EE 526X Deep Machine Learning: Theory and Practice — Homework 4
Assigned: 11/10. Due: 11/17

Problem 1. Use the `tf.keras` available in Tensorflow 2.0 to design a CNN for classifying fashion clothes. The dataset to be used is the Fashion MNIST dataset (see `https://github.com/tensorflow/docs/blob/master/site/en/tutorials/keras/classification.ipynb`). You need to decide the number of layers, the kernel sizes, mini-batch size, whether to use dropout, training optimizer, learning rate, etc. In your report, you need to include the following items:

(a) A summary of the model used, including the number of parameters, and the other hyper-parameters used.

(b) A calculation of the number of floating point operations needed in each forward pass.

(c) Assume that the backward propagation's complexity is three-times the complexity of the forward pass, estimate the number of floating point operations per iteration step (per mini-batch).

(d) Estimate the number of floating point operations per epoch, and the total number of operations over the whole training process.

(e) A plot of the training accuracy and test accuracy as a function of the number of iterations.

The source code should be submitted as a separate .py file.

Problem 2. The data file "data.npy" contains a matrix, of 100 rows and 500 columns. Each row represents a signal, which contains the superposition of a large number (larger than 20) of sinusoids with randomly generated amplitudes, frequencies, and phases. The matrix can be loaded using `numpy` command `load(‘‘data.npy’’)`.

Design a recurrent neural network (RNN) that uses a signal $x[n]$ as input and outputs a signal $y[n]$, which predicts the value of $x[n+1]$. The designed network should have the following features:

- Should have no more than 4 layers and no less than 3 layers;
- The last layer should be non-recurrent, and uses linear activation, i.e., no non-linear activation;
- The other layers should all be recurrent;
- The total number of neurons in the RNN before unrolling should be no more than 50;
- Mean squared error should be used as the error criterion.

The network should be trained based on the first 66 signals and tested using the last 34 signals. The implementation should be based on Tensorflow 2.0, using `tf.keras` interface.

A typed report should be submitted describing the network designed, issues encountered, training and testing performance, and how the design choices were made. The source code should be submitted as a separate .py file.

END OF ASSIGNMENT