STOR566
Due date: Sep 18, 11:55PM
Keywords: SVM, MLP, CNN

Homework 2
Introduction to Deep Learning, Fall 2022

Submission Instruction:

- Please answer the questions using the jupyter notebook script.
- For submission, please include your code, code output and answers in the script and submit it on sakai.
- Please don't modify existing cells in the script. But you can add cells between the exercise statements.
- To make markdown, please switch the cell type to markdown (from code) - you can hit 'm' when you are in command mode - and use the markdown language. For a brief tutorial see: `https://daringfireball.net/projects/markdown/syntax`.

References:

- You can follow the setup instructions at here.
- A useful tutorial on learning pytorch by examples at here.
- More illustrations of different optimizers could be found here.
- Check Pytorch optimization methods at here.
- Check Pytorch data augmentation options at here.

Evaluation Metrics of Classifiers:

- Average loss of an epoch:

$$\frac{1}{B}\sum_{b=1}^{B}\sum_{d=1}^{D_b}\frac{loss(y_{bd}, f(\boldsymbol{x}_{bd}))}{D_b}$$

  for each training epoch.
  - B: the total number of batches
  - D_b: the number of observations in b-th batch
  - f: the model (Logistic regression or Linear SVM or MLP or CNN)
  - loss: logistic loss or the loss of linear SVM or cross-entropy
  - $(\boldsymbol{x}_{bd}, y_{bd})$: the d-th pair of input data and label in b-th batch
  - An epoch is defined as one iteration over all observations in the training dataset
- Testing accuracy:

$$\frac{1}{N}\sum_{i=1}^{N}\mathbf{1}(\hat{y}_i = y_i)$$

  - N: the total number of samples in the testing set
  - y_i: true label of sample i
  - $\hat{y}_i$: predicted label by the model

1 Problem 1. (40 points)

In this problem you will practice implementing Linear SVM and Logistic Regression to classify **handwritten digit 0 and 1**.

Data. You will use MNIST digit classification dataset. Pytorch/torchvision has provide a useful dataloader to automatically download and load the data into batches. In this homework, you need two class, digit 0 and digit 1, for binary classification. Code of the data loader has been provided in the template. Please don't modify the data loading part.

Problem Description.

1. (20 points) Implement **Logistic Regression** with Pytorch to do handwritten digit 0 vs. 1 classification. Pick an optimizer yourself.
   (a) (5 points) Report the hyper-parameters (number of epochs, learning rate, momentum etc).
   (b) (10 points) Report the **Average loss of an epoch** for every epoch by generating Average Loss vs. Epoch plot. Please report at least **10** epochs.
   (c) (5 points) Report the final testing accuracy of trained model.
2. (20 points) Implement **Linear SVM** with Pytorch to do handwritten digit 0 vs. 1 classification. Pick an optimizer yourself.
   (a) (5 points) Report the hyper-parameters (number of epochs, learning rate, momentum etc).
   (b) (10 points) Report the **Average loss of an epoch** for every epoch by generating Average Loss vs. Epoch plot. Please report at least **10** epochs.
   (c) (5 points) Report the final testing accuracy of trained model.

2 Problem 2. (60 points)

In this problem you will practice implementing MLP and CNN to classify daily life images (CIFAR10).

Data. You will use CIFAR10 classification dataset (10 classes). Pytorch/torchvision has provide a useful dataloader to automatically download and load the data into batches. Code of the data loader has been provided in the template. Please don't modify the data loading part.

Problem Description.

1. (20 points) Implement a 7 layers fully-connected neural networks with ReLU activation to do image classification.
   (a) (5 points) Print the model architecture.
   (b) (10 points) Report the **Average loss of an epoch** for every epoch by generating Average Loss vs. Epoch plot. Please report at least **10** epochs.
   (c) (5 points) Report the final testing accuracy of trained model.
2. (30 points) Implement a 7 layers CNN with 4 convolutional layers, 3 fully-connected layers and ReLU activation function. The input dimension of the 1st fully-connected layer must be 4096.
   (a) (5 points) Print the model architecture.
   (b) (10 points) Report the **Average loss of an epoch** for every epoch by generating Average Loss vs. Epoch plot. Please report at least **10** epochs.
   (c) (5 points) Report the final testing accuracy of trained model.
   (d) (10 points) Write a new cifar_loaders function to try different data augmentation methods.
3. (10 points) Please compare results of the models (MLP and CNN).