CSE/ECE 343/543: Machine Learning
Assignment-3B

Max Marks:60 **Due Date**: 11:59PM Nov. 29, 2019

Instructions

- Keep collaborations at high level discussions. Copying/Plagiarism will be dealt with strictly.
- Start early, solve the problems yourself. Some of these questions may be asked in Quiz/Exams.
- Use Google Colab to write and train your models. You can follow this tutorial to get started with Google Colab. You can upload your data on Google Drive and access it from there.
- Late submission penalty: As per course policy.
- Submission guidelines: Please refer to the post on google classroom. We will strictly follow them.

PROGRAMMING QUESTIONS

1. (35 points) **Neural Networks.**

   Implement the forward pass and back propagation algorithms to train an artificial neural network for MNIST (attached) from scratch. You are not permitted to use any external libraries in this part. Split the data into a train, validation and test set. The test set should be held out.

   a) (*20 points*) The arguments to your function would be the number of layers and the number of nodes in each layer. Assume a Sigmoid activation function in each layer and Softmax in the output layer. Use cross-entropy Loss. You have to construct a neural network architecture using 3 hidden layers - [100, 50, 50 units] and the activations mentioned above. Implement forward and backward propagation, which can be used for a general fully-connected neural network with Softmax layer as the output. Use this function to train this architecture and test them using the MNIST dataset. Save the weights of your best models. Please ensure that your code is general and does not hard code equations.

   b) (*5 points*) Show 'loss vs iterations' and 'accuracy vs iterations' graphs on the train and validation set to establish that your model is well-trained and can generalize well. Report the final accuracy and loss on your train, validation and test set. Give proper justifications with challenges you faced and counter-measures you took.

   c) (*5 points*) Visualize the features (output of the last hidden layer) of the final hidden layer by creating tsne plots. You may use inbuilt library functions from sklearn for this.

d) (*5 points*) Implement part A using sklearn. Report the accuracy and loss, as compared to what you received with your own network. You do not have to plot graphs for this part. Explain reasons for an observed difference in accuracies, if any.

2. (25 points) Use the binary CIFAR 10 subset (attached) for this part. Use the existing AlexNet Model from PyTorch (pretrained on ImageNet) as a feature extractor for the images in the CIFAR subset. You should use the fc8 layer as the feature, which gives a 1000 dimensional feature vector for each image. Train a linear SVM from the sklearn library over these extracted feature vectors for classification. Report the test accuracy along with the confusion matrix and the ROC curve. You may use sklearn functions for the confusion matrix and ROC curve.