

ROB311 – TP4 – SVM Digit Recognition

Introduction

Today, you will use **Support Vector Machines** and **Python** in order to implement a **digit recognition** algorithm. The database used to train and test your algorithm is the **MNIST dataset**, containing grayscale (8-bit), 28x28 pixels images of hand written digits. This is one of the reference digit recognition datasets in the world and, as you can see, state-of-the-art SVM algorithms can achieve error rates as low as 0.56 to 1.4 %.

Files

The two .csv files containing the MNIST dataset (both training and test set) [can be downloaded here](#) (using the Download button). Each of the two files contains 785 columns, the first column corresponding to the **label** of each sample (a digit from 0 to 9), while the other 784 columns contain the **colour intensity value** (8-bit, 0 to 255) for each of the pixels of a 28x28 image.

The training set (*mnist_train.csv*) contains 60.000 samples, while the test set (*mnist_test.csv*) contains 10.000 samples.

Objectives

Implement the digit recognition algorithm using Support Vector Machines trained on the MNIST hand written digit dataset contained in the *mnist_train.csv* file. Then, test your algorithm on the provided *mnist_test.csv* file.

You will have to compute:

- **the overall detection accuracy** (the percentage of correctly recognised digits from the test set)
- **a confusion matrix** (of size 10x10)

Both the detection accuracy and confusion matrix can be simply displayed in a terminal, but feel free to use any graphic libraries you want to display them.

Good news!

You can use sklearn or any other library you want :)