

### Assignment 5a

- a) In this assignment, we run train and test the classification of CIFAR 10 datasets using VGG 16 architecture
- b) Use your own code for each of the layer. You can use conv2d, relu, maxpool function anyway from pytorch. That's okay. You should keep each layer separate (each convolution layer, each maxpool layer, each fully connected layer, relu layer, activation layer) so that you can change layer architecture such as changing activation function, add batch normalization, adding dropout in a layer, etc.
- c) Write your training module so that you can run the training for 100 epochs. Also write the code in a way that you can change learning rate anytime with appropriate algorithms.
- d) For optimizers, you should understand the implementation function for all the optimization methods that we have learned in theory. We should be able to write your own optimization method.
- e) Accuracy and F1-score on the test set
- f) Print Confusion Matrix

### Assignment 5b

- a) In this assignment, we run train and test the classification of CIFAR 10 datasets using VGG 16
- b) You will use your VGG 16 code from 5a
- b) Implement following modules with your own code
  - a. Batch normalization function. Use this function after dense layers to do batch normalization
  - b. Implement dropout function. Use the dropout function in each dense layer
  - c. Add L2 regularization term with your cross entropy loss.
  - d. Use Xavier initialization for your weights. Understand hows the xavier initialization works.
  - e. Implement a learning rate scheduler algorithm. Linear/Cosine/square root. Use this function to use the learning rate change over epochs.
  - f. Write a normalization function to normalize your input data to standard normal. Std: .5, mean: .5
  - g. Implement an augmentation function that will create five different versions of one image. Use the five versions with the original one for training. You can create augmented images from all the training data and keep them in a folder with their label information. Then use the proper data loader for training.
- c) Write codes to train the VGG 16 model with augmented train dataset from CIFAR. In this case the Net/model will be the VGG 16. Use 80% as training, 20% as validation data from whole training data
- d) Plot loss curve and accuracy curve for training and validation
- e) Accuracy and F1-score on the test set.
- f) Print Confusion Matrix