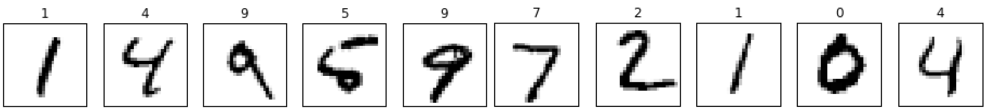
Intelligent Systems: Reasoning and Recognition

James L. Crowley

MoSIG M1 — Winter Semester 2020-2021

Exercise 5 — 11 March 2021

Recognizing Digits using Neural Networks



The MNIST (Modified National Institute of Standards and Technology) dataset is a large collection of handwritten digits composed of 60,000 training images and 10,000 test images. The database was created by "re-mixing" samples of digits from NIST's original datasets taken from American Census Bureau employees and American high school students as part of the 1990 US Census. The black and white images from NIST were normalized to fit into a 28x28 pixel bounding box and anti-aliased, which introduced gray-scale levels.

Your task is to design and evaluate neural network architectures that can recognize hand-drawn digits using the gray-scale MNIST images. Your networks can consist of convolution layers, fully connected layers, an auto-encoder, or any combination of these or other neural network techniques. You may use weights that are pre-trained on a different dataset or trained from scratch. Your must provide objective performance evaluation, using metrics such as error rates, ROC curves, accuracy, precision and recall. It is recommended that you use Keras with the tensorflow backend libraries running in a minconda environment to help you in completing this exercise. However, you may use any software tools and framework that you wish.

This project should be performed by teams of 3 students, and should be described by a written report with descriptions and analysis of performance evaluation results for the techniques that are tested. Reports may be written in French or English. Programming teams are given freedom in their choice of techniques to evaluate. The following is an indicative barometer for grading. Actual grades will depend on a subjective appreciation for the amount of effort deployed and the depth of understanding displayed in the results, and the clarity of the report. Creativity is encouraged and will be rewarded! Reports are due midnight Monday 12 April 2021.

Grade	Example of Criteria (Max grade is 20)
8	Construct and describe a fully connected multi-layer network to recognize MNIST digits. Divide the MNIST training data into a training set (80%), an evaluation set (10%) and a test set (10%). Use the training set to train your network over multiple epochs while displaying accuracy for your training set and your evaluation set. Stop training when the accuracy scores diverge. Present results for Accuracy, Precision, Recall, F1, AUC for ROC curves, using the test set.
+1 to +4	Using your initial network as a baseline technique, demonstrate the effects of changing the number of layers (+1) and the number of units per layer (+1), learning with different learning rates (+1) and using different optimization techniques, by providing Accuracy, Precision, Recall, F1, and AUC for ROC curves for each network.
8	Implement a multi-layer Convolutional network to recognize MNIST digits similar to the network provided as an example on the course web site, or similar to LeNet5. Use the training set to train your network over multiple epochs while displaying accuracy for your training set and your evaluation set. Stop training when the accuracy scores diverge. Present results for Accuracy, Precision, Recall, F1, AUC for ROC curves, using the test set.
+1 to +4	Using your initial convolution network as a baseline technique, demonstrate the effects of changing the number of convolutional layers (+1) and the number of filters per layer (+1), different types and sizes of pooling layers (+1), and number of fully connected layers (+1), by providing Accuracy, Precision, Recall, F1, and AUC for ROC curves for each network.
+1	Demonstrate the effects for different values of dropout.
+2	Document the performance of a published network (LeNet5, VGG, Yolo, etc) for recognizing MNIST Digits.
+2 to +6	Use your best network to build a real time system to recognize your own hand written digits.