

This cheatsheet will take you step-by-step through training a convolutional neural network in Python using the famous MNIST dataset for handwritten digits classification. Our classifier will boast over 99% accuracy.

Keras is our recommended library for deep learning in Python, especially for beginners. Its minimalist, modular approach makes it a breeze to get deep neural networks up and running.

To see the most up-to-date full tutorial, as well as installation instructions, visit the online tutorial at [elitedatascience.com](http://elitedatascience.com).

## SETUP

Make sure you have the following installed on your computer:

- Python 2.7+ or Python 3
- SciPy with NumPy
- Matplotlib (Optional, recommended for exploratory analysis)
- Theano\*

\*note: TensorFlow is also supported (as an alternative to Theano), but we stick with Theano to keep it simple. The main difference is that you'll need to reshape the data slightly differently before feeding it to your network.

## IMPORT LIBRARIES AND MODULES

```
import numpy as np
```

```
np.random.seed(123) # for reproducibility
```

```
from keras.models import Sequential
```

```
from keras.layers import Dense, Dropout, Activation, Flatten
```

```
from keras.layers import Convolution2D, MaxPooling2D
```

```
from keras.utils import np_utils
```

```
from keras.datasets import mnist
```

## LOAD PRE-SHUFFLED MNIST DATA INTO TRAIN AND TEST SETS

```
(X_train, y_train), (X_test, y_test) = mnist.load_data()
```

## PREPROCESS INPUT DATA

```
X_train = X_train.reshape(X_train.shape[0], 1, 28, 28)
```

```
X_test = X_test.reshape(X_test.shape[0], 1, 28, 28)
```

```
X_train = X_train.astype('float32')
```

```
X_test = X_test.astype('float32')
```

```
X_train /= 255
```

```
X_test /= 255
```

## PREPROCESS CLASS LABELS

```
Y_train = np_utils.to_categorical(y_train, 10)
```

```
Y_test = np_utils.to_categorical(y_test, 10)
```

## DEFINE MODEL ARCHITECTURE

```
model = Sequential()
```

```
model.add(Convolution2D(32, 3, 3, activation='relu',  
                        input_shape=(1,28,28)))
```

```
model.add(Convolution2D(32, 3, 3, activation='relu'))
```

```
model.add(MaxPooling2D(pool_size=(2,2)))
```

```
model.add(Dropout(0.25))
```

```
model.add(Flatten())
```

```
model.add(Dense(128, activation='relu'))
```

```
model.add(Dropout(0.5))
```

```
model.add(Dense(10, activation='softmax'))
```

## COMPILE MODEL

```
model.compile(loss='categorical_crossentropy',
```

```
              optimizer='adam',
```

```
              metrics=['accuracy'])
```

## FIT MODEL ON TRAINING DATA

```
model.fit(X_train, Y_train,
```

```
        batch_size=32, nb_epoch=10, verbose=1)
```

## EVALUATE MODEL ON TEST DATA

```
score = model.evaluate(X_test, Y_test, verbose=0)
```

To see the most up-to-date full tutorial, explanations, and additional context, visit the [online tutorial at elitedatascience.com](http://online tutorial at elitedatascience.com). We also have plenty of other tutorials and guides.