VirginiaTech
Invent the Future
The Bradley Department
Electrical & Computer Engineering

ECE/CS 6524 – Deep Learning

Homework Assignment: Multilayer Perceptrons (MLPs)

In this assignment, you are asked to use Keras and Tensorflow to implement MLPs and evaluate the performance of MLPs for multiclass classification.

If this is your first Keras/Tensorflow project, please read a tutorial on machine learning with Keras to learn about Keras/Tensorflow: “Your First Deep Learning Project in Python with Keras Step-By-Step” **(**https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/).

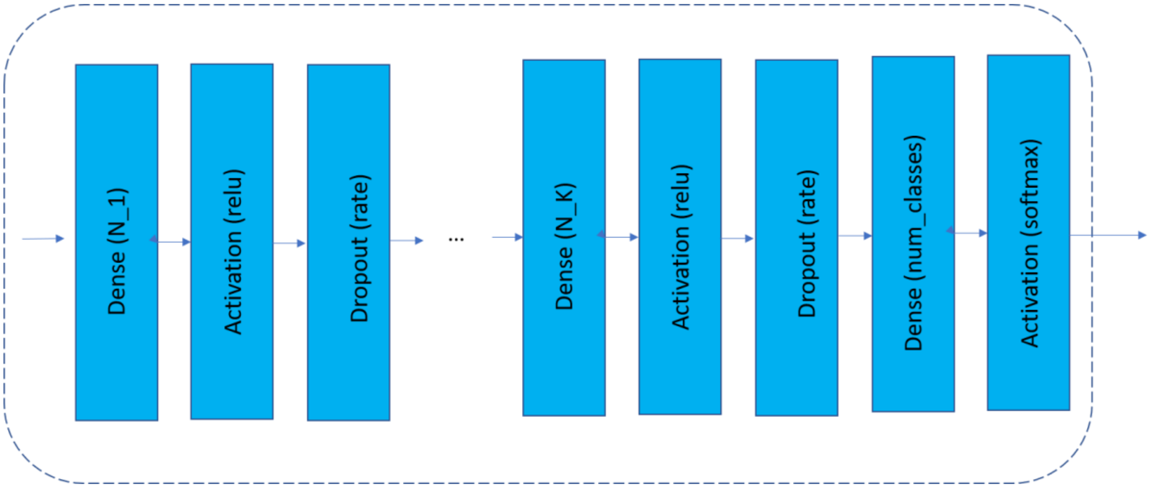
Note that for installing Keras and Tensorflow, you may also find a doc with detailed instructions: “How to install TensorFlow and Keras using Anaconda Navigator — without the command line” in the Canvas system (this assignment).

(1) Implement feedforward neural networks, i.e., multilayer perceptrons, for class identification in the **CIFAR10** and **CIFAR100** datasets, which can be loaded as follows:

```
tensorflow.keras.datasets.cifar10.load_data()
tensorflow.keras.datasets.cifar100.load_data(label_mode="fine")
```

(*hint:* The intensity of images should be scaled from [0, 255] to [0.0, 1.0], i.e., the pixel value is normalized to avoid large gradient values that could make training difficult. The label/class variable should be converted to **one-hot vector,** a vector with all elements 0, except for the index of the specified class. The loss function for one-hot vector is termed ‘categorical_crossentropy’ in Keras/Tensorflow.)

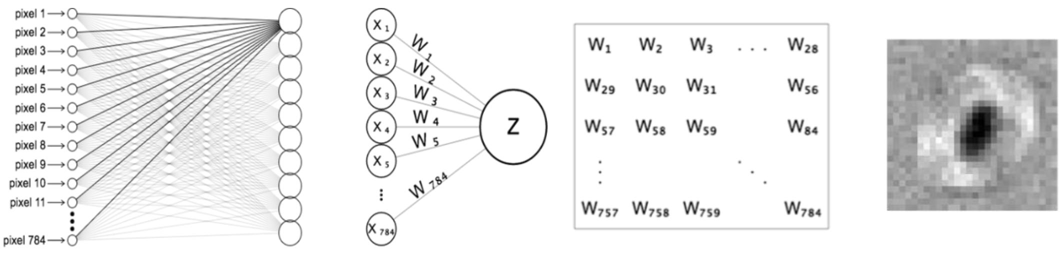
The MLP model shown below can be used for multiclass classification. The output layer consists of *num_classes* nodes with softmax activation function.



(2) Evaluate the performance of MLPs with (i) **different network configurations**: number of layers (e.g., 2, 3, 4, 5, …) and number of hidden nodes (e.g., 64, 256, 512, …); (ii) **with or without regularization** (i.e., with or without dropout (e.g., dropout rate = 0.2, 0.4, 0.6)); (iii) **different optimizers** (e.g., SGD, Adam, etc.).

Summarize the classification performance (accuracy) in a table and **discuss** about your understanding of the experimental results in terms of network configurations, dropout and optimization techniques.

(3) For a simple neural network without hidden layers (i.e., connecting input layer and output layer directly), visualize the weights to understand some ‘features’ extracted by the neural network. (Below is *an example* of such ‘features’ shown to be possibly related to digit 0.) For MLPs with hidden layers, what about the weights in the first layer? What about the weights in the hidden layers?



You need to prepare **a written report** (in the pdf format) including the following two sections: (1) Experimental Results, and (2) Discussion.

In addition, you need to attach your **implementation codes** as separate files to the report.