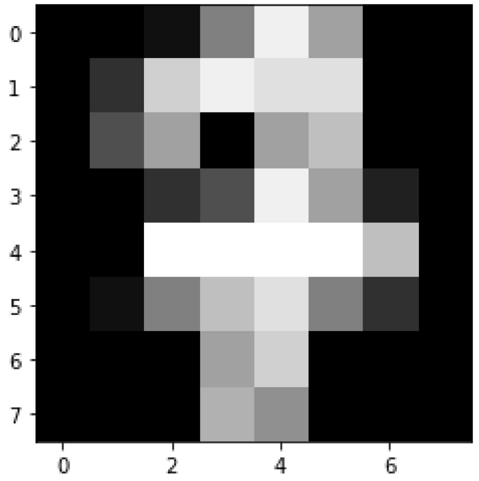
```
In [1]: from sklearn.datasets import load_digits
```

```
In [2]: digits = load_digits()
```

```
In [3]: %matplotlib inline
        import matplotlib.pyplot as plt
```

```
In [4]: idx = 17
        plt.imshow(digits['images'][idx], cmap=plt.cm.gray, interpolation='none')
```

```
Out[4]: <matplotlib.image.AxesImage at 0x216045fe588>
```



```
In [5]: digits['target'][idx]
```

```
Out[5]: 7
```

```
In [6]: digits['images'].shape
```

```
Out[6]: (1797, 8, 8)
```

```
In [7]: digits['data'].shape
```

```
Out[7]: (1797, 64)
```

In [8]:
```
from sklearn.model_selection import train_test_split
from keras.utils import np_utils
```

```
Using TensorFlow backend.
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:516: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:517: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:518: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:519: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:520: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorflow\python\framework\dtype
s.py:525: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is
deprecated; in a future version of numpy, it will be understood as (type,
(1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:541: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:542: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:543: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:544: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:545: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
```

```
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\danal\anaconda3\lib\site-packages\tensorboard\compat\tensorflow_stub
\dtypes.py:550: FutureWarning: Passing (type, 1) or '1type' as a synonym of t
ype is deprecated; in a future version of numpy, it will be understood as (ty
pe, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
```

In [11]:

```
x = digits['data']
y = digits['target']
```

In [12]:

```
y = np_utils.to_categorical(y)
```

In [13]:

```
y[0]
```

Out[13]:

```
array([1., 0., 0., 0., 0., 0., 0., 0., 0., 0.], dtype=float32)
```

In [14]:

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3)
```

In [15]:

```
in_dim = x.shape[1]
out_dim = y.shape[1]
```

In [16]:

```
from keras.models import Sequential
from keras.layers import Dense, Activation
```

In [18]:

```
model = Sequential()
model.add(Dense(128, input_shape=(in_dim,)))
model.add(Activation('relu'))
model.add(Dense(out_dim))
model.add(Activation('sigmoid'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accura
```

In [19]:

```
model.fit(x_train, y_train)
```

```
WARNING:tensorflow:From C:\Users\danal\anaconda3\lib\site-packages\tensorflow\p
ython\ops\math_grad.py:1250: add_dispatch_support.<locals>.wrapper (from tensor
flow.python.ops.array_ops) is deprecated and will be removed in a future versio
n.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From C:\Users\danal\anaconda3\lib\site-packages\keras\backen
d\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please
use tf.compat.v1.global_variables instead.

Epoch 1/1
1257/1257 [==============================] - 0s 92us/step - loss: 2.0412 - accu
racy: 0.3731
```

Out[19]:

```
<keras.callbacks.callbacks.History at 0x2160cdae288>
```

In [20]:

```
loss, accuracy = model.evaluate(x_test, y_test)
accuracy
```

```
540/540 [==============================] - 0s 54us/step
```

Out[20]:

```
0.6703703999519348
```

In [21]:

```
model.predict(x_test[:3])
```

Out[21]:

```
array([[2.02798843e-03, 4.67240810e-03, 6.99571967e-01, 8.75752270e-01,
        8.69631767e-05, 8.13013315e-03, 1.24342740e-02, 4.33176756e-04,
        9.66221094e-04, 1.23033345e-01],
       [4.86245751e-03, 9.29091334e-01, 3.73214483e-04, 5.63561916e-05,
        9.84420002e-01, 5.46365976e-04, 1.46991193e-01, 2.25435793e-02,
        8.16902518e-02, 1.12779737e-02],
       [5.88893890e-05, 5.11716545e-01, 5.74392080e-03, 3.22785974e-03,
        8.60227447e-04, 2.50576343e-03, 3.03164968e-04, 4.67498787e-02,
        7.11476028e-01, 2.20089321e-04]], dtype=float32)
```

In [22]:

```
model.predict(x_test[:3]).argmax(axis=1)
```

Out[22]:

```
array([3, 4, 8], dtype=int64)
```

In [24]:

```
y_test[:3].argmax(axis=1)
```

Out[24]:

```
array([3, 4, 1], dtype=int64)
```

In [25]:

```
model.save('digits.h5')
```

In [26]:

```
from keras.models import load_model
model1 = load_model('digits.h5')
```