Project

Announcement: 25 February 2022

Submission Deadline: 29 March 2022

Description

The objective of the project is for you to gain a practical understanding of the methods discussed in the class relating to image classification. You will learn about linear classifiers, convolutional neural networks, loss functions, and activation functions.

Instructions

We provide a Python Jupyter notebook with the skeleton code. Mundane tasks are already implemented e.g. downloading and preparing the KMNIST image datasets. Your task is to read through the skeleton code and comments, and implement the missing parts which start with the comment "**### TODO:**". Where applicable, argument types and variable names are explained. There are 3 parts:

Part 0: Import KMNIST dataset

1. visualize a sample image and corresponding label from KMNIST

Part 1: Feed-forward Neural Network

1. Activation functions (Sigmoid, ReLU, Identity)
2. Loss function (SoftMax, Cross Entropy)
3. Network (weights, biases, gradients)
4. Training (train/test loss/accuracy)

Part 2: Convolutional Neural Networks

1. Stanford Dogs datasets
2. Create a custom CNN using Pytorch
3. Optimization (Adam, learning rate)
4. Visualize filters

Any libraries other than the ones already imported in the provided script **are not allowed.**

You can either complete the project on your local machine or using an online service such as Google Colab. We recommend using Google Colab since you will have free access to GPUs which will reduce the training time. However, **using Google Colab is not a requirement** and you should have no trouble completing the project on your local machine using only CPUs. In this case, you will first have to install all the Python libraries used in the script.

Extra credit for undergraduates/Compulsory for graduates:

A. Implement a my_conv2d class to replace torch.nn.Conv2d.

Extra credit for all:

A. Implement Xavier initialization to initialize network weights. Do not use the torch.nn.init module for this task.

Submission (electronic submission through EAS only)

Please create a zip file containing your Python script (**1 source file**) and a readme text file (.txt). In the readme file document the features and functionality you have implemented, and anything else you want the grader to know i.e. control keys, keyboard/mouse shortcuts, etc, if applicable.