Columbia University
IEOR4742 – Deep Learning for OR & FE (Hirsa)
Assignment 1 – Feedforward Neural Networks

- Objective: Implement a simple feedforward neural network from scratch to classify a dataset (`MNIST`).
- Tasks:
  - build and train a neural network without using deep learning frameworks.
  - visualize the training process, including loss and accuracy over epochs.
  - discuss the impact of different hyperparameters (e.g., learning rate, activation functions).

Problem 1 (Impact of non-linear activation functions on Learning): One can show a feed-forward neural network with linear activation function and any number of hidden layers is equivalent to just a linear neural neural network with no hidden layer. Prove it.

Problem 2 (Impact of different number of layers, different activation functions, and optimization on learning): The code *logistic_regression_multi_2layer.ipynb* is a 2-layer logistic regression model. The goal is to extend it to 3- and 4-layer logistic regression with different numbers of neurons and activation functions for each layer, utilizing different optimization routines to assess their impact on accuracy. Consider the following six feedforward architectures:

(1) 1^{st} layer: tanh, 2^{nd} layer: sigmoid, 3^{rd} layer: leaky ReLU

(2) 1^{st} layer: tanh, 2^{nd} layer: sigmoid, 3^{rd} layer: sigmoid, 4^{th} layer: ReLU

(3) 3 layers all sigmoid

(4) 3 layers all leaky ReLU

(5) 4 layers all tanh

(6) 4 layers all leaky ReLU

Try using 50 and 100 neurons per layer. Use `RMSProp` and `Adam` as the optimizers and compare their performance. For a few different starting points, use the linear interpolation discussed during the lecture to visualize the error surface.