Artificial Neural Network and Deep Learning

Project #1: Back-Propagation Neural Network Implementation and Application to Handwritten Digits Recognition

Project Deadline: October 21, 2022
Presentation Date: October 28, 2022

Subject:

Implement the **back-propagation** neural network algorithm to classify/recognize the handwritten digits. A database includes more than 40,000 examples of MNIST handwritten digits. The dataset contains ten folders, namely 0, 1, 2, …, and 9. Each folder contains 4,000 sample images of the corresponding digit. The MNIST database is a large database of handwritten digits that is commonly used for training various image processing systems. This database is also widely used for training and testing in the machine learning field. It was created by "re-mixing" the samples from NIST's original datasets. All digits have been size-normalized and centered in a fixed-size image (28*28 pixel, PNG format) as shown in the following examples.

It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal effort on preprocessing and formatting.

Note:

Design your code using any programming language. **DO NOT** apply any open source code of BP nor existing BP library/function for this project. To ensure the program is designed by yourself, you will be requested on-site on October 21st to re-train the BP model for a given dataset with part of the digits and the variations.

Evaluation Method:

1. For network performance evaluation, you will need to meet TA on October 21st. The evaluation time and place will be announced later. TA will give you a new set of digits, you need to modify your program and train on the new dataset. After you have finished the training as requested on-site, meet TA for evaluation.

TA will prepare a set of test images with file names 0001.png, 0002.png, …, and 5000.png as illustrated in the following figure.

2. Your program needs to read all test images, perform the recognition for each digit image, and then generate a text file called Student_ID.txt. The output format is shown in the following format. The output file must contain two columns: first column is the filename; the second column is the recognition result of the digit. For example, 0001 represents a file name of test image 0001.png and 0 represents your recognition result as digit 0.
3. Then TA will check your output file and compute the accuracy of your recognition results. You have two chances of evaluations. However, if you want to do the test again, the evaluation will be counted based on your second test. Accuracy= total # of correct outputs/total # of test images

Example: testing images

Output: Student_ID.txt