Neural Networks and Deep Learning
Lab Assignment 1: Feedforward Neural Networks

Summary: In this assignment, you will demonstrate you understand how to train, evaluate, and analyze feedforward neural networks. Your submission must include one PDF file that merges two separate PDF documents showing (1) your code and (2) report.

1. **Neural Network Hyperparameters [15 points]:**

   (a) Design and conduct your experiment (Code)

   - Load a real dataset not covered in class that is designed for the classification problem; e.g., from sklearn.datasets, Kaggle, your own data, etc. If you are seeking assistance to find a dataset, one good option is to revisit the TA's programming tutorials. You can pre-process the dataset if you wish.
   - Create a 70/30 train/test split of the dataset.
   - Train at least nine neural network models using all possible combinations of at least three different numbers of hidden layers and three types of activation functions. Set all other hyperparameters constant when training; e.g., numbers of neurons per layer, number of iterations for training, batch size, and optimization approach.
   - Evaluate each model on the test set using a confusion matrix and accuracy.

   (b) Report your methods, results, and analysis (Report)

   - Describe the methods you used for your experiment. This should include a discussion of the dataset (e.g., source? number of examples?) and what parameters/hyperparameters were used to train all the models.
   - Report the results for each tested model.
   - Discuss your analysis of general trends that emerge from your results. Your discussion should consist of 2-4 paragraphs. To format each paragraph, please first identify one general trend observed from the results and then offer insights/speculations into why you think the trend/results may occur (whether you deem the results good or bad). Possible trends to consider include: Did a certain number of hidden layers lead to consistently better results? Did a certain type of activation function lead to consistently better results? What insights are gained by looking at the different evaluation metrics?

2. **Impact of Training Duration and Training Data [10 points]:**

   (a) Design and conduct your experiment (Code)

   - Load a real dataset not covered in class that is designed for the classification problem.
   - Create a 70/30 train/test split of the dataset.
   - Train neural networks using five approaches: train with 20%, 40%, 60%, 80%, and 100% of the training data respectively. For this experiment, select one set of hyperparameters to use for the neural networks and keep those constant when training; e.g., number of hidden layers, number of neurons per layer, activation function, batch size, and optimization approach.
   - Evaluate the models by creating one plot with five learning curves; i.e., plot the performance of each of the five approaches with respect to the number of epochs used during training. For evaluation, use the accuracy metric.

   (b) Report your methods, results, and analysis (Report)

   - Describe the methods you used for your experiment. This should include a discussion of the dataset and the parameters/hyperparameters used to train all the models.
   - Show the plot that visualizes the performance for each of the five approaches.
   - Discuss your analysis of general trends that emerge from your results. Your discussion should consist of 2-4 paragraphs. To format each paragraph, please first identify one general trend observed from the results and then offer insights/speculations into why you think the trend/results may occur (whether you deem the results good or bad). Possible trends to consider include: What is the influence of the amount of training data? What is the influence of the the training duration?

How to Submit Lab Assignment 1: Please submit a PDF file named with your first and last name; i.e., firstname_lastname.pdf. It must merge two separate PDF documents: one showing your code[1] and another showing your report. Your report must be a self-contained document to get full credit. We will only review the code in detail when the report is not a self-contained document in order to provide partial credit.

Collaboration versus Academic Misconduct: Collaboration with other students is permitted, but the work you submit must be your own. Copying/plagiarizing work from another student is not permitted and is considered academic misconduct. For more information about University of Colorado Boulder's Honor Code and academic misconduct, please visit the course syllabus.

[1]We require submitting the code as a PDF to avoid many issues that we have observed in the past with being able to access submitted code. These issues have arisen, in part, because we make no programming language requirements. Issues also have arisen from students not providing read permissions for links to their files; e.g., for Google Colab.