MA3832-Practical 4: Multi-layer perceptron neural network

Dr. Kelly Trinh

Sofwares

1. Install `Tensorflow` (version 2 or >) and `Keras` on your computer.

Goals

1. Implementing techniques for regularisation, dealing with unstable gradients, and adaptive learning rate.
2. Transfer learning

Data Description

In this section, we work with the Fashion MNIST dataset, which contains 60000 small suare 28×28 pixel grayscale images of 10 types of clothing 0: T-shirt/top
1: Trouser
2: Pullover
3: Dress
4: Coat
5: Sandal
6: Shirt
7: Sneaker
8: Bag
9: Ankle boot

Our task is to construct a MLP to classify the types of clothing. Comments on the performance of the MLP.

1. Repeat step 1-3 from Practical 3
2. Build a neural network with the following structure:
   - Number of Hidden layers = 2: The first hidden layer contains 64 neurons while the second layer includes 32 neurons.
   - Activation function: Both hidden layers will use a non-saturating activation function, and `he_normal` initialization.

- Include a dropout layer at the last layer
- Optimizer: Momemtum optimisation with learning rate is 0.001 and momentum = 0.9
- Use early stop in training a model

3. Repeat the exercise using `nadam` optimiser with exponential scheduling, `selu` activation function, `lecun_normal` initialization.

4. **Reuse pretrained layers**
   Let's split the fashion MNIST training set into two datasets:

   - X_train_A: all images of all items except for sandals and shirts (classes 5 and 6).
   - X_train_B: the first 200 images of sandals and shirts.

   We will first

   - train a model on dataset A (classification task with 8 classes). The model used to train dataset A has the following structure:
     - It has 5 hidden layers. The first hidden layer contains 300 neurons, the second has 100 neurons, the last three hidden layers contain 50 neurons each.
     - the activation function is `scaled relu`
     - loss function is `sparse_categorical_crossentropy`
   - reuse the model for dataset A to classify sandals and shirts in dataset B.