Homework 3

Special Topics in Advanced Machine Learning
Spring 2017
Instructor: Anna Choromanska

Homework is due 03/28/2017.

Problem 1 (20 points): Classification

Download dataset "dataset.t7b", which consists of 4 two-dimensional Gaussians that correspond to 4 different classes. Randomly permute this data set. Design a fully-connected network with at least one hidden layer that solves the resulting classification problem. Implement this network in Torch. Take 10% of the data as testing and the rest as training. Use NLL (negative log-likelihood) as your loss function. Show the architecture of your design, plot the train and test loss as a function of the training time (report the loss after each data mini-batch), and report the final training and test accuracy.

Problem 2 (20 points): Regression

Generate a 2-D data set, where data points (x, y) are randomly drawn from a uniform distribution on a plane $[-10, 10] \times [-10, 10]$. Let the data set have 5000 data points. Let $f(x, y) = x^2 + xy + y^2$. Design a fully-connected neural network with at least one hidden layer to predict the value of the function f given (x, y). Implement this network in Torch. Take 10% of the data as testing and the rest as training. Use MSE (mean-squared error) as your loss function. Show the architecture of your design, plot the train and test loss as a function of the training time (report the loss after each data mini-batch), and report the final training and test loss.

Problem 3 (20 points:): Neural Networks

Look at the TensorFlow Playground as given in `http://playground.tensorflow.org/`. For each of the four data sets (3 points (first data set) + 4 points (second data set) + 5 points (third data set) + 8 points (fourth data set)), where each data set consists of data points from 2 different classes (they are shown on the left side of the screen), design an architecture that solves the binary classification problem. Try to design as small network as possible (minimum number of neurons). Play with the: architecture, learning rate, activation functions, and regularization. Use default values for the ratio of train to test data, noise, and batch size. In the final write-up show for each data set one screenshot (the screenshot should show your entire screen: the architecture, parameters, and the obtained output).

Problem 4 (40 points): CNNs

Design and implement in Torch a CNN architecture which as input takes two images of digits from MNIST dataset and outputs 1 if both images correspond to the same digit and 0 otherwise. Use only 10% of MNIST training data set as your training data set and only 10% of MNIST test data set as your test data set. The obtained datasets need to be then modified appropriately for this learning task. You network should take as input a 4-dimensional tensor: (batch size, plane count - *each plane corresponds to a different input digit image*, image height, image width). Report the architecture and the full evaluation study of the performance of your network.