Assignment: Regularization, Perceptron, and Deep Neural Networks

Problem 1 (20 points)

Consider a learning scenario where the goal is to learn the target function $f(x) = sin(\pi x)$ for $-1 \leq x \leq +1$ from two points in the training sets. The two training points in R^2 have a uniform distribution between -1 and +1. You will create two models in linear hypothesis set $y = mx + b$: 1) unregularized, 2) weight-decay regularized (use L_2 regularization with $\lambda = 0.1$).

1. (5 points) Generate 10,000 hypotheses for each version. Report the average hypothesis $\bar{g}(x)$ in each case.
2. (5 points) Find and report $bias^2$ for each model.
3. (5 points) Find and report variance for each model
4. (5 Points) For each case, plot $\bar{g}(x) \pm \sqrt{var}$ along with $\bar{g}(x)$ and target function $f(x) = sin(\pi x)$. Which model will you choose? Why?

   Round your answers to 3 decimal places.

Problem 2 (20 points)

An online rental video company is interested in creating a model to make movie recommendations to one of its customers, Ms. X. As a consultant to this company, you are provided with the history of the movies that she accepted or rejected to watch. She makes her selections solely based on the movie genre and critic ratings. The data is in *movieData.csv* on Canvas.

1. Train a perceptron that will create a linear boundary decision that will help the company to make future recommendations to Ms. X. After how many iterations does the algorithm converge?
2. Upon creating a model, plot the boundary line along with all the data points and axes clearly marked.

Problem 3 (40 points)

In this problem you will use the data in *siCoData.csv* file to train a neural network. Use the backpropagation algorithm to train a 3-layer (input, hidden, output) neural network. Use stochastic gradient decent (SGD) technique and assume that the activation function for the hidden layer and output layer are tanh and linear, respectively. (You must write *your own code* for BP and SGD).

1. The stopping criteria for training in this problem should be a combination of achieving a minimum in-sample error

$$E_{in} = \frac{1}{N} \sum_{n=1}^{N} e_n$$

and reaching a maximum number of epochs (In this expression N is the number of observations in the data set and e_n is the error corresponding to each individual training point). Report the minimum E_{in} that you could achieve along with the related weights and number of iterations.

2. Graph the original data (y vs. x) and the predicted values ($\hat{y}$ vs. x) on two separate scatter plots.

Problem 4 (20 points)

Use the keras package to create an MLP deep NN and a convolutional NN to classify mnist data set. Examine different architectures and fine tune parameters to improve classification results.

1. Report the architecture of the MLP deep NN and hyper parameters that resulted in the best performance. What is the classification error?

2. Report the architecture of the convolutional NN and hyper parameters that resulted in the best performance. What is the classification error?

3. Report the most important architectural manipulations that lead to improving the performance.