Homework on Nearest Neighbor (Python Version)

In this homework you will implement the *Nearest Neighbor* classification algorithm that you have learned in class. The database that you will be working on is the MNIST. The MNIST database contains in total of 60000 train images and 10000 test images of handwritten digits. This database is a benchmark for many classification algorithms, including neural networks. For further reading, visit the following link (**note that you should manually copy the whole link**): https://web.archive.org/web/20220331130319/https://yann.lecun.com/exdb/mnist

The implementation of the classification algorithm will be done with Python.

- Open the `students_python.zip` file attached to the assignment. The zip contains three files:

  1. `assignment.py` - The main script, where you will write your code.
  2. `utils.py` - Contains aid functions. The first is to load the database, and the second is to plot a digit from the database.
  3. `MNIST_3_and_5.mat` - A prepared dataset of digits 3 and 5 from the MNIST database.

  Make sure you've installed the right packages in the headers of the `.py` files. If everything is alright, after running the main script you should see a new `.txt` file in the folder.

- You should first load `MNIST_3_and_5.mat` and get a feel of the data. The data is arranged in matrixes and vectors, in the following variables:

  1. `Xtrain, Ytrain` - 11552×784 matrix with images and 11552×1 vector with the corresponding digits.
  2. `Xvalid, Yvalid` - 1522×784 matrix with validation images, and 1522×1 vector with the corresponding digits.
  3. `Xtest` - 1902×784 matrix with test images.

  In all datasets, each row of the matrix is an image of size 28×28. The `plot_sample(..)` function from `utils.py` is an example of how to plot a digit. Some of the digits from the database are depicted in Fig. 1. As you can see, not all the digits look the same.

- Write a code in the saved place in `assignment.py`. You code should performs classification for the unlabeled data in `Xtest` using the *K-Nearest Neighbor* algirithm. Use `Xtrain, Ytrain` as the training set for the algorithm.

  The algorithm requires you to choose two main parameters:

  1. $d(x, \tilde{x})$ - a metric for measuring a distance between two samples.

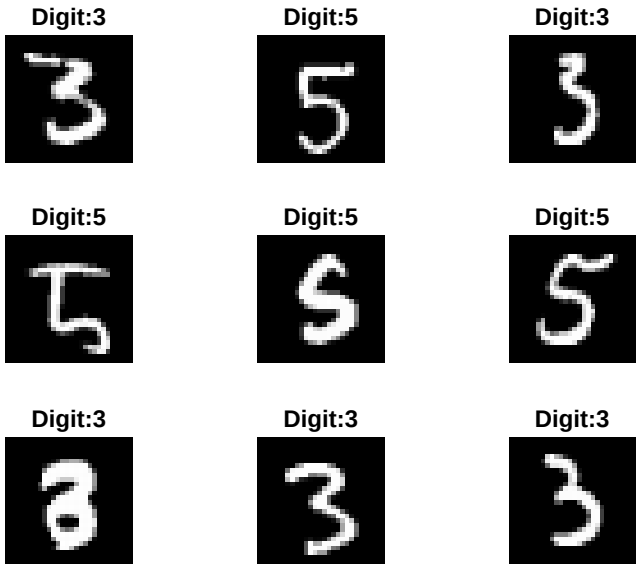

Figure 1: Example of digits from the MNIST database

2. K - The number of nearest neighbors to use for a classification.

Use `Xvalid, Yvalid` as a validation set, in order to determine which metric and number of neighbors to choose.

- Classify the images in `Xtest`. Your classification should result in a vector `Ytest`, where the i-th element is a classification result for the i-th row of `Xtest`
- Save the vector of the results in a `ID.txt` (your ID as the file name). Each row in the text file should be a classification of the corresponding example (first row for first example and so on).

  For example, a student with ID 123456789 should create a file `123456789.txt`, that contains 1902 rows.

Good luck!