Stepping up to images

Puteaux, Fall/Winter 2020-2021

```
################################
##                            ##
##  Deep Learning in Python   ##
##                            ##
################################
```

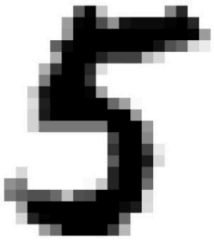
```
§1 Introduction to Deep Learning in Python
```

```
§1.4 Fine-tuning keras models
```

1 Stepping up to images

1.1 How to recognize handwritten digits?

- The MNIST dataset.
- 28 x 28 grid flattened to 784 values for each image.
- Value in each part of the array denotes the darkness of that pixel.



1.2 Practice exercises for model validation:

▶ Package pre-loading:

[1]:
```
import pandas as pd
from keras.layers import Dense
from keras.models import Sequential
from keras.utils import to_categorical
```

▶ Data pre-loading:

[2]:
```
df = pd.read_csv('ref8. MNIST.csv', header=None)
X = df.iloc[:, 1:].to_numpy()
y = to_categorical(df.iloc[:, 0])
```

▶ Evaluating model accuracy on validation dataset practice:

[3]:
```
# Create the model: model
model = Sequential()

# Add the first hidden layer
model.add(Dense(50, activation='relu', input_shape=(784, )))

# Add the second hidden layer
model.add(Dense(50, activation='relu'))

# Add the output layer
model.add(Dense(10, activation='softmax'))

# Compile the model
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# Fit the model
model.fit(X, y, validation_split=0.3)
```

```
44/44 [==============================] - 0s 11ms/step - loss: 44.9763 -
accuracy: 0.2184 - val_loss: 6.8900 - val_accuracy: 0.5524
```

[3]:
```
<tensorflow.python.keras.callbacks.History at 0x7fc623d39950>
```