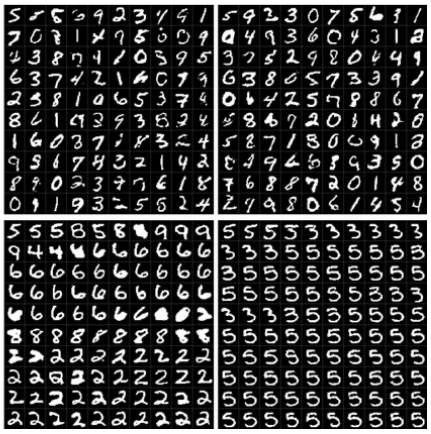
277B: Machine Learning Algorithms

Homework #6: Deep Learning and Regularization

Assigned Mar. 5 and Due Mar. 17

1. Bias-variance tradeoff (25 pts). We will use the MNIST data set to train, validate, and test using a deep learning network. The input representation will be raw image vector for each number comprised of 32x32 pixels(=1024 in length) with one channel of real values between 0 (black) and 255 (white). Use cross-entropy as the loss function.



(a) (5 pts) normalize the entire data set by dividing each number image by its maximum pixel value for numerical stability.

(b) (10 pts) The data folder you are given is already organized into a training set and a test set. Therefore, to create a validation set, divide the original training data into 3-fold groups of training (2/3 of the training data), validation (1/3 of the training data). 3-fold is because you will run 3 independent training/validation sets.

(c) (5 pts) Devise an ANN that has 2 computing layers: a hidden layer of size 3 neurons and the final output layer of 10 output neurons and use a sigmoid activation function. Use the ADAM optimizer with learning rate of 2e-3, batchsize of 128, and 50 epochs. The prediction y_j^μ is maximum posteriori (winner take all). Converge your training to where the loss function is minimal, and plot your training curve and validation accuracy without any further regularization, and comment on the bias-variance tradeoff with this choice.

(d) (5 pts) Devise another ANN with hidden layer of size 50. Do the same as in (1c). Plot your training and validation curve, and comment on the bias-variance tradeoff with this choice.

2. Deep Learning and regularization (20 pts)

(a) (5 pts) Using the ANN from 1(d), utilize dropout with 15%. Compare your training and test accuracy to results in (d).

(b) (5 pts) Using the ANN from 1(d), utilize L2 regularization with lambda=1e-5. How does the result compare to (d)?

(c) (5 pts) Use principal component analysis on the input, to create a reduced set of input features, keeping 99% of the variance. This is a type of data transformation! How many parameters do you have in this case and how does it compare to the original model (i.e. the ANN from 1(d))?

(d) (5 pts) Use the regularization settings (2(a) or 2(b)) that give the best result so far, and using the reduced input space from (c) run the model again. Is the training faster and better this time?