MULTILAYER PERCEPTRON (MLP)

Filename: -

1) Neural Network in ipynb format.

2) MLP.py

Running Instruction – Build the architecture by stacking Linear layer classes and activation classes into a list. Load the dataset and batch loader is used to divide the dataset into batches and pass the batches along with the neural network list, optimizer type, and learning as parameters into the trainnetwork function.

Methodology

1) The feed forward neural network with back propagation is implemented using classes and functions.
2) The Neural network designed can run with all kinds of optimisers.

Helper Functions

1) Class Linear – It implements the Layer which takes an input and outputs the affine combination of weights, biases, and input. It has two important methods forward and backward and rest of the methods are optimisers such as SGD, NAG, Adam, RMSProp, AdaGrad.
2) forward () – It gives the affine combination as discussed above.
3) backward () – it updates the weights with the help of gradient received from subsequent layers and forward the gradient to the next layer.
4) Class Activation – It implements the activation function such ass tanh, sigmoid, ReLu.
5) Trainnetwork () – Function to train the network.
6) Dataloader () – It does one hot encoding for multiclass datasets to feed into neural network.
7) BatchLoader () – It divides the dataset into batches.
8) Plot () – It plots the loss vs epoch curve for each architecture.
9) Save_model_dict () – It saves the trained model into pkl format.
10) Load_model_dict () – It loads the saved model for testing or further tuning.

Pre-processing

1) One hot encoding of labels is performed.
2) The dataset is divided into batches.

3) Min max normalisation is performed over the dataset.