



DEEP LEARNING PROGRAMMING EXERCISE № 3

DM873 / DS809, Fall 2022

05/10/2022

To do these exercises, you will use Python 3 and the following packages:

- [TensorFlow](#). The used backend used for Keras.
- [Keras](#). Good front-end package for easier integration of neural networks.
- [Matplotlib](#). This package allows you to graph your data and data transformations.

These can all be installed using pip, the python package manager. You are not strictly forced to use these packages, but it is highly recommended. Feel free to use other packages you think are necessary.

Installations

Run the following commands on terminal if not installed;

Listing 1: Installations of required Python packages

```
1 pip install tensorflow
2 pip install matplotlib
```

Problem

This exercise will extend on the model created during Exercise 1.

Listing 2: Load MNIST dataset.

```
1 import numpy as np
2 from keras.datasets import mnist
3 from keras.utils import to_categorical
4 import matplotlib.pyplot as plt
5 # Loads the data
6 (train_data, train_labels), (test_data, test_labels) = mnist.load_data()
7 # Plots a single digit from the data
8 plt.imshow(train_data[1])
9 plt.show()
10 # Reshapes the data to work in a FFN
11 train_data = train_data.reshape((60000, 28*28))
12 test_data = test_data.reshape((10000, 28*28))
13 num_classes = len(np.unique(train_labels))
14 train_labels = to_categorical(train_labels, num_classes)
15 test_labels = to_categorical(test_labels, num_classes)
```

What to do?

- Create an identical model, this time adding some regularization and using the Functional API. Can you increase the accuracy of the model?
- Try adding dropout between some layer, where do you find this has the highest impact?
- How does L1 compare to L2 regularization?
- Vary the number of units, the amount of layers, activation functions, etc. to obtain the best accuracy you can.