Engineering Applications of Machine Learning and Data Analytics
Homework #4 [Due: 03/30/2022]

I acknowledge that this exam is solely my effort. I have done this work by myself. I have not consulted with others about this exam in any way. I have not received outside aid (outside of my own brain) on this exam. I understand that violation of these rules contradicts the class policy on academic integrity.

Name: Zhaohui Yang

Signature: Zhaohui Yang

Date: 03/19/2022

Instructions: There are four problems. X Partial credit is given for answers that are partially correct. No credit is given for answers that are wrong or illegible. Write neatly.

You must submit two PDFs on D2L. The first PDF has the results to the analytical questions as well as figures that are generated

Problem 1: ____________ Problem 2: ____________

Problem 3: ____________ :

Total: ____________

1 Multi-Layer Perceptron [20pts]

In class we discussed the derivation of the backpropagation algorithm for neural networks. In this problem, you will train a neural network on the CIFAR10 data set. Train a Multi-Layer Perceptron (MLP) neural network on the CIFAR10 data set. This is an opened implementation problem, but I expect that you implement the MLP with at least two different hidden layer sizes and use regularization.

- Report the classification error on the training and testing data each configuration of the neural network. For example, you should report the results in the form of a table

	Classification Error	
	training	**testing**
50HLN+no regularization	0.234	0.253
50HLN+L_2 regularization	0.192	0.203
250HLN+no regularization	0.134	0.153
250HLN+L_2 regularization	0.092	0.013

  List all the parameters that you are using (i.e., number of learning rounds, regularization parameters, learning rate, etc.)
- I would suggest using Google's TensorFlow, PyTorch or Keras library to implement the MLP; however, you are free to use whatever library you'd like. If that is the case, here is a link to the data

- I recommend using a cloud platform such as Google Colab to run the code.

2 Adaboost [20pts]

Write a class that implements the Adaboost algorithm. Your class should be similar to sklearn's in that it should have a `fit` and `predict` method to train and test the classifier, respectively. You should also use the sampling function from Homework #1 to train the weak learning algorithm, which should be a shallow decision tree. The Adaboost class should be compared to sklearn's implementation on datasets from the course Github page.

3 Recurrent Neural Networks for Languange Modeling [20pts]

Read "LSTM: A Search Space Odyssey" (https://arxiv.org/abs/1503.04069). One application of an RNN is the ability model language, which is what your phone does when it is predicting the top three words when you're texting. In this problem, you will need to build a language model.

You are encouraged to start out with the code here. While this code will implement a language model, you are required to modify the code to attempt to beat the baseline for the experiments they have implemented. For example, one modification would be to train multiple language models and average, or weight, their outputs to generate language. Write a couple of paragraphs about what you did and if the results improve the model over the baseline on Github.