

Summer DS 2020 Python Project

Set up: Install TensorFlow

Follow the instructions at <https://www.tensorflow.org/install>. Test installation to avoid problems. For Windows machines, we recommend using a virtual environment if you encounter problems with the installation or linking.

MNIST Dataset

In this assignment we will be using the MNIST digit dataset (<http://yann.lecun.com/exdb/mnist/>). The dataset contains images of hand-written digits (0-9), 28x28 pixels and the corresponding labels. You can use the tensorflow build-in functionality to download and import the dataset into python.

```
mnist = tf.keras.datasets.mnist
```

Whether you choose the above, or downloading the datasets from the website, please make sure you have the following: training set: 60K samples, testing set: 10k samples.

We suggest that you use tensorflow for your neural network models. We also suggest that you install scikit-learn if you would like to experiment with non-neural network based techniques.

Task: MNIST - Python classification

Here we will be looking at implementing models to classify MNIST digits. We would like you to implement a variety of neural network models of your choosing, this could be anything from shallow networks, deep networks, convolutional, or even recurrent nets. We would like their performance to be compared and contrasted with details of the hyper-parameters, training and testing errors, and any other parameters used in the analysis. Also consider training times, where did the models perform poorly, and anything else you can think of to differentiate. Comparisons with non-neural techniques are also highly recommended.

Credit will be given for the implementation of a wide range of techniques, particularly those that are deemed novel and/or creative. Consider such things as the network architecture, transfer functions, optimiser.

Output: Deadline 17th July

1. Team presentation (10 minutes) discussing your findings:
 - What different techniques did you use?
 - Which things worked and why do you think they worked?
 - Which things didn't work and why do you think they didn't?
 - How did you work as a team?
 - What lessons did you learn?
 - **Please do not use the PwC branding at any point**
2. Email a zip file containing:
 - README - explaining how to run the code if it isn't simply running a .py
 - Code directory - clearly named files etc
 - requirements file

Resources:

- <http://neuralnetworksanddeeplearning.com/index.html>
- <https://www.tensorflow.org/tutorials/quickstart/beginner>
- https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html
- Books
 - Pattern Recognition and Machine Learning
 - The Elements of Statistical Learning
- Weekly office hour - Tuesdays at 11
 - Book ahead via email (will.barnsley@pwc.com)